REGULARISED METHODS FOR HIGH-EFFICIENCY PROPAGATION

Jacco Geul[*], Erwin Mooij[†], and Ron Noomen[‡]

Although regularised propagation methods have a good performance (accuracy versus evaluations), they suffer from a number of practical difficulties, such as propagation to a fixed time, making them ill-suited for practical applications. Several methods that address these limitations are proposed, thoroughly discussed, and analysed on diverse test cases. Dromo outperforms the conventional propagation methods significantly. It is shown that regularised methods, through some adaptations, can be successfully applied to different orbit problems. The proposed method is recommended especially for computationally demanding problems.

INTRODUCTION

Numerical integration has become the standard for solving problems in orbital mechanics, in favour of analytical and semi-analytical techniques [1]. Cowell's method, however, has the major drawback of being far less efficient, due to the many computations at intermediate time steps. Regularised propagation methods offer a significant reduction in steps for the same numerical accuracy [2, 3]. Through the transformation of the independent variable and equations of motion (EOMs), the numerical and dynamical stability is improved.

Research in satellite orbit propagation is mainly focused on the modelling of different perturbations. These models present the dominant contribution in terms of accuracy. However, these more advanced models come at a computational price. Especially when the propagation of a large catalogue of objects is concerned, the computational efficiency becomes crucial. Recently, regularisation methods have regained momentum [4–11].

Regularised methods, nonetheless, have only been scarcely adopted for solving real orbit problems. This can likely be attributed to the more difficult implementation and practical issues that arise from the transformations of the independent variable (*i.e.,* time) and the EOMs. Especially, propagating to a fixed time presents a major practical challenge for such methods, as time has become a dependent variable through the time transformation. Several solutions are proposed, thoroughly discussed, and analysed. The proposed solution are implemented for Dromo [10], a specific regularised propagation method. Dromo features

[*] PhD Candidate, Astrodynamics and Space Missions, Delft University of Technology, Kluyverweg 1, 2629 HS, Delft, The Netherlands. j.geul@tudelft.nl, +31 (0)15 27 85367.

[†] Assistant Professor, Astrodynamics and Space Missions, Delft University of Technology, Kluyverweg 1, 2629 HS, Delft, The Netherlands. e.mooij@tudelft.nl, +31 (0)15 27 89115.

[‡] Assistant Professor, Astrodynamics and Space Missions, Delft University of Technology, Kluyverweg 1, 2629 HS, Delft, The Netherlands. r.noomen@tudelft.nl, +31 (0)15 27 85377.

only eight EOMs (formulated as variations of parameters (VOP) elements) and a separate treatment of perturbations that can and cannot be derived from a potential. Dromo with the proposed adaptations is compared against two (standard) propagation methods, namely Cowell's method and Modified Equinoctial Elements (MEEs) [12]. The methods are compared for a number of representative test cases.

The aim of the paper is to analyse how regularised methods compare to conventional methods, when their practical limitations have been addressed, and to identify uses for regularised methods. The focus is mainly on efficiency, defined as computational cost over accuracy.

First, regularisation and the formulation of Dromo are first explained. Second, the methodology of the proposed solutions is given. Thirds, the experimental set-up is explained, including the test approach, problems, and cases. Fourth, the results are presented and discussed. Finally, the main conclusions and recommendations are given.

BACKGROUND

The theory of regularised propagation is introduced, after which the theory of Dromo given.

Regularised propagation

Regularisation aims to remove singularities and dynamical instability from the EOMs, leading to less integration errors and more efficient integration. It commonly follows a two-step approach. First, a transformation of the independent variable is introduced. The general form of this transformation is known as the Sundman transformation, as given by Eq. (1). The relationship between the fictitious time s and physical time t is chosen such that for a constant steps size $h = \Delta s$, $t \to 0$ as $r^m \to 0$. Considering a highly eccentric orbit, the step size will become very small at pericentre and very large at apocentre. Analytic step-size regulation is achieved when the truncation becomes uniform at each time step. Second, the EOMs are transformed into a more suitable set of parameters for describing orbital motion, further improving the numerical properties [13].

$$\mathrm{d}s = A\frac{\mathrm{d}t}{r^m} \tag{1}$$

where A is a scaling factor and m the order of the time transformation. The optimal combination of m and A will depend on the type of orbit, integrator, and perturbations.

Additionally, the regularised EOMs can be reformulated through the theory of VOP. The transformed parameters, referred to as (orbital) elements, are chosen such that they are constant for idealised motion (e.g., two body). As the elements are constant (or linear) for idealised motion, only the perturbations have to be integrated.

Dromo and comparison of regularised methods

Many different regularised methods exist. The most popular classical methods are Kustaanheimo-Stiefel (KS) [14] and Sperling-Burdet (SB) [15]. Recently, there has been development towards leaner methods that take advantage of quaternions [4–11]. In particular, Dromo seems very interesting and is a VOP method derived from the Burdet-Ferrándiz [16] regularization. Dromo employs only eight EOMs, which is close to non-regularised methods (often six) and less than other regularised methods: Sperling-Burdett and Kustaanheimo-Stiefel feature 12 and 10 differential equations, respectively [17]. Other important advantages include: non-degenerate quaternions, perturbing accelerations in the orbital frame, unique formulation for all perturbed conics, and a seperate treatment of perturbing forces which are partially or fully derivable from a potential [10].

A number of classical and modern methods are compared in Table 1. Stiefel-Scheifele (SS) represents the benchmark for this problem. The table reports the final position (x, y, and z) after 288.127 689 41 mean solar days (roughly 50 revolutions), the number of steps per revolution (steps/rev), and the absolute error with respect to the benchmark. Additional results for Dromo, MEEs, and Cowell were obtained. The unified state model (USM) [18, 19] uses only seven ODEs (which can be reduced to six) based on the velocity hodograph and quaternions.

Table 1: Comparison of several methods for the oblate Earth plus Moon after 288.12768941 days.

	SS [3]	SB [3]	KS [3]	Cowell[a] [3]
x [km]	−24 219.050	−24 218.818	−24 219.002	−24 182.152
y [km]	227 962.106	227 961.915	227 962.429	227 943.989
z [km]	129 753.442	129 753.343	129 753.822	129 744.270
Steps/rev	500	62	62	240
Error [km]		0.318	0.501	42.5

	Dromo	Cowell[b]	MEE	USM [19]
x [km]	−24 218.829	−24 256.391	−24 256.980	−24 219.049
y [km]	227 961.980	227 980.068	227 979.117	227 962.106
z [km]	129 753.414	129 762.509	129 761.798	129 753.442
Steps/rev	113	453	188	372
Error [km]	0.256	42.4	42.4	42.1

[a] Note that Cowell's method here indicates the formulation by Bond and Hanssen [20], which uses the total energy as an independent parameter.

[b] Note that Cowell's method here indicates the Cartesian formulation.

It can be seen that on the test problem, all regularised methods (SB, KS, and Dromo) perform much better than the others. This is not at all surprising as the orbit is highly eccentric. Dromo is found to be a bit more expensive than KS on this problem. This is confirmed by Baù and Bombardelli [11], who further show that Dromo dominates the other methods for problems with lower eccentricities ($e = 0, 0.3, 0.7$).

Dromo formulation

Dromo (derived from Greek, meaning: a running or race) was initially developed by Peláez et al. [8]. Baù et al. [10] extended the method for a seperate treatment of perturbing forces which are derivable from a potential. A time element was finally introduced by Baù and Bombardelli [11].

Dromo employs eight elements: three for orbital shape and motion in the orbital plane ζ_{1-3}, four to describe the orientation ζ_{4-7}, and one related to time ζ_0.

Note that all quantaties in Dromo are non-dimensional, by introducing units for time and length as n_0^{-1} and R_0, respectively. Here R_0 is the initial radius and n_0 the initial circular mean motion (i.e., $n_0 = n(a = R_0) = \sqrt{\mu/R_0^3}$).

The perturbations are expressed in the orbital frame. This is convenient for many perturbations (e.g., drag, thrust, etc.). Moreover, perturbations derivable and non-derivable from a potential are treated separately, allowing for propagation that is consistent with total energy. The total perturbing acceleration $\mathbf{f}$ is given by:

$$\mathbf{f} = \mathbf{P} + R = \mathbf{P} - \frac{\partial R(t,)}{\partial \mathbf{r}_{orb}} \tag{2}$$

where $\mathbf{P}$ is the sum of all non-potential accelerations, R the total perturbing potential, the gradient operator, and $\mathbf{r}_{orb}$ represents the orthonormal basis $\{\mathbf{i}, \mathbf{j}, \mathbf{k}\}$, given by $\{\frac{\mathbf{r}}{r}, \mathbf{k}\times\mathbf{i}, \frac{\mathbf{h}}{h}\}$, where the angular momentum is $\mathbf{h} = \mathbf{r} \times \mathbf{v}$. Similarly, the perturbing accelerations $\mathbf{f}$ and $\mathbf{P}$ are composed of the radial, transverse, and normal components.

Time transformation The time is transformed from physical time t to fictitious time φ, using a Sundmann time transformation of order two:

$$\frac{\mathrm{d}t}{\mathrm{d}\varphi} = \frac{r^2}{\tilde{h}} = \frac{1}{\zeta_3 s^2} \tag{3}$$

where $A = \tilde{h}$ and $m = 2$. ζ_3 and s will be defined in the next section. The pseudo-angular momentum $\tilde{h}$ is defined as

$$\tilde{h} = r\sqrt{2\left(\mathcal{E} + \frac{1}{r}\right) - \left(\frac{\mathrm{d}r}{\mathrm{d}t}\right)^2} \tag{4}$$

where $\mathcal{E}$ is the total orbital energy and the relation between $\tilde{h}$ and h is $\tilde{h} = \sqrt{h^2 + 2r^2 R}$.

The independent variable φ can be considered as a perturbed version of the true anomaly ν. and is, therefore, also referred to as the pseudo true-anomaly. φ varies linearly with ν with an additional angular drift γ, such that $\Delta\varphi = \Delta\nu + \gamma$. γ is defined as:

$$\gamma = \Delta\omega + \int_{\Omega_0}^{\Omega} \cos i \,\mathrm{d}\Omega + 2\int_{0}^{t} \frac{R}{\tilde{h} + h}\mathrm{d}t \tag{5}$$

The numerical solution can be further improved by removing time t also from the EOMs. Looking at Eq. (3) it can be seen that even for unperturbed motion the physical time does not necessarily vary linearly with respect to φ. Due to the time transformation, the problem has been partly moved to EOMs, giving rise to a number of complications. First, the integration of physical time leads to errors in approximating time itself. Second, large variations of the time derivative can present a bottleneck, driving the integration accuracy of the entire state. Lastly, the time accuracy directly influences the state accuracy when the state at a certain epoch is desired.

Baù and Bombardelli [11] present two different alternative time formulations: the linear and constant time elements. In total this gives three options for the expression of time. These options will from hereon be referred to as non-dimensional physical time t, linear time element ζ_{0_l}, and constant time element ζ_{0_c}.The relations between time t and the time elements ζ_{0_l} and ζ_{0_c} are given by:

$$\zeta_{0_l} = t - \frac{V_r}{2\mathcal{E}\zeta_3 s} - \frac{1}{\mathcal{E}\sqrt{-2\mathcal{E}}} \arctan\left(\frac{V_r}{s + \sqrt{-2\mathcal{E}}}\right) \tag{6}$$

$$\zeta_{0_c} = \zeta_{0_l} - a^{3/2}\varphi \tag{7}$$

where V_r is the radial velocity, s the pseudo-transverse velocity, and ζ_3 the inverse pseudo angular-momentum, to be introduced in the next section. The differential equations are:

$$\frac{\mathrm{d}\zeta_{0_l}}{\mathrm{d}\varphi} = a^{3/2}\left[1 + \frac{\mathrm{d}\mathcal{E}}{\mathrm{d}\varphi}\left(6a\arctan\left(\frac{V_r}{k_4 + k_3}\right) + k_1\right) + \left(\frac{f_r}{\zeta_3 s} - 2R\right)k_2\right] \tag{8}$$

$$\frac{\mathrm{d}\zeta_{0_c}}{\mathrm{d}\varphi} = a^{3/2}\left[\frac{\mathrm{d}\mathcal{E}}{\mathrm{d}\varphi}\left(6a\arctan\left(\frac{V_r}{k_4 + k_3}\right) - 3a\varphi + k_1\right) + \left(\frac{f_r}{\zeta_3 s} - 2R\right)k_2\right] \tag{9}$$

where $\frac{\mathrm{d}\mathcal{E}}{\mathrm{d}\varphi}$ is the rate-of-change of the total energy and f_r the radial component of the perturbing acceleration. k_{1-4} are relations that simplify the notation and are defined to be:

$$k_1 = \frac{\sqrt{a}V_r}{s^2}\left(\frac{\zeta_3 + s}{k_4} + \frac{2k_3}{\zeta_3} + 1\right) \qquad k_2 = \frac{1}{s^2}\left(\frac{k_4}{\zeta_3} + \frac{k_3}{k_4} + \frac{V_r^2}{k_4 s}\right)$$

$$k_3 = \zeta_1\cos\varphi + \zeta_2\sin\varphi \qquad k_4 = \zeta_3 + \sqrt{-2\mathcal{E}}$$

Shape and orientation As mentioned before the set ζ_{1-7} can be split up into elements concerning the shape and dynamics ζ_{1-3}, and the orientation of the orbital plane ζ_{4-7}. The first three elements are obtained by introducing the generalised orbital element $\zeta_3 = \tilde{h}^{-1}$, from which ζ_1 and ζ_2 follow as integration constants. The remaining four constants of motion ζ_{4-7} describe the orientation and are closely related to quaternions.

The ODEs of the elements with respect to the independent parameter are

$$\frac{\mathrm{d}\zeta_1}{\mathrm{d}\varphi} = \frac{\sin\varphi}{s}\left(\frac{f_r}{\zeta_3 s} - 2R\right) - \left(\frac{s}{\zeta_3} + 1\right)\cos\varphi\frac{\mathrm{d}\zeta_3}{\mathrm{d}\varphi} \tag{10}$$

$$\frac{\mathrm{d}\zeta_2}{\mathrm{d}\varphi} = \frac{\cos\varphi}{s}\left(2R - \frac{f_r}{\zeta_3 s}\right) - \left(\frac{s}{\zeta_3} + 1\right)\sin\varphi\frac{\mathrm{d}\zeta_3}{\mathrm{d}\varphi} \tag{11}$$

$$\frac{\mathrm{d}\zeta_3}{\mathrm{d}\varphi} = -\frac{1}{s^4}\left[V_r\zeta_3 s\left(2R - \frac{\zeta_3 s}{s+\zeta_3}\frac{\partial R}{\partial\zeta_3}\right) + V_t P_t + \frac{\partial R}{\partial t}\right] \tag{12}$$

$$\frac{\mathrm{d}\zeta_4}{\mathrm{d}\varphi} = \frac{1}{s}\left[\frac{f_n}{\zeta_3 s V_t}(\zeta_7\cos\mathit{\Delta}\varphi - \zeta_6\sin\mathit{\Delta}\varphi) + \zeta_5(V_t - s)\right] \tag{13}$$

$$\frac{\mathrm{d}\zeta_5}{\mathrm{d}\varphi} = \frac{1}{s}\left[\frac{f_n}{\zeta_3 s V_t}(\zeta_6\cos\mathit{\Delta}\varphi + \zeta_7\sin\mathit{\Delta}\varphi) - \zeta_4(V_t - s)\right] \tag{14}$$

$$\frac{\mathrm{d}\zeta_6}{\mathrm{d}\varphi} = -\frac{1}{s}\left[\frac{f_n}{\zeta_3 s V_t}(\zeta_5\cos\mathit{\Delta}\varphi - \zeta_4\sin\mathit{\Delta}\varphi) - \zeta_7(V_t - s)\right] \tag{15}$$

$$\frac{\mathrm{d}\zeta_7}{\mathrm{d}\varphi} = -\frac{1}{s}\left[\frac{f_n}{\zeta_3 s V_t}(\zeta_4\cos\mathit{\Delta}\varphi + \zeta_5\sin\mathit{\Delta}\varphi) + \zeta_6(V_t - s)\right] \tag{16}$$

where the pseudo-transverse s, transverse V_t, and radial velocity V_r are given by

$$s = \zeta_3 + \zeta_1\cos\varphi + \zeta_2\sin\varphi \tag{17}$$

$$V_t = \sqrt{s^2 - 2R} \tag{18}$$

$$V_r = \zeta_1\sin\varphi - \zeta_2\cos\varphi \tag{19}$$

It is important to note that $\mathit{\Delta}\varphi$ in Eqs.(13-16) represents the difference between the current and initial value $\mathit{\Delta}\varphi = \varphi - \varphi_0$, and not the step size. The initial value of φ_0 can be arbitrarily chosen when the initial Dromo state is constructed. A common choice for φ_0 is the initial true anomaly $\varphi_0 = \nu_0$ or just $\varphi_0 = 0$.

Alternatively, when the perturbations are mainly conservative, the third element ζ_3 can be replaced by the total energy $\mathcal{E}$, replacing Eq. (12) by

$$\frac{\mathrm{d}\mathcal{E}}{\mathrm{d}\varphi} = \frac{1}{\zeta_3 s^2}\left(uP_r + P_t\sqrt{s^2 - 2R} + \frac{\partial R}{\partial t}\right) \tag{20}$$

where ζ_3 can then be obtained through the following equation:

$$\zeta_3 = \sqrt{\zeta_1^2 + \zeta_2^2 - 2\mathcal{E}} \tag{21}$$

In conclusion, only eight first-order ODEs need to be numerically integrated with Dromo. Different options for a set of eight equations have been present, these are: one related to time (Eq. (3), (8), or (9)), three related to the shape of the orbit (Eqs. (10), (11), and (12) or (20)) and four related to the orientation of the orbital plane (Eqs. (13) to (16)). The Dromo state thus consists of the following eight elements

$$= [t \text{ or } \zeta_{0_l} \text{ or } \zeta_{0_c},\ \zeta_1,\ \zeta_2,\ \zeta_3 \text{ or } \mathcal{E},\ \zeta_4,\ \zeta_5,\ \zeta_6,\ \zeta_7]$$

By default (unless stated otherwise) the following set is used: $= [\zeta_{0_c}, \zeta_1, \zeta_2, \mathcal{E}, \zeta_4, \zeta_5, \zeta_6, \zeta_7]$.

METHODOLOGY

The formulation of Dromo presented has a number of practical limitations. These limitations are generally true for any regularised method. To facilitate the implementation of these methods, the issue of propagating to a final time is considered. Although this treatment considers time, the methodology presented can in fact be used for finding the final state at any value of one of the dependent parameters, such as the orbital altitude (in case of re-entry) or any of the instantaneous orbital elements.

Fixed time propagation

Although the time transformation has many advantages, the major drawback is that time has become a dependent parameter and the relationship between both is non-linear. Therefore, propagating to a certain epoch is not straightforward. The final value of the independent parameter φ_f corresponding with the final physical time t_f can only be obtained numerically. This presents a problem for applications that require the state at an exact time, e.g., orbit determination.

The approximations of the final independent parameter and dependent time are in this section denoted as $\hat{\varphi}_f$ and $\hat{t}_f$.

The procedure of propagating to a final time has a number of drawbacks. First of all, finding the corresponding $\hat{\varphi}_f$ can be computationally expensive, as will be demonstrated. Secondly, since $\hat{\varphi}_f$ is only approximated and thus differs from the true unknown value, also the integrated $\hat{t}_f$ will not be exactly the desired t_f. Lastly, a small variation (or error) in $\hat{\varphi}_f$ can lead to a very large deviation in $\hat{t}_f$. This sensitivity is illustrated in Figure 1, where a change of about $0.35\,\mathrm{rad}$ in φ leads to a variation in t of almost $250\,000\,\mathrm{s}$. The error in time translates into additional state errors. It is therefore important to find a very good approximation in a computationally lean manner.

Three different solutions to this problem exist, which will be explained in more detail. Each method will be treated based on their implications on function evaluations, accuracy, and ease of implementation. To gain better insight in the differences in accuracy, the error in the final time can be split up as follows:

$$\varepsilon = |t_f - \hat{t}_f| = \varepsilon_{numerical} + \varepsilon_{approximation} + \varepsilon_{estimation} \tag{22}$$

where $\varepsilon_{numerical}$ is the numerical error due to the numerical integration (containing both round-off and truncation errors) and is dependent on the integration scheme. $\varepsilon_{approximation}$ is the error due to approximation of $\hat{\varphi}_f$ by the different methods, such that $|t(\hat{\varphi}_f) - t(\varphi_f)| = \varepsilon_{approximation}$. This error stems from convergence tolerance (if iterations are required). It has to be noted that this tolerance should be set individually from the integrator tolerances. Finally, $\varepsilon_{estimation}$ is an additional error introduced by the methods in finding $\hat{\varphi}_f$. The last source is not immediately intuitive, but will become clear. The error due to estimation is defined as:

$$\varepsilon_{estimation} = |\hat{t}_f(\text{integrator} + \text{estimator}(t_f)) - \hat{t}_f(\text{integrator}(\hat{\varphi}_f))| \tag{23}$$

where the first $\hat{t}_f$ is the result of the integration and estimation procedure combined and the second $\hat{t}_f$ is when the integration is repeated (without the estimation procedure) using the estimate of the independent parameter $\hat{\varphi}_f$. The difference between these two is the impact of the estimation.

Householder Householder methods are a class of iterative root-finding algorithms. Newton's method (sometimes also Newton-Raphson) and Halley's method are first- and second-order Householder members, respectively. Newton only uses the function gradient, while Halley uses additional second-order derivative information. Halley's method is considered to be more stable and generally faster, as it has a cubic upper bound for the rate of convergence, as opposed to quadratic for Newton's method. The scheme is initialised with an initial guess $\hat{\varphi}_{f_0}$, which is updated after each iteration until converged to within a set tolerance. The iterative schemes for Newton's and Halley's method are given by:

$$\hat{\varphi}_{f_{n+1}} = \hat{\varphi}_{f_n} - \frac{f}{f'} \qquad\qquad \hat{\varphi}_{f_{n+1}} = \hat{\varphi}_{f_n} - \frac{2ff'}{2f'^2 - ff''} \tag{24, 25}$$

where the derivatives are found using Eqs.(3) and (17):

$$f = t_n - t_f \tag{26}$$

$$f' = \frac{\mathrm{d}f}{\mathrm{d}\hat{\varphi}} = \frac{1}{\zeta_3 s^2} \tag{27}$$

$$f'' = \frac{\mathrm{d}^2 f}{\mathrm{d}\hat{\varphi}^2} = \frac{-2(-\zeta_1 \sin\hat{\varphi}_{f_n} + \zeta_2 \cos\hat{\varphi}_{f_n})}{\zeta_3 s^3} \tag{28}$$

This approach is both very accurate and expensive. As the integration is restarted for each new guess $\hat{\varphi}_f$, the error due to estimation is not present ($\varepsilon_{estimation} = 0$). However, the number of function evaluations can be high: roughly equal to the number of iterations times number of evaluations necessary to propagate a single trajectory. The number of iterations depends on the quality of the initial guess and the required level of convergence. Also, convergence is not guaranteed.

Bisection The second method uses bisection to successively approximate the φ_f. This is a simple root-finding method that repeatedly bisects an interval. As the maximum bisection interval is limited to the maximum step size, the state is first propagated until it overshoots the target t_f. The step is then reverted to the last point corresponding to φ_n, such that $t_n < t_f$. The step size is subsequently halved and the procedure repeated.

The approach is very simple to implement. The convergence is linear, which is slower than the other methods. However, the procedure requires no derivatives, initial guesses, and restarts. Therefore, requiring only a moderate number of additional function evaluations. However, the accuracy is only moderate, as additional numerical errors are introduced in the process, through the increasingly smaller steps.

Hermite interpolation The third and final method solves φ_f using interpolation. Out of the many different choices that exist (e.g., Lagrange polynomials, splines, etc.), a Hermite polynomial is chosen. Hermite polynomials (as opposed to Lagrange/Newton polynomials) use higher-order derivative information. Here, only the states and its first-order derivatives are used. Since this method is very akin to an extrapolation polynomial used by numerical integrators, the error made will be of the same order as the integration error, making it suitable for combination with numerical integration.

The first integration step after overshoot $t_n > t_f$ is denoted φ_n. Along with the state at φ_n, a number of additional points before and after are used. The interpolation of time is illustrated in Figure 1.

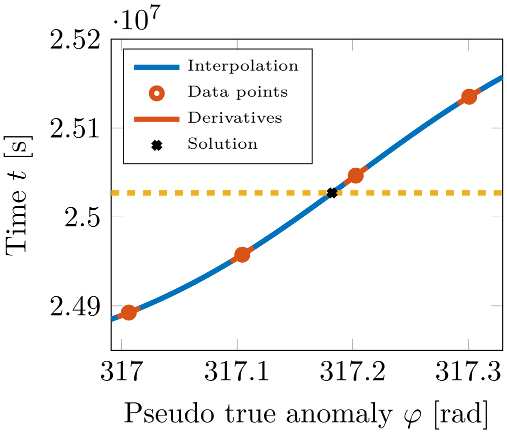

Figure 1: Hermite interpolation of the final time after 49.5 revolutions for problem 1.

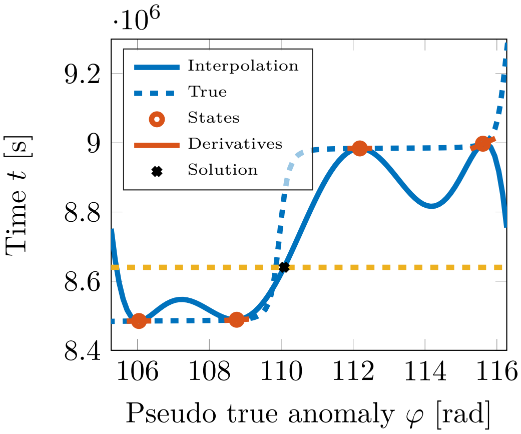

Figure 2: Improper interpolation of the final time due to too large step-size using time elements.

The values of the independent and dependent parameter are reduced according to Eqs.(29) and (30). For the independent parameter, it is essential for a good fit that the values are around zero. Moreover, the final time is subtracted, such that the polynomial $p(\bar{\varphi}_f) = 0$ needs to be solved.

$$\bar{\varphi}_i = \varphi_i - \varphi_n \qquad\qquad \bar{t}_i = t_i - t_f \qquad (29,\ 30)$$

The polynomial can then be generated using the set of points $\{\bar{t}_i,\ \dot{t},\ \bar{\varphi}_i\}$. The usual way would be to (numerically) compute the divided-difference table, from which the polynomial can then easily be found. However, it is proposed to solve the expression for the polynomial coefficients analytically as a function of the data points, such that $\mathbf{c} = f(,\bar{\mathbf{t}},\dot{\mathbf{t}})$, making it more efficient to set-up the polynomial. Note that one such set of expression for the coefficients needs to be found for each integration order. The polynomial then becomes:

$$\bar{t} = p(\bar{\varphi}; \mathbf{c}) = \sum_{i=1}^{N} c_i \bar{\varphi}^{i-1} \qquad (31)$$

The roots of the polynomial are then solved to find potential candidates for $\bar{\varphi}_f$. For polynomials of up to order four, closed-form analytic solutions exist. For higher orders, the roots must be solved numerically, which are found as the eigenvalues of the polynomial's companion matrix, defined as:

$$C(\mathbf{c}) = \begin{bmatrix} 0 & 0 & \cdots & 0 & -c_1 \\ 1 & 0 & \cdots & 0 & -c_2 \\ 0 & 1 & \cdots & 0 & -c_3 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & -c_N \end{bmatrix} \tag{32}$$

The number of roots (real and imaginary, including multiplicity) is equal to the polynomial order. To find the correct root, knowledge about the solution is applied to eliminate candidates. The following conditions are used to identify the proper root:

$$\bar{\varphi}_f \in \mathbb{R} \qquad\qquad \bar{\varphi}_{n-1} < \bar{\varphi}_f < 0 \qquad (33,\ 34)$$

The final independent variable is then $\hat{\varphi}_f = \varphi_n + \bar{\varphi}_f$.

The method is both accurate and efficient. The accuracy of the solution is driven by the fit of the polynomial alone, as the solving of the eigenvalues is exact (only round-off). Similarly, the estimation error is of the same order as the numerical errors for a single step, thus $\varepsilon_{estimation} \approx 0$. The method uses even less evaluations than bisection. Lastly, no initial guess is required and the method can be applied for any parameter. The implementation, however, is perhaps the most involved of the three methods.

If the parameter is part of the state, the step-size controller of the integrator ensures that a good spacing of data points for the polynomial is obtained. However, this can not be guaranteed when the parameter is derived from the state (in case of a time element), resulting in improper interpolation of the final time. Figure 2 illustrates the problem of improper interpolation. Although the polynomial satisfies the state and derivative constraints, the solution for $\hat{\varphi}_f$ is off. In such a case, time (instead of a time element) can be integrated around the solution. Note that the error of the interpolation can be easily obtained by integrating from φ_{n-1} to the estimated $\hat{\varphi}_f$. Using this error, the step size can be modified and the quality interpolation checked in a trial-and-error approach.

EXPERIMENTAL SET-UP

The general set-up of the numerical simulation is explained below, followed by an introduction to the numerical integration techniques used. Finally the test problems are given. Dromo (with the proposed solution) is tested for the following cases:

Case A: Time-state options investigates the effects of using different options for time.

Case B: Fixed-time propagation compares the three solutions for propagating to a fixed end in terms of time and positional accuracy.

Case C: Force modelling analyses the difference between conservative and dissipative perturbations.

Case D: Numerical integrators investigates different numerical integration techniques.

Case E: Different orbits compares the effect of orbits for the test problems.

General set-up

The aim is to make the comparison between the regularised and non-regularised methods as realistic as possible. The following two important aspects ensure this.

First, Runge-Kutta Dormand-Prince 5(4) (DOPRI5) used for all methods by default, unless stated otherwise. This integrator features step-size control, even though regularised methods could propagate with high accuracy using a fixed step-size integrator, through their analytic step-size control. However, doing so for the non-regularised methods would result in very poor performance and an unfair comparison.

Second, all problems are integrated to a fixed final time, not a pre-computed final value of the independent parameter that corresponds to the final time. This distinction is important, as it was already discussed, that integration to a final time is not straightforward with regularised methods. Usually for benchmarking the final independent parameter is first estimated with very high precision and the state is then propagated to the final value of the independent parameter instead. This approach, however, does not accurately reflect the computational costs associated with finding the final value φ_f.

Error and reference position

For each of the problems the orbit is integrated to the final time to obtain the final state for various numerical tolerances between 10^{-6} and 10^{-14}. The final state is converted to the Cartesian position $\mathbf{r}$ and then compared against a reference position $\mathbf{r}$. This comparison highlights the computational effort required to reach a certain accuracy (and vica-versa). The error is computed as the minimum between the absolute and relative error:

$$\varepsilon = \min\left(\|\mathbf{r} - \mathbf{r}\|, \left\|\frac{\mathbf{r} - \mathbf{r}}{\max(|\mathbf{r}|, |\mathbf{r}|)}\right\|\right) \tag{35}$$

A reference solution is not available for most tests. Unless the motion is unperturbed, the solution has to be found numerically. For this, a solution is computed using Dromo with the householder approach. The tolerance chosen is at least one order lower than the minimum tolerance for the test.

Force modelling

In addition to two-body unperturbed motion, the effects of the Earth's oblateness and atmosphere are analysed. The models, introduced in more detail below, can be considered crude. However, they represent two fundamental ingredients, namely conservative potential-derived and dissipative perturbations. Therefore, they are representative for more advanced models and will suffice in highlighting differences among the methods.

Oblateness Variations in the Earth's gravity field can be expressed as both an perturbing acceleration $\mathbf{f}$ or potential R. Accounting only oblateness, the potential and acceleration are given by [13]:

$$R_{J_2} = \frac{J_2 \mu R_E^2}{2r^3} \left(1 - 3\sin^2 i \sin^2(\nu + \omega)\right) \tag{36}$$

$$\mathbf{f}_{J_2} = -R_{J_2} \tag{37}$$

To facilitate comparison with literature, the following constants were assumed [2]: $\mu = 398\,601\,\mathrm{km^3\,s^{-2}}$, $R_E = 6371.22\,\mathrm{km}$, and $J_2 = 1.082\,65 \times 10^{-3}$.

Drag acceleration Due to the presence of an atmosphere the satellite will experience drag, dissipating the satellite's orbital energy. A simple tabulated exponential density model, based on USSA1976 and CIRA1972, is assumed [21]. A spherical Earth is assumed and the co-rotation neglected. The drag acceleration is:

$$\mathbf{f}_{drag} = \frac{1}{2}\rho \mathbf{v}^\top \mathbf{v} \frac{SC_D}{m} \tag{38}$$

where the constants $C_D = 2.2$ and $S/m = 0.01\,\mathrm{m^2\,kg^{-1}}$ are assumed [10].

Test problems

The analysis is executed for several test problems. Table 2 gives an overview. By default Problem 1 is used for Cases A to D, for Case E all three problems are analysed. The initial positions are given in Table 3.

Table 2: Overview of the orbits of the different test problems.

#	a [km]	e [−]	i [°]	T [d]	Revs [−]	Note
1	136 000	0.95	30.0	100	17.43	"Example 2" [2]
2	6777	0.0012	51.4	2	31.12	ISS [*]
3	29 600	0.0005	55.2	20	34.57	GSAT0101 [†]

Table 3: Initial position and velocity of different test problems.

#	x_0 [km]	y_0 [km]	z_0 [km]	v_{x_0} [km s^{-1}]	v_{y_0} [km s^{-1}]	v_{z_0} [km s^{-1}]
1	0	−5888.9727	3400	10.691 338	0	0
2	−276.511	4783.577	4790.565	6.899 16	2.172 63	−2.554 63
3	−6836.512	2513.848	−28 685.790	−0.990 54	−3.533 21	−0.071 89

Problem 1 was originally introduced as "Example 2" by Stiefel and Scheifele [2] and is commonly used throughout literature [3, 8, 10]. Problem 2 is a near-circular low-earth orbit (LEO). The orbit of the ISS (orbit 925) is used as a reference. Of course, for the propagation in LEO a more accurate atmospheric model is necessary. Nonetheless, the

[*] `http://spaceflight.nasa.gov/realdata/sightings/SSapplications/Post/JavaSSOP/orbit/ISS/SVPOST.html`, accessed on 02/20/2015.

[†] `https://www.space-track.org/basicspacedata/query/class/tle_latest/ORDINAL/1/NORAD_CAT_ID/37846/orderby/TLE_LINE1ASC/format/tle`, accessed on 02/20/2015.

exponential model that was selected will show the effect of a dissipative force. Finally, Problem 3 uses the orbit of the first Galileo satellite (GSAT0101) as a reference. The TLE at epoch 15 051.142 848 34 is used for the initial conditions. As the satellite is in MEO, it will not experience the effects of the atmosphere or oblateness as much as a LEO satellite. Nonetheless, both the drag and J_2 perturbations are modelled.

Integrators

Save for unperturbed motion, the EOMs need to be integrated numerically. The choice of integrator also has a large influence on the accuracy and efficiency. Distinctly different types of integrators are analysed to assess their performance. These include a single-step (DOPRI), multi-step (Adams-Bashforth-Adams-Moulton or ABAM), and Taylor-Series integrators (TSI).

For each integrator, only the absolute and relative tolerances are changed to obtain different data points of the accuracy versus the number of function evaluations. The tolerances in each case are set equal for each parameter, such that $\varepsilon_{abs_i} = \varepsilon_{rel_i} = \varepsilon \forall i$.

Dormand-Prince 5(4) DOPRI is a member of the Runge-Kutta (RK) family and convenient for adaptive step-size. RK techniques approximate a Taylor-series extrapolation of a function by evaluating the first derivative at different points within the interval and do not rely on previous solutions. Unfortunately, the truncation error is not available as part of the solution and requires the evaluation of another RK method of one order less or higher [22]. DOPRI5(4) is a fifth-order integration method, where the fourth-order is used for error control. The method is implemented in Matlab as ode45, and is used as is.

Adams-Bashforth-Adams-Moulton Multi-step methods take advantage of the history previously evaluated steps. In case of predictor schemes the new step is predicted using only the history available. A corrector uses the derivative at this new point to further improve the solution. ABAM, uses the AB predictor and AM corrector. The difference between the predicted and corrected values offers a measure of the error at each step. Only two derivatives are evaluated for each step, independent of the order. However, these methods are more complex to program and require special starting methods [23]. The integrator is implemented in Matlab as ode113, with adaptive order and step size based on the theory of Shampine and Gordon [24].

Taylor Series Integration TSI uses higher-order derivatives to expand the solution around the known point. In theory, a solution can be represented by an infinite series. In practice, the series is truncated after a desirable order K. TSI is very efficient as large steps can be made. However, it is difficult to implement. Automatic derivation up to any order is achieved by rewriting the EOMs using recursion formulas. To derive the recursion formulas, additional state variables need to be introduced. Due to this overhead, TSI is only more efficient when a problem can be solved many times [25]. Complications arise from models that are discrete/discontinuous (e.g., look-up tables). Most models can only be made piecewise continuous. The derivatives are therefore only valid within a certain region, limiting the maximum step size to its boundary. The value of the independent parameter at the boundaries need to be found using a root-finding algorithm. Interpolation of models can either be done once (for simple models) or on a need basis. Derivatives of numerical (black-box) models can be obtained through finite differencing. **Scott2008** presents a practical introduction of the subject.

The order K and step size h need to be controlled to keep the truncation error in bounds, while maximising computational efficiency. An estimate of the truncation error is generally obtained from the higher-order terms. The following step-size controller, which computes the Lagrange remainder using the last two terms, is used [26]:

$$
\begin{aligned}
h_0 &= \left(\min_{\forall i} \left\{ \frac{\varepsilon_i}{|(x_i)_{K-1}|} \right\} \right)^{\frac{1}{K-1}} \\
h_1 &= \exp\left[\frac{1}{K-1} \ln \left(\min_{\forall i} \left\{ \frac{\varepsilon_i}{|(x_i)_{K-1}| + K|(x_i)_K| h_1} \right\} \right) \right]
\end{aligned}
\tag{39}
$$

The estimate for the step size can be iterated further (by obtaining h_2 by substituting $h_0 = h_1$ etc.), but only the first iteration is applied. The order is assumed to be fixed throughout the integration and is optimised for each tolerance.

Simplifications of the Atmospheric Modelling The simplify integrating with TSI, the following atmospheric model is assumed (for Case D only): $\rho = \exp(-0.003\,731\,343\,283\,582\,1h - 29.702\,507\,460\,603\,904\,1)$, giving a good fit for $h \geq 800\,\mathrm{km}$. The EOMs for the gravity with oblateness (J_2) are taken from Bergsma and Mooij [26], with the following additionally derived equations and recurrence relations:

$$f_4 = \mathbf{v}^\top \mathbf{v} \qquad (f_4)_k = \sum_{j=0}^{k} (\mathbf{v})_k^\top (\mathbf{v})_{k-j} \tag{40, 41}$$

$$v = \sqrt{f_4} \qquad (v)_k = \frac{1}{(f_4)_0} \sum_{j=1}^{k} \left(\frac{3j}{2k} - 1 \right) (f_4)_j (v)_{k-j} \tag{42, 43}$$

$$\rho = a \exp(-br) \qquad (\rho)_k = a(-b)^k \sum_{j=1}^{k} \frac{j(r)_j(\rho)_{k-j}}{k} \tag{44, 45}$$

$$f_5 = v\rho \qquad (f_5)_k = \sum_{j=0}^{k} (v)_j (\rho)_{k-j} \tag{46, 47}$$

$$\mathbf{f} = -\frac{1}{2} f_5 \mathbf{v} \frac{SC_D}{m} \qquad (\mathbf{f})_k = \frac{1}{2} \frac{SC_D}{m} \sum_{j=0}^{k} (f_5)_j (\mathbf{v})_{k-j} \tag{47, 48}$$

where f_4 and f_5 represent some auxiliary introduced variables and $a = \exp(bR_E - 29.702\,507\,460\,603\,904\,1)$ and $b = 0.003\,731\,343\,283\,582\,1$.

RESULTS AND DISCUSSION

Each of the cases is analysed and their results are presented and discussed.

Case A: Time-state options

The linear and constant time element have been presented. The numerical properties of the three options has been extensively studied by Baù and Bombardelli [11]. In general, it is found that the time elements outperform the physical time on all test problems. As expected, for circular orbits with only J_2, all three options perform roughly equal. Similarly, they converge with higher number of function evaluations. For these two cases the physical time can be considered, as the equations are much simpler.

The time elements were found to be most accurate (up to 3-6 orders in magnitude) for low number of function evaluations, where the accuracy of the time t is dominating the integration. The evolution of all three options over two full orbits on Problem 1 are shown in Figure 3. As can be seen from the plot the physical (non-dimensional) time exhibits a sigmoid-type curve for every revolution. For this specific problem, this behaviour is driven by variations in the radius r. However, looking at Eqs. (3) and (4), a similar trend occurs for variations of the orbital energy. As time is a strictly increasing function, it is by definition hard to approximate with a polynomial and therefore difficult to integrate. Only when the orbit is circular (and mainly conservatively perturbed) or the integration steps become small, a good fit can be obtained.

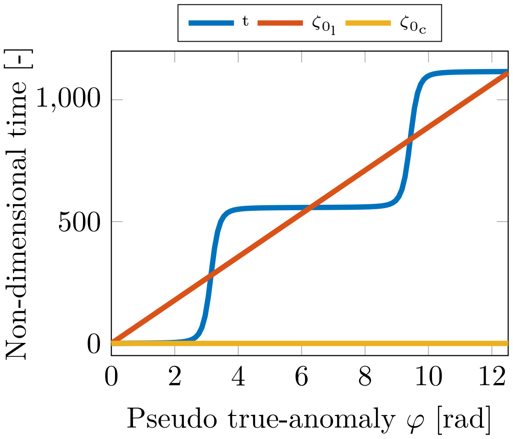

Figure 3: Evolution of different time elements (Case A) for two revolutions on problem 1.

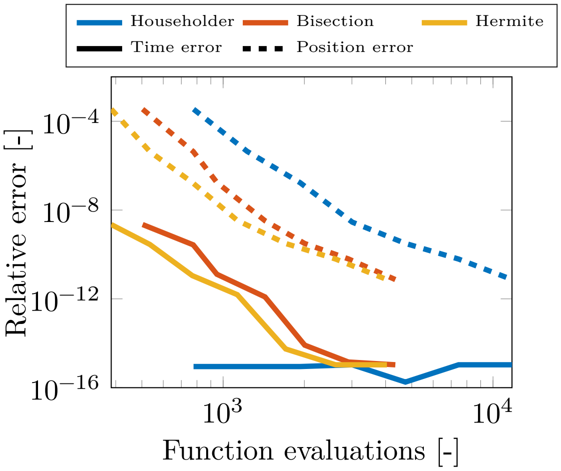

Figure 4: Error in final time as a function of the derivative evalutions for different approximation methods (Case B).

Case B: Fixed-time propagation

Three solutions for propagating to a fixed time have been proposed. Figure 4 shows the error made in approximating final time (solid lines) and in position (dashed lines) as a function of the number of function evaluations.

The ability of the householder method to estimate time correctly is constant, regardless of the number of function evaluations. This is to be expected considering the earlier discussion about how the iterative scheme decouples the error of estimation from the solution. The bisection and Hermite interpolating methods show only a small difference. Both approach the accuracy of the householder method as the number of function evaluations increases. As the step size goes down with increasingly stringent tolerances, the integration steps get smaller and thus closer to the value of t_f that is to be estimated, therefore reducing the impact of estimation.

Looking at the position error (dashed lines), the householder method actually performs worst, due to its inefficiency. The difference between the other two methods is again relatively small. However in terms of function evaluations the Hermite interpolation method is between 7 (for high accuracy) to 24 % (for low accuracy) more efficient than bisection.

Case C: Force modelling

Next, the effects of different perturbations are investigated. First, the unperturbed motion is analysed, followed by perturbed motion due to the Earth's oblateness (J_2), and finally the atmospheric drag is added. The results are shown in Figure 5.

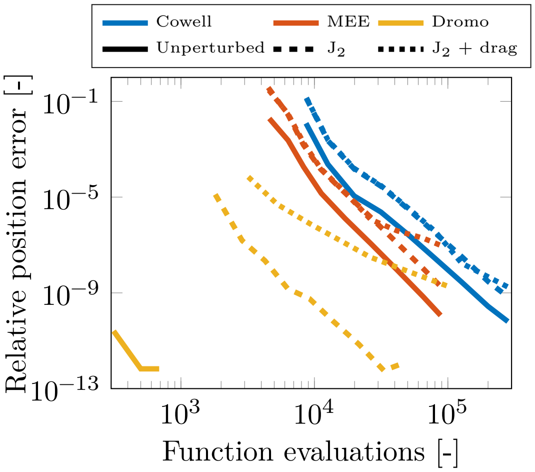

Figure 5: Comparison of different force models (Case C) on test problem 1.

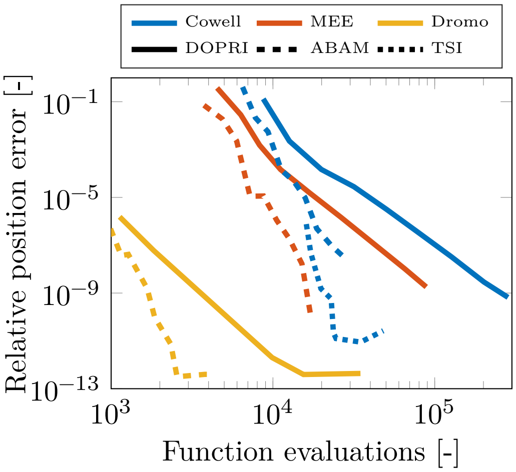

Figure 6: Comparison of different integration methods (Case D) on test problem 1.

For the unperturbed case (solid line), Dromo clearly performs best and the function evaluations and precision are hardly impacted by the tolerances settings. Ideally the precision would be perfect, as the elements are not impacted by the integration as their derivatives are zero and thus remain constant. The small error that is present and the associated number of function evaluations stem from the approximation of φ_f and any numerical errors during conversion from and to Cartesian coordinates.

It can further be seen that MEE is about an order more accurate compared to Cowell, for a given number of function evaluations. The errors originate from the integration of the true longitude alone (as the other five are constant).

For the second case (J_2, dashed line), again Dromo performs best. Compared to MEEs, Dromo has a better accuracy and is more efficient. It is remarkable that for a low number of function evaluations the other methods do not even offer a useful solution, Moreover, for Dromo the minimum error remains unaffected, though the number of function evaluations has increased. This shows the benefit of Dromo's approach to treating the perturbation as a conservative potential, in which the total energy does not change. The other two methods show a shift upwards, signifying a reduction in accuracy for a given number of function evaluations.

Lastly, for the case with both a conservative and a non-conservative perturbation (J_2 + drag, dotted line), all methods show a reduction in accuracy. Especially Dromo shows a significantly reduced accuracy. Nonetheless, the method still performs best, especially with a low number of function evaluations, where it is over two orders more accurate than MEE.

Case D: Numerical integrators

The fully-perturbed problem is analysed for three different variable step-size integrators: DOPRI5(4), ABAM, and TSI. Figure 6 shows the results. The differences among the integrators are most profound for the more stringent tolerance settings (higher evaluations). ABAM is found to be more efficient than DOPRI5(4) (or any other RK-family integrator) on this problem, especially for high-accuracy applications. For Dromo and MEE there is only an improvement in function evaluations. ABAM for Cowell's method shows better performance at a reduction in accuracy. This is likely due to the large variation of the states for this method, as ABAM relies on previous states.

Only results for TSI of Cowell's method were obtained. It can be seen that TSI performs significantly better than DOPRI and approaches ABAM for the lower number of function evaluations. Increasing the tolerance above 10^{-13} yields no additional benefits in terms of accuracy at the expense of evaluations. Three factors however have to be taken into account when viewing these results. First, the atmospheric model was further simplified for the comparison. If more advanced models are used, the step size has to be limited further. Secondly, the dimension of the Cartesian TSI state is 22 as opposed to 6 for the regular implementation, which has not been taken into account in the number of function evaluations. Lastly, finding the optimal order for TSI was found to be non-trivial. No order controller gave consistent results on this problem and was thus optimised for each run, giving:

$$\begin{aligned}\mathbf{K}^{\star}(&[10^{-7},\ 10^{-8},\ 10^{-9},\ 10^{-10},\ 10^{-11},\ 10^{-12},\ 10^{-13},\ 10^{-14},\ 10^{-15}])\\ =&[14\quad,\ 16\quad,\ 16\quad,\ 16\quad,\ 18\quad,\ 20\quad,\ 18\quad,\ 18\quad,\ 19\quad]\end{aligned}$$

Case E: Different orbits

The fully perturbed case of Problem 1, given by Figure 5, was already analysed. Problems 2 and 3 are also analysed. Their results are shown in Figures 7 and 8, respectively. Comparing Problem 2 to Problem 1, Dromo and MEE are much closer to each other. For the lower number of function evaluations, Dromo is about 1-2 orders more accurate than MEE, compared to 2-4 orders for the fully perturbed case of Problem 1. For a higher number of function evaluations two methods even converge and reach the same minimum error.

As can be seen from the figures, Dromo also performs best for these problems. Both Dromo and MEE achieve a maximum accuracy, after which higher tolerances do not result in increased accuracy. This is caused by the total round-off error becoming dominant over the decrease in the total truncation error. Dromo compares favourably in this respect against MEE, by achieving a lower minimum error of around one order less.

Both Problems 2 and 3, compared to 1, show Dromo and MEE closer to each other. From this it can be concluded that MEE performs relatively worse on highly elliptic orbits. The convergence of Dromo to MEE, which is present in Problem 2, but not in 3 indicates that this should be attributed to atmospheric drag.

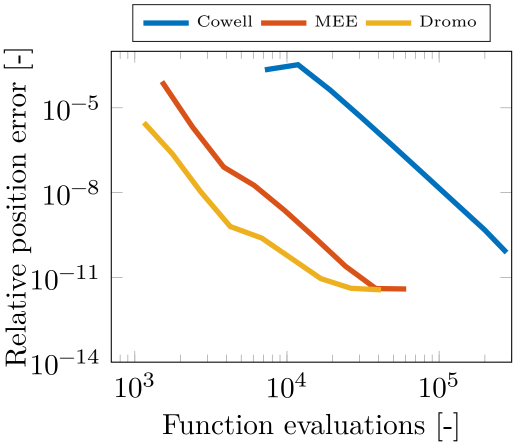

Figure 7: Comparison of methods on circular LEO (Case E, Problem 2).

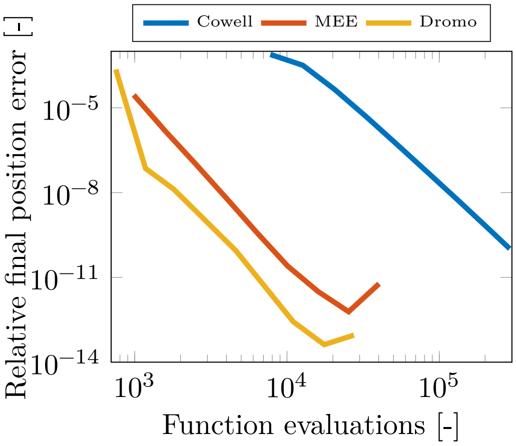

Figure 8: Comparison of methods on circular MEO (Case E, Problem 3).

CONCLUSIONS

The time elements have very interesting properties and prove a useful addition to the Dromo formulation, especially for low-accuracy computations on problems where the orbital radius and/or energy vary.

Two methods for propagating to a fixed epoch have been proposed, namely bisection and Hermite interpolation. Both methods work very well. In terms of accuracy they approach the classical Householder method for stringent tolerance settings. Although their absolute accuracy lacks behind the iterative scheme, overall their performance in terms of positional accuracy over function evaluations is better in all cases. The differences between bisection and Hermite interpolation are largest in the low-accuracy region, where Hermite interpolation is up to 25 % more efficient in terms of function evaluations.

Dromo dominates the other methods for all cases and problems, especially in terms of computational effeciency. Cowell's method performed worst on all problems. Dromo and MEE performed roughly equal for a small portion of the regime with high accuracy and number of function evalutions on Problem 2 (LEO with drag). Dromo is found to be very efficient, finding relatively accurate solutions for a limited number of evaluations. Similarly, Dromo is able to propagate more accurately than other methods, demonstrating a higher maximum accuracy, at a fraction of the computational cost of the other methods.

The choice of numerical integrators has an effect on performance. ABAM works very well with VOP formulations. TSI outperforms ABAM and DOPRI5(4) for Cowell's method. However, a number of practical issues have been presented that have to be addressed, especially expansion for discontinuous models and order control. TSI for Dromo should provide an additional gain in performance; it remains to be seen how much additional overhead is introduced. The number of additional state variables that needs to be introduced is large, due to Dromo's more complex ODEs.

In conclusion, it is demonstrated that Dromo, through the proposed adaptations, can be successfully applied to real orbit problems, providing significant benefits in both accuracy and number of functions evaluations. These findings should hold true for regularised propagation methods in general. The advantage in terms of computational efficiency is most significant for low- to medium-accuracy applications. For the propagation of LEO satellites, Dromo offers an improvement in accuracy (for a given number of function evaluations) of up to seven orders in magnitude, compared to the standard Cowell's method using Cartesian coordinates. The proposed method is recommended for all orbit problems that have been analysed. Especially great potential is foreseen for highly demanding applications, such as the propagation of a large catalog of objects, where currently (semi-)analytic techniques are employed.

ACKNOWLEDGEMENT

This work is supported by the European Office for Aerospace Research and Development (EOARD), grant FA9550-14-1-0344.

REFERENCES

[1] F. R. Hoots and R. G. France, "The Future of Artificial Satellite Theories", *Celestial Mechanics and Dynamical Astronomy*, Vol. 66, No. 1, 1997, pp. 51–60.

[2] E. Stiefel and G. Scheifele, *Linear and Regular Celestial Mechanics*, Springer-Verlag, 1971.

[3] V. R. Bond and M. C. Allman, *Modern Astrodynamics*, Princeton University Press, 1996.

[4] J. Waldvogel, "Quaternions and the perturbed Kepler problem", *Celestial Mechanics and Dynamical Astronomy*, Vol. 95, No. 1-4, Aug. 2006, pp. 201–212.

[5] T. Fukushima, "New Two-body Regularization", *The Astronomical Journal*, Vol. 133, No. 1, 2007, pp. 1–10.

[6] T. Fukushima, "Numerical Comparison of Two-body Regularizations", *The Astronomical Journal*, Vol. 133, No. 6, 2007, pp. 2815–2824.

[7] J. Waldvogel, "Fundamentals of Regularization in Celestial Mechanics and Linear Perturbation Theories", *Seminar for Applied Mathematics*, 2007.

[8] J. Peláez, J. M. Hedo, and P. R. de Andrés, "A Special Perturbation Method in Orbital Dynamics", *Celestial Mechanics and Dynamical Astronomy*, Vol. 97, No. 2, 2007, pp. 131–150.

[9] J. Waldvogel, "Quaternions for Regularizing Celestial Mechanics – the Right Way", *Celestial Mechanics and Dynamical Astronomy*, Vol. 102 2008, pp. 149–162.

[10] G. Baù, C. Bombardelli, and J. Peláez, "A New Set of Integrals of Motion to Propagate the Perturbed Two-body Problem", *Celestial Mechanics and Dynamical Astronomy*, Vol. 116, No. 1, 2013, pp. 53–78.

[11] G. Baù and C. Bombardelli, "Time Elements for Enhanced Performance of the Dromo Orbit Propagator", *The Astronomical Journal*, Vol. 148, No. 3, 2014, pp. 43–58.

[12] R. A. Broucke and P. J. Cefola, "On the Equinoctial Orbit Elements", *Celestial Mechanics*, Vol. 5, No. 3, 1972, pp. 303–310.

[13] K. F. Wakker, *Fundamentals of Astrodynamics*, Institutional Repository Delft University of Technology, 2015.

[14] P. Kustaanheimo and E. Stiefel, “Perturbation Theory of Kepler Motion based on Spinor Regularization”, *Journal für die Reine und Angewandte Mathematik*, Vol. 1965, No. 218, 1965, pp. 204–219.

[15] C. A. Burdet, “Regularization of the Two Body Problem”, *Zeitschrift für angewandte Mathematik und Physik*, Vol. 18, No. 3, 1967, pp. 434–438.

[16] J. Ferràndiz, “A General Canonical Transformaiton Increasing the Number of Variables with Application to the Two-body Problem”, *Celestial mechanics*, Vol. 41, No. 1968, 1988, pp. 343–357.

[17] D. J. Jezewski, “A Comparative Study of Netwonian, Kustaanheim/Stiefel, and Sperling/Burdet Optimal Trajectories”, *Celestial Mechanics*, Vol. 12, No. 3, 1975, pp. 297–315.

[18] S. P. Altman, “A Unified State Model of Orbital Trajectory and Attitude Dynamics”, *Celestial mechanics*, Vol. 6, No. 4, 1972, pp. 425–446.

[19] V. Vittaldev, E. Mooij, and M. C. Naeije, “Unified State Model theory and application in Astrodynamics”, *Celestial Mechanics and Dynamical Astronomy*, Vol. 112, No. 3, 2012, pp. 253–282.

[20] V. R. Bond and V. Hanssen, “The Burdet Formulation of the Perturbed Two-body Problem with Total Energy as an Element”, *NASA-JSC-Internal Note*, No. 73-FM-86 (JSC-O8004), 1973.

[21] D. A. Vallado, *Fundamentals of Astrodynamics and Applications*, Third Ed, Microcosm Press, 2007.

[22] M. Allione, A. Blackford, J.C.Mendez, and M.M.Wittouck, “The N-Body Problem and Special Perturbation Techniques”, *Guidance, Flight Mechanics and Trajectory Optimization*, Volume VI, National Aeronautics and Space Administration, 1968.

[23] R. Hamming, *Numerical methods for scientists and engineers*, Dover Publishers, 1987.

[24] L. F. Shampine and M. K. Gordon, “Local Error and Variable Order Adams Codes”, *Applied Mathematics and Computation*, Vol. 1, No. 1, 1975, pp. 47–66.

[25] G. Corliss and Y. F. Chang, “Solving Ordinary Differential Equations Using Taylor Series”, *ACM Transactions on Mathematical Software*, Vol. 8, No. 2, 1982, pp. 114–144.

[26] M. C. W. Bergsma and E. Mooij, “Application of Taylor Series Integration to Reentry Problems”, *AIAA Science and Technology Forum and Exposition*, Unpublished, 2016.