AAS 14-230

APPLICATION OF THE STARK PROBLEM TO SPACE TRAJECTORIES WITH TIME-VARYING PERTURBATIONS

Noble Hatten,* Ryan P. Russell†

Three methods of obtaining solutions to the Stark problem – one using Jacobi elliptic and related functions, one using Weierstrass elliptic and related functions, and one using F and G Taylor series extended to the Stark problem – are compared qualitatively and quantitatively. In the current implementations, the Jacobi formulation is found to be more computationally efficient than the Weierstrass formulation, while the series formulation provides the most efficiency if the order is not excessively high. The effectiveness of the Stark model in approximating space trajectories with continuously time-varying perturbations is compared to the ubiquitous Sims-Flanagan model. A method based on Adams-Bashforth numerical integration formulae is applied to the determination of disturbing acceleration vectors for each model. The method uses disturbing acceleration information from previous propagation steps to calculate a higher-order approximation of the effects of the perturbation between two steps. The increase in computation time compared to a first-order approximation is minimal because no additional propagations are required. Results for two example low-Earth orbits perturbed by J_2 and atmospheric drag, respectively, are presented.

INTRODUCTION

When propagating spacecraft trajectories, a balance must always be struck between speed and accuracy. For low-to-medium-fidelity applications like mission-planning trade studies, the ability to rapidly approximate many trajectories often takes precedence. A commonly used method to improve propagation speed is to use analytical solutions rather than resort to classic numerical integration. Due to its computational efficiency, Kepler propagation is frequently used as an approximation in situations when perturbations are small and accuracy is not the highest priority. Notably, the so-called Sims-Flanagan model is a reduced model that uses pure Keplerian arcs connected at nodes with impulsive velocity vector discontinuities ($\Delta \boldsymbol{v}$'s).[1] This technique has been applied successfully to produce approximations of low-thrust trajectories in software such as MALTO[2] and GALLOP.[3]

A lesser-known model of astrodynamical significance that also admits an analytical solution is the Stark problem, which describes a particle subject to both an inverse-square force field and a second force that is inertially constant in both magnitude and direction.[4] Though known to be integrable since the work of Lagrange,[5] work to express the solution of the Stark problem in various forms

*Graduate Student, Department of Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin, W. R. Woolrich Laboratories. 1 University Station, C0600, 210 E 24th St., Austin, TX 78712; Student member AIAA; noble.hatten@utexas.edu

†Assistant Professor, Department of Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin, W. R. Woolrich Laboratories. 1 University Station, C0600, 210 E 24th St., Austin, TX 78712; Senior member AIAA; ryan.russell@austin.utexas.edu

has continued to this day.[6,7,8,9,10,11,12] Of particular interest, the last few years alone have seen the presentation of the solution in terms of Jacobi elliptic and related functions,[6] Weierstrass elliptic and related functions,[12] and Taylor series expansions.[11] In addition to the qualitative characterization of the system, the solution methods may be applied to quantitatively propagate the trajectory of a spacecraft. Indeed, the Stark problem provides an exact model for the case of a spacecraft in a two-body gravitational field subjected to piecewise-constant thrust acceleration. The model has been applied successfully to low-thrust trajectory optimization problems with observed computation speed improvements compared to classic numerical integration.[13,14,11]

This study has two purposes. The first is to qualitatively and quantitatively compare the recent Jacobi, Weierstrass, and Taylor series formulations of the Stark problem for the so-called bounded case, when the motion of the particle remains bound to the central body for all time.* Within this class of motion, the Stark framework is useful for modeling a variety of disturbing accelerations such as natural perturbations to geocentric satellites and low-thrust maneuvers. The second purpose is to determine the applicability and efficiency of these Stark models, as well as the Sims-Flanagan model, as approximations to a spacecraft subjected to disturbances that vary continuously in time. A method of determining the appropriate $\Delta\boldsymbol{v}$'s for the Sims-Flanagan model and the disturbing accelerations for the Stark model adapted from linear multistep numerical integration algorithms is presented.

The three Stark solution methods are described, and the speed and accuracy of each method in the computation of a single trajectory segment are analyzed. The Sims-Flanagan model is then briefly summarized, and a procedure for using the Adams-Bashforth method in conjunction with both the Sims-Flanagan and Stark models to approximate a trajectory subject to time-varying perturbations is presented. Results are compared to classic numerical integration. The differences between the three models are shown in Figure 1.

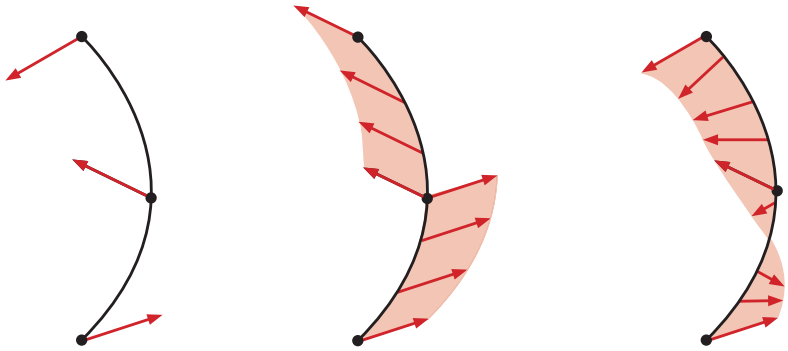


Figure 1: From left to right: The Sims-Flanagan model, the Stark model, and the numerically integrated model. For the Sims-Flanagan model, the arrows represent $\Delta\boldsymbol{v}$'s; for the Stark and numerically integrated models, the arrows represent continuous disturbing acceleration.

*Given the initial state of the particle, the gravitational parameter of the central body, and the disturbing acceleration vector, the determination of boundedness may be obtained analytically from either the Jacobi or Weierstrass solution methods.[6,12]

ANALYTICAL SOLUTION OF THE STARK PROBLEM IN TERMS OF JACOBI ELLIPTIC FUNCTIONS

The analytical solution of the Stark problem in terms of Jacobi elliptic functions has been presented in varying degrees of completeness by several authors.[8,7,6] Lantoine and Russell provide the most complete derivations and solutions for all possible classes of motion, and the discussion and implementation presented here is based on their presentation.

The Stark model may be described without loss of generality by

$$\ddot{\boldsymbol{r}} = -\frac{\mu}{r^3}\boldsymbol{r} + \epsilon\hat{\boldsymbol{k}}, \tag{1}$$

where $\boldsymbol{r}$ is the particle's position vector; $r = ||\boldsymbol{r}||$; μ is the gravitational parameter of the central body; ϵ is the magnitude of the disturbing acceleration (of arbitrary magnitude); and $\hat{\boldsymbol{k}}$ is a unit vector in the $+z$ direction. The problem is solved by transforming the state into parabolic coordinates. In the parabolic system, the Hamiltonian is separable, and explicit integration of the equations of motion yields solutions for the parabolic coordinates in terms of Jacobi elliptic functions and elliptic integrals. The solution process is complicated by the existence of multiple formulae for different cases. The correct formula to use for a given situation is determined by the initial state of the particle, μ, and ϵ. The first determination that must be made is whether the disturbing acceleration vector lies in the particle's plane of motion. If so, the problem is planar or two-dimensional, and there are seven possible solution types. For the three-dimensional problem, there are three solution types, each of which is obtained via a simple transformation from a corresponding 2D formula. Importantly, due to the structure of the 3D parabolic coordinate system, the 3D solutions do not reduce to the 2D formulae when given 2D initial conditions. Thus, both 2D and 3D solutions are needed for a complete implementation.

The required workload may be reduced significantly if it is known in advance that the orbit is bounded. In this case, only two of the seven 2D solutions and one of the three 3D solutions are relevant. As many natural perturbations and low-thrust propulsive accelerations result in bounded trajectories, boundedness is often not an unreasonable assumption for astrodynamics applications, and the complexity of the solution process may be reduced accordingly.

In addition to the spatial transformation into parabolic coordinates, a transformation of the independent variable is also used. For both the 2D and 3D cases, the independent variable is τ, defined such that

$$\mathrm{d}t = 2r\mathrm{d}\tau. \tag{2}$$

This transformation is a specific case of the general Sundman transformation $\mathrm{d}t = c_s r^{\alpha_s}\mathrm{d}\tau$ with $c_s = 2$ and $\alpha_s = 1$.[15] (It is well known that the $\alpha_s = 1$ case makes τ proportional to the eccentric anomaly.) The integrals that give $t(\tau)$ are available for each solution case in terms of Jacobi elliptic functions and elliptic integrals. However, the reverse transformation is not available analytically, and a numerical root-finding procedure must be used to determine $\tau(t)$. This feature mimics the inversion procedure required to solve Kepler's equation; thus, Eq. (2) is deemed the Stark equation.[6,12]

The 3D solution procedure uses an additional independent variable τ_2, whose value for the bounded case may be obtained analytically as a function of τ in terms of elliptic integrals.

ANALYTICAL SOLUTION OF THE STARK PROBLEM IN TERMS OF WEIERSTRASS ELLIPTIC FUNCTIONS

The Stark problem has recently been solved analytically in terms of Weierstrass elliptic functions by Biscani and Izzo,[12] who have also made available Python scripts that implement the described method.* Like the Jacobi case, the derivation begins with the formulation given by Eq. (1) and takes advantage of the separability of the Hamiltonian in parabolic coordinates. However, Biscani and Izzo claim several advantages of the Weierstrass formulation over the Jacobi formulation:

- There is no need to categorize the solution type based on initial conditions.
- Only a single independent variable transformation is used (Eq. (2)).
- The 3D case is solved directly rather than as a transformation of the 2D case.

For the purposes of implementation as a trajectory propagation tool for the bounded motion case, these characteristics provide less benefit than may be initially assumed. First, the categorization advantage becomes less meaningful in computation because calculation of the Weierstrass elliptic function $\wp$ is commonly performed in terms of Jacobi elliptic functions and elliptic integrals, and the calculation proceeds differently depending on the sign of the modular discriminant Δ.[16] Thus, three solution cases must be taken into account.† The differences in implementation between these three cases are quite small compared to those between the seven 2D and three 3D cases of the Jacobi formulation. However, the Jacobi solution for all bounded motion requires only two 2D solution methods and one 3D solution method. Thus, for investigations restricted to this common case, the difference in the number of solution cases between the Jacobi and Weierstrass methods is greatly reduced. The single form of the Weierstrass solution is therefore of more consequence if the Stark framework is used to model, for example, powered flybys or low-thrust escape maneuvers.

Second, the Weierstrass method's lack of a second time transformation eliminates the need to calculate the Jacobi method's independent variable τ_2. However, the cost is that the complexity of the final expression for the parabolic coordinate ϕ‡ is significantly increased.

Finally, as is the case for the Jacobi formulation, the Weierstrass solution to the 3D problem is not applicable to the 2D problem, meaning that both 2D and 3D solutions are required for a complete solution algorithm. Biscani and Izzo do not provide the formulae required for implementing the 2D solution.

Another difference between the Weierstrass and Jacobi methods is that the Weierstrass formulation makes heavy use of complex variables. A disadvantage of this feature is that easily obtainable code packages for calculating elliptic integrals and Jacobi elliptic functions often require real variables for all arguments, meaning that users must adapt these packages to be valid in the complex plane.[16]

*`https://github.com/bluescarni/stark_weierstrass`

†The script made available by Biscani and Izzo uses this technique.

‡$\phi = \texttt{atan2}(y, x)$

SOLUTION OF THE STARK PROBLEM IN TERMS OF TAYLOR SERIES EXPANSIONS

A separate technique for trajectory propagation within the Stark framework is based on an extension of the F and G Taylor series technique used to calculate a trajectory in a two-body force field.[11,17] Since, unlike the two-body problem, motion in the Stark model is not necessarily confined to a plane, a third basis vector is introduced to the series formulation so that the position vector is given by

$$\boldsymbol{r}(\tau') = \boldsymbol{r}(\tau'_0 + \Delta\tau') = F\boldsymbol{r}_0 + G\boldsymbol{v}_0 + H\boldsymbol{p}, \tag{3}$$

where $\boldsymbol{r}_0$ and $\boldsymbol{v}_0$ are the particle's initial position and velocity vectors, respectively; $\boldsymbol{p}$ is the disturbing acceleration vector; and F, G, and H are series expanded in increasing orders of $\Delta\tau'$. For the series method, the independent variable τ' may be any Sundman-transformed independent variable. Physical time is then calculated from another series expansion T, and velocity is calculated by

$$\boldsymbol{v} = \frac{\mathrm{d}\boldsymbol{r}}{\mathrm{d}t} = \frac{\mathrm{d}\boldsymbol{r}}{\mathrm{d}\tau'}\frac{\mathrm{d}\tau'}{\mathrm{d}t}. \tag{4}$$

Pellegrini, *et al*. develop recursion relations to facilitate the derivation of the terms of F, G, H, and T to arbitrarily high orders. Fortran driver routines and coefficient files up to order 18 are available as an electronic supplement.* The algebra required to carry out the recursions beyond the lowest orders is unmanageable by hand, and a symbolic manipulator (*e.g.*, Maple) is used calculate the terms of the series. However, the complexity of the expressions for coefficients beyond approximately order 25 leads to prohibitively long run times for (1) the symbolic manipulator and (2) the compilation of the resulting codes. The method is found to be most efficient for orders 12 – 18, depending on a variety of circumstances. As with any series method, truncation error eventually prevails as the step size increases. Of course, such a limit does not exist for either of the two analytical Stark methods discussed (Jacobi or Weierstrass).

If the desired step size is small enough, however, the series method has several advantages over the analytical methods:

- Though the series solution requires a high number of mathematical operations if taken to high orders, it does not require the calculation of elliptic integrals, Jacobi or Weierstrass elliptic functions, or the use of complex variables. Also, transformations to and from parabolic coordinates are avoided.
- The series solution is contained in a single set of formulae; there are no subcases or distinctions to be made based on initial conditions. Additionally, the 2D and 3D cases are handled with the same formulae.
- Propagation may be performed naturally in any Sundman-transformed τ', including those proportional to time ($\alpha_s = 0$), eccentric anomaly ($\alpha_s = 1$), true anomaly ($\alpha_s = 2$), and intermediate anomaly ($\alpha_s = 3/2$).† Both analytical formulations are naturally propagated

*`http://russell.ae.utexas.edu/index_files/fgstark.htm`

†The $\alpha_s = 1$ case for the series solution is found to be significantly more efficient than the other transformations.

only using the specific transformation given in Eq. (2), and propagation in physical time is possible only through numerical inversion of a nonlinear equation.

- Since the series method does not assume a direction for $\boldsymbol{p}$, the coordinate transformations required when using either of the analytical methods are avoided.

Other than the possibility of truncation error, the primary disadvantages of using the series method for propagation are the requirement of a symbolic manipulator to determine the series coefficients and the large file sizes required to hold the coefficients. (However, as previously mentioned, the coefficients are precomputed and associated files are available online.) It is noted that the Modern Taylor Series method can be used to alleviate these latter two drawbacks, but suffers from numerical problems at the extreme high orders and is less efficient than the F and G series method at lower orders.[11]

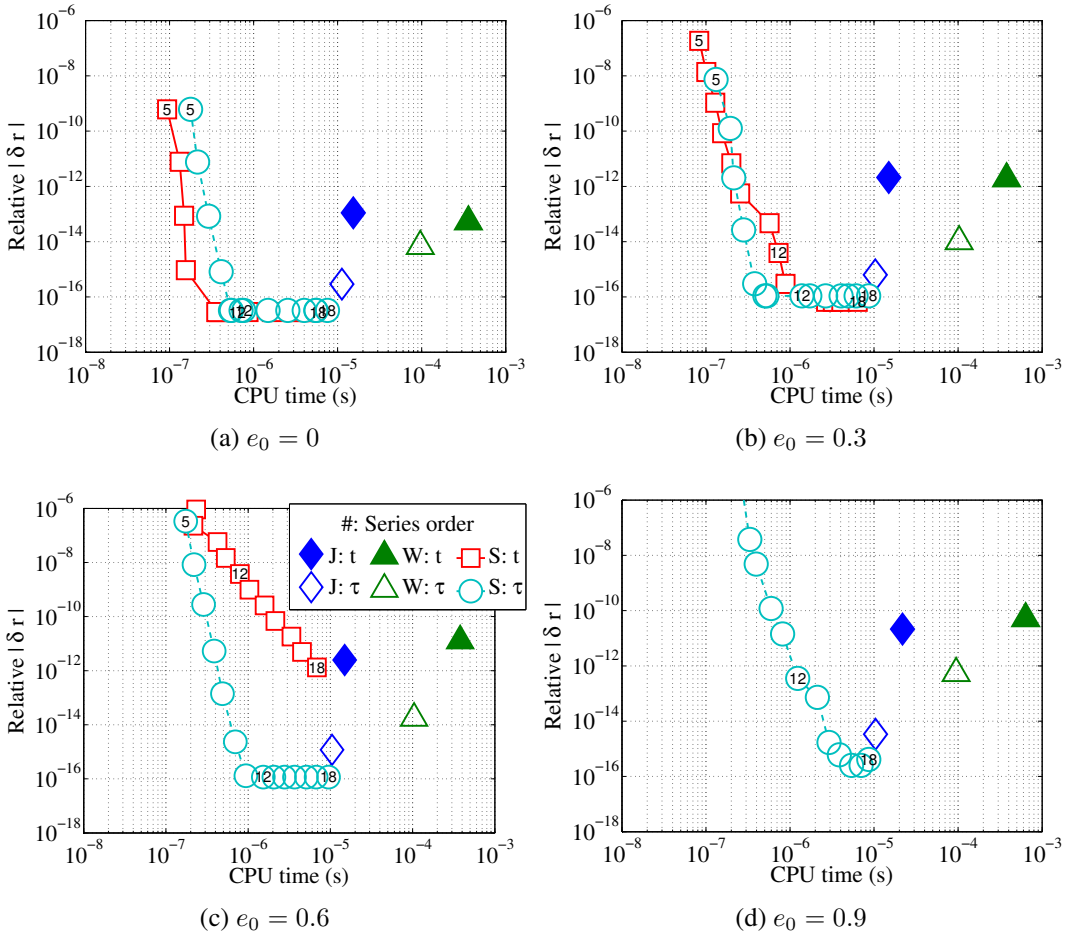
COMPUTATIONAL COMPARISON OF STARK PROBLEM SOLUTION TECHNIQUES

The Jacobi, Weierstrass, and Taylor series solution methods are compared against one another in the computation of a single trajectory segment in which the disturbing acceleration vector is held inertially fixed. This comparison is performed to assess the relative speed and accuracy of the three solution techniques in a situation that models the dynamics exactly. Each method is implemented in Fortran; the Jacobi algorithm is based on the work of Lantoine and Russell,[6] the Weierstrass algorithm is a translation* of the Python script made available by Biscani and Izzo, and the Taylor series algorithm is the implementation made available by Pellegrini, *et al*. All simulations in this study use the GNU Fortran compiler (gfortran) 4.8.2 on a Mac OS X 10.9 workstation with a dual-core Intel Core i5-3210M CPU with 2.5 GHz clock speed and 16 GB of RAM. The authors acknowledge that timing results are highly sensitive to choices in algorithm implementation, hardware, compiler, compiler settings, and other circumstances. The results are intended to allow a practitioner to qualitatively calibrate the relative performance of the multiple methods, provided that a good-faith effort has placed all of the methods on a level playing field (*i.e.*, same language, compiler, compiler settings, *etc*.).

For each test condition and each solution method, propagation is performed using both time and τ (Eq. (2)) as the independent variable. The order of the series is varied from 5 – 18, the highest order made available by Pellegrini, *et al*. Accuracy is assessed by comparing the states returned by each of the solution methods to a variable-order, variable-step, linear multistep integrator (LSODE[18]), implemented using quadruple-precision arithmetic, with relative tolerance for all state variables set to 10^{-25}. Computation time for the three Stark methods is determined using the intrinsic Fortran routine `cpu_time`; 10,000 trials of each solution method are run, and the reported computation time is the average time required for a single trial. The tests are performed in normalized units with $\mu = 1$, $\epsilon = 10^{-3}$ and initial classical orbital elements $a_0 = 1$, $i_0 = 28.5^\circ$, $\Omega_0 = 5^\circ$, and $\omega = 10^\circ$. The initial eccentricity is varied from 0 to 0.9. The value of ϵ is selected because the Taylor series solution becomes less efficient as ϵ increases, and $\epsilon = 10^{-3}$ is similar in magnitude in normalized units to the perturbing acceleration caused by typical time-varying perturbations, such as low thrust or Earth oblateness effects. The spacecraft begins at periapsis for all tests because

*The algorithm is modified slightly in the Fortran implementation. Much of the object-oriented framework of the Python script is removed, and capabilities are added to determine the final velocity vector and to invert the Stark equation. However, as much as possible of the original algorithm is kept intact.

periapsis is the most computationally challenging state for the series solution method and the Stark equation inversion techniques.



(a) $e_0 = 0$ (b) $e_0 = 0.3$

(c) $e_0 = 0.6$ (d) $e_0 = 0.9$

Figure 2: Magnitude of relative position error for Jacobi ("J"), Weierstrass ("W"), and series ("S") solutions for $\Delta M = 5$ deg. For each series solution, the order progresses with each data point. Orders 5, 12, and 18 are labeled.

Fig. 2 shows the relative position difference between each method and the numerically integrated solution as a function of CPU time required for a modest advancement in mean anomaly, $\Delta M = 5$ deg. The value of $\Delta\tau$ corresponding to ΔM is determined using quadruple-precision LSODE integration; thus, Stark equation inversion is eliminated as a source of inaccuracy for the propagations in τ. Data points corresponding to series orders 5, 12, and 18 are labeled for reference. The series solution propagated in t is not shown in Fig. 2d because the series is divergent for the high eccentricity case.

Several observations pertaining to solution accuracy may be made immediately. First, as the order of the series increases, the relative error generally decreases (until a floor is reached) and computation time generally increases. Second, the inaccuracy in the Jacobi and Weierstrass solutions propagated in t relative to the same solutions propagated in τ is due to the numerical inversion of the Stark equation. Finally, regardless of whether the Stark equation is inverted, inaccuracy ap-

pears in the analytical solutions due to extensive reliance on transcendental functions. The highly nonlinear nature of these functions results in the growth of the effects of calculation inaccuracy and roundoff error as the solution procedure progresses. Quadruple-precision arithmetic could be used to improve accuracy at the expense of computation time.

Of the analytical methods, the Weierstrass solution is consistently slower than the Jacobi solution.* However, the authors acknowledge that more efficient versions of both the Jacobi and Weierstrass solutions could provide different results (despite efforts to achieve efficient versions of both). In particular, efficiency may be gained by directly computing the Weierstrass elliptic function $\wp$ rather than calculating $\wp$ as a function of Jacobi elliptic functions and elliptic integrals.

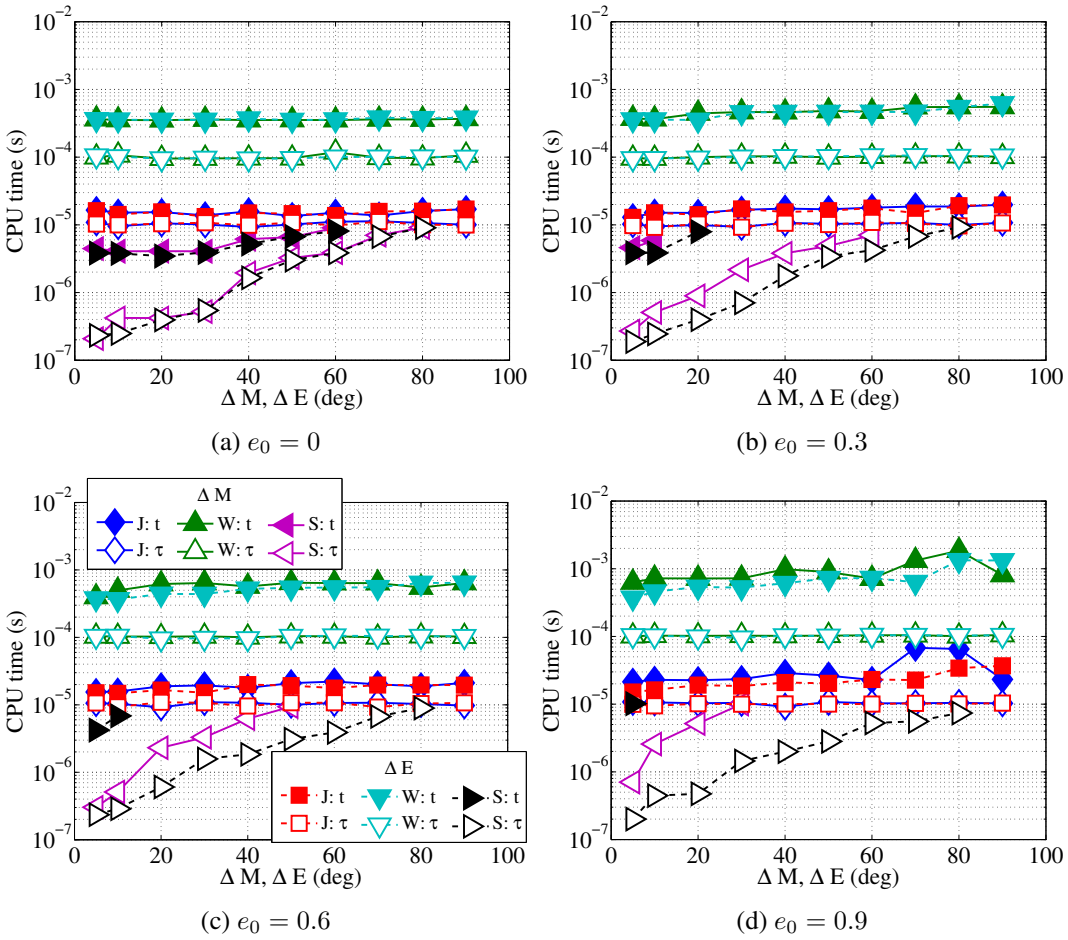


Figure 3: CPU time of Stark step vs. advancement of mean anomaly and eccentric anomaly for Jacobi ("J"), Weierstrass ("W"), and series ("S") solution methods. Series order is varied to achieve relative error tolerance 10^{-12} compared to numerically integrated solution.

From Figure 2, it is clear that the series solutions are more computationally efficient than the

*Timing tests are also carried out on the original Python scripts made available by Biscani and Izzo. The Python version is found to be multiple orders of magnitude slower than the current Fortran implementation. Of course, such a comparison is not fair because Python is not a compiled language. Other differences that affect timing include (1) the software packages used to evaluate elliptic functions and integrals and (2) the object-oriented structure of the original Python script.

analytical solutions for a small step size when propagated in t at low-to-moderate eccentricities and when propagated in τ at eccentricities up to and including 0.9. However, one motivation for using the Stark framework is the potential for gaining efficiency through the use of large step sizes. Figure 3 shows the variation in required computation time as a function of advancement of both mean anomaly and eccentric anomaly. For the series solution, the order of the series plotted corresponds to the minimum order such that

$$\frac{|\boldsymbol{r}_{f,series} - \boldsymbol{r}_{f,LSODE}|}{|\boldsymbol{r}_{f,LSODE}|} < r_{tol}, \quad r_{tol} = 10^{-12}, \tag{5}$$

where $\boldsymbol{r}_f$ is the position vector of the particle at the end of the propagation period. As initial eccentricity increases, the ability of the series method to provide an accurate solution is adversely affected, particularly when propagating in t. At low eccentricities, however, this unfavorable characteristic is offset by the speed of the series solutions relative to the analytical solutions.

The results of Figure 3 at high eccentricities also motivate the use of a constant step size in a unit other than mean anomaly (*i.e.*, time) for eccentric orbits. In an eccentric orbit, constant steps in mean anomaly cluster near apoapsis – where the particle's states change most slowly – and are widely separated near periapsis, where the particle's states change most rapidly. On the other hand, when constant steps are taken in eccentric anomaly, the distribution of steps is geometrically symmetric over the orbit. As a result, the series solutions propagated in eccentric anomaly retain their efficiency characteristics and are applicable for larger step sizes at high eccentricities compared to the constant ΔM case.

However, regardless of the method of step size determination, the series solutions suffer from a separate issue that can degrade their speed advantage over the analytical methods: How does one determine the necessary number of terms to achieve a desired accuracy without the benefit of an independent truth model? As Fig. 2 makes clear, the efficiency discrepancy between the series methods and the Jacobi elliptic function method almost disappears as the number of series terms approaches 18. One solution is to dynamically adjust the order of the series based on accuracy approximations of a previous step.

SUMMARY OF STARK PROBLEM SOLUTION TECHNIQUES

A summary of the characteristics of the Stark problem solution methods discussed in this study is presented in Table 1.

SIMS-FLANAGAN MODEL

The Sims-Flanagan model approximates disturbed two-body dynamics through the use of a series of pure Keplerian arcs.[1] At the node connecting two arcs, a discontinuity in the velocity vector is introduced to account for the disturbing acceleration experienced by the spacecraft in the vicinity of the node. The discontinuity $\Delta\boldsymbol{v}$ is given by

$$\Delta\boldsymbol{v} = \boldsymbol{p}\Delta t, \tag{6}$$

where $\boldsymbol{p}$ is a disturbing acceleration vector and Δt is the time difference between nodes. This model has been widely used to approximate low-thrust trajectories.[2,3] Here, the method is used to model

Table 1: Summary of Stark problem solution techniques.

	Jacobi elliptic	**Weierstrass elliptic**	***F* & *G* Taylor series**
Relative execution speed	Medium	Slow	Medium to fast, depending on number of terms used
Relative code file size	Medium	Small	Large
2D solution	Distinct from 3D solution	None published	Same as 3D solution
Independent variable	τ for 2D; τ and τ_2 for 3D; t with numerical inversion	τ; t with numerical inversion	Any Sundman-transformed independent variable
Required capabilities	Elliptic integrals of the first, second, and third kinds; Jacobi elliptic functions	Weierstrass elliptic functions; complex arithmetic	Initial use of symbolic manipulator to derive coefficients
Primary sources of inaccuracy	Calculation of Jacobi elliptic functions and other transcendental functions; inversion of Stark equation	Calculation of Weierstrass elliptic functions and other transcendental functions; inversion of Stark equation	Truncation error due to large step sizes

the trajectory of a spacecraft subjected to natural, continuously time-varying perturbations. The results, along with a classic numerical integration method, are compared to the Stark propagation techniques. As with the Stark problem, there exist multiple methods for solving Kepler's problem. To provide a mirroring comparison to the Stark-based results, two techniques are compared in the implementation of the Sims-Flanagan model. The first is an analytical method that uses the universal variable formulation,[17] and the second is the F and G Taylor series solution as implemented by Pellegrini, *et al.*[11]

APPLICATION OF THE SIMS-FLANAGAN AND STARK MODELS TO TIME-VARYING PERTURBATIONS

The method by which the Stark model is applied to time-varying perturbations is similar to how the Sims-Flanagan model uses Keplerian arcs and $\Delta\boldsymbol{v}$'s. Because the Stark model requires that the disturbing acceleration be inertially fixed, application to time-varying perturbations requires discretization of the trajectory. At the node between two trajectory segments, the disturbing acceleration vector is allowed to change discontinuously. Within each segment, the Stark model is used to approximate the true system dynamics.

Determination of the Disturbing Acceleration Vector

Determination of a representative disturbing acceleration vector $\boldsymbol{p}$ at each node of a trajectory is fundamental to the accuracy of both the Stark model and the Sims-Flanagan model. The simplest method is to evaluate $\boldsymbol{p}$ using the spacecraft state at the current node, which, for the Sims-Flanagan model, gives

$$\Delta\boldsymbol{v}_{t_i \to t_{i+1}} = \boldsymbol{p}\left(t_i, \boldsymbol{X}_i\right)\left(t_{i+1} - t_i\right), \tag{7}$$

where $\boldsymbol{X}_i$ is the state of the spacecraft at t_i. As shown in Figure 4, if $\Delta\boldsymbol{v}$ is thought of as the time integral of the disturbing acceleration, Eq. (7) may be recognized as analogous to the forward Euler method of numerical integration.[19,13] However, just as the forward Euler method is not the only numerical integration algorithm, a myriad of other options exist for computing the $\Delta\boldsymbol{v}$ at each node in the Sims-Flanagan model or the acceleration vector for each segment of a trajectory in the Stark model. In particular, it is desired to take advantage of the fact that the disturbing acceleration vectors caused by natural perturbations are generally smooth and differentiable. Thus, a determination scheme based on the explicit, linear multistep integration method known as Adams-Bashforth[19,20] is implemented. In general, the Adams-Bashforth method approximates the state equations of a system of differential equations by an interpolating polynomial of degree k. The previous $k+1$ data points are used to determine the coefficients of the polynomial. Analytical integration of the polynomial is trivial and is used to obtain an approximate value for the states at the next time step. For example, the one-step Adams-Bashforth formula is simply the forward Euler method. The second-through-fifth-order Adams-Bashforth formulae in the Sims-Flanagan context for a fixed time step Δt are, respectively,[20]

$$\Delta \boldsymbol{v}_{t_i \to t_{i+1}} = \frac{1}{2}\Delta t \left(3\boldsymbol{p}\left(t_i, \boldsymbol{X}_i\right) - \boldsymbol{p}\left(t_{i-1}, \boldsymbol{X}_{i-1}\right)\right) \tag{8}$$

$$\Delta \boldsymbol{v}_{t_i \to t_{i+1}} = \frac{1}{12}\Delta t \left(23\boldsymbol{p}\left(t_i, \boldsymbol{X}_i\right) - 16\boldsymbol{p}\left(t_{i-1}, \boldsymbol{X}_{i-1}\right) + 5\boldsymbol{p}\left(t_{i-2}, \boldsymbol{X}_{i-2}\right)\right) \tag{9}$$

$$\Delta \boldsymbol{v}_{t_i \to t_{i+1}} = \frac{1}{24}\Delta t \left(55\boldsymbol{p}\left(t_i, \boldsymbol{X}_i\right) - 59\boldsymbol{p}\left(t_{i-1}, \boldsymbol{X}_{i-1}\right) + 37\boldsymbol{p}\left(t_{i-2}, \boldsymbol{X}_{i-2}\right) - 12\boldsymbol{p}\left(t_{i-3}, \boldsymbol{X}_{i-3}\right)\right) \tag{10}$$

$$\begin{aligned}\Delta \boldsymbol{v}_{t_i \to t_{i+1}} = \frac{1}{720}\Delta t \big(&1901\boldsymbol{p}\left(t_i, \boldsymbol{X}_i\right) - 2774\boldsymbol{p}\left(t_{i-1}, \boldsymbol{X}_{i-1}\right) + 2616\boldsymbol{p}\left(t_{i-2}, \boldsymbol{X}_{i-2}\right) \\ &-1274\boldsymbol{p}\left(t_{i-3}, \boldsymbol{X}_{i-3}\right) + 251\boldsymbol{p}\left(t_{i-4}, \boldsymbol{X}_{i-4}\right)\big).\end{aligned} \tag{11}$$

For the Stark model, Eq. (7) – Eq. (11) are used to obtain a disturbing acceleration rather than a velocity discontinuity:

$$\boldsymbol{p}_{Stark, t_i \to t_{i+1}} = \frac{\Delta \boldsymbol{v}_{t_i \to t_{i+1}}}{\Delta t} \tag{12}$$

The use of the Adams-Bashforth method to determine $\Delta \boldsymbol{v}_{t_i \to t_{i+1}}$ at each node for the Sims-Flanagan model or $\boldsymbol{p}_{Stark, t_i \to t_{i+1}}$ for each segment for the Stark model presents several advantages. First, the Adams-Bashforth method gives the Sims-Flanagan and Stark models some ability to predict – and take advantage of – the evolution of $\boldsymbol{p}$ between t_i and t_{i+1}. Second, at each step, the method uses quantities that have already been calculated at previous steps. These quantities may be saved rather than recalculated, meaning that only a single Kepler or Stark propagation need be performed at each step. If the number of propagation segments is held constant, the Adams-Bashforth method therefore requires less computation time than, for example, the Runge-Kutta solution family, which requires multiple propagations at each step.[19] Finally, since Adams-Bashforth is an explicit integration method, the solution of a nonlinear system of equations is not required, and less computation time is required than for implicit integration methods (*e.g.*, Adams-Moulton).[19,20]

The Adams-Bashforth method is not without its disadvantages, however. One is the necessity of a startup procedure. Since the previous $k+1$ data points are required for the method, at early steps when fewer than $k+1$ previous points exist, another procedure must be followed. In this implementation, the order of the method is reduced to the maximum that is possible given the number of available backpoints. That is, if using a k-step method, if $j < k+1$ backpoints are available, the method used is reduced to a j-step method.

There are also several concerns related to step size. First, as with any integration method, too large of a step size leads to inaccurate results. Second, the most basic implementations of the linear multistep methods (such as that presented in Eq. (7) – Eq. (11)) require constant Δt between backpoints. If a change in step size is desired, a new set of backpoints is needed with the correct constant spacing.[21] As demonstrated in Figure 3, a constant Δt is undesirable for appreciably eccentric orbits. One approach is to adaptively change the step size in time by following a constant step size in a more suitable independent variable, such as eccentric anomaly. For the Sims-Flanagan case, propagating in constant steps of eccentric anomaly presents a negligible increase in computation time because Δt as a function of ΔE is available analytically via Kepler's equation. Position and velocity as a function of ΔE may also be computed analytically.[22] However, because disturbing

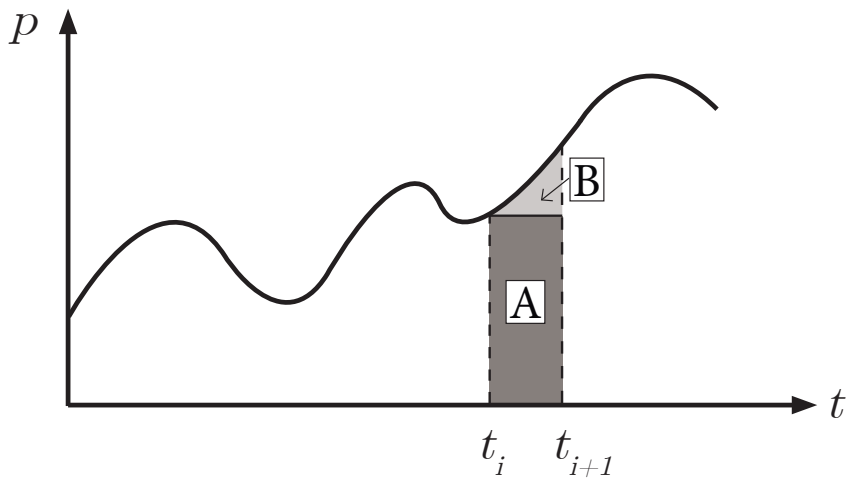


Figure 4: Analogy of $\Delta \boldsymbol{v}$ as integral of disturbing acceleration $\boldsymbol{p}$ (scalar case). The area of region A is the one-step Adams-Bashforth approximation of $\Delta \boldsymbol{v}_{t_i \to t_{i+1}}$, while the combined area of regions A and B is the true $\Delta \boldsymbol{v}_{t_i \to t_{i+1}}$.

accelerations are generally known as second derivatives of position with respect to t and not with respect to E, this procedure still requires that $\Delta \boldsymbol{v}_{t_i \to t_{i+1}}$ be computed using products of disturbing accelerations and changes in t. Since each Δt may be calculated as a function of the constant ΔE, this is possible, but Eq. (7) – Eq. (11) must be adapted: The coefficients weighting the previous data points are changed at each step to account for the uneven (in time) distribution of backpoints.[21]

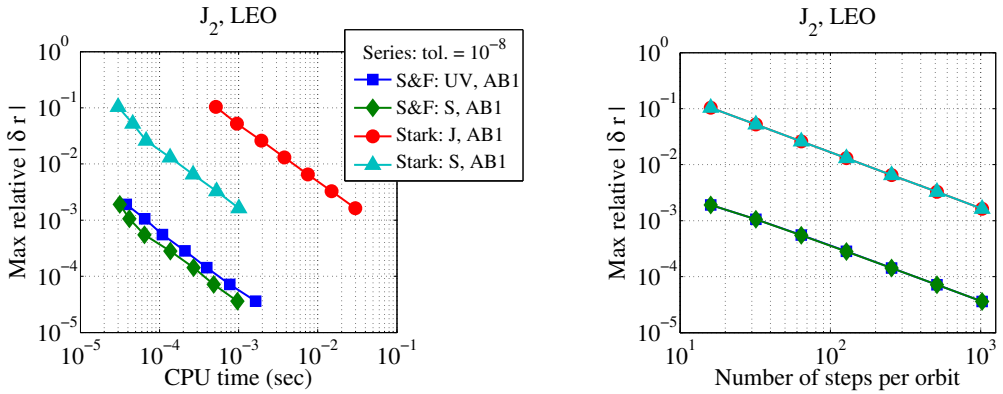
For the Stark case, propagating in constant steps of ΔE circumvents the need to invert the Stark equation since τ is proportional to E, and Δt as a function of $\Delta \tau$ is available analytically. However, as with the Sims-Flanagan model, adaptation of the Adams-Bashforth coefficients is still required because of the unequal steps in time. In this study, simulations are restricted to the constant Δt case for both the Sims-Flanagan and Stark models.

Results

Two simulations are presented to (1) assess the effectiveness of the Sims-Flanagan and Stark frameworks in modeling continuously time-varying perturbations and (2) obtain preliminary indications of the utility of the Adams-Bashforth algorithm in calculating the disturbing accelerations used by each model. It is emphasized that more extensive study is required to determine fully the merits of the Adams-Bashforth method presented in this paper. Each simulation begins in LEO: $a_0 = 6578$ km, $e_0 = 0$, $i_0 = 28.5°$, $\Omega_0 = 5°$. In simulation A, the object is disturbed by Earth's oblateness (J_2). In simulation B, the object is disturbed by atmospheric drag; the ballistic coefficient of the object is 152 $\mathrm{kg/m^2}$. These two conditions are selected due to their high applicability for objects in LEO and their strongly time-varying nature. For each simulation, the object is propagated for three orbit periods.* The number of steps per orbit is varied from 16 to 1024 in powers of two. Figures 5 and 6 show the accuracy of the Stark and Sims-Flanagan models when a first-order Adams-Bashforth method is applied. Figures 7 and 8 show the effect of increasing the order of the

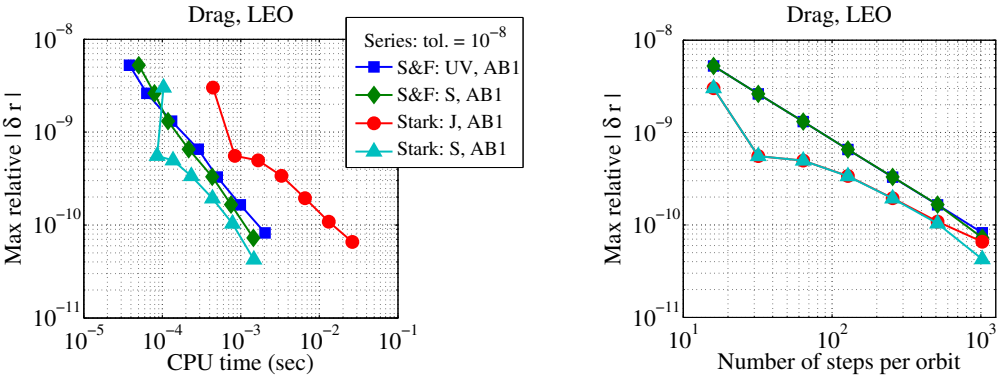
*Of course, since the orbit is perturbed, the period is continuously changing. For propagation duration purposes, the period is calculated using the initial conditions and held constant throughout the propagation.

Adams-Bashforth method.† The Stark Weierstrass formulation results are omitted to improve readability because the Weierstrass method is shown to produce the same results as the Jacobi method, but require more computation time in the current formulation (Figures 2 and 3).



(a) (b)

Figure 5: Accuracy over three orbital periods for Simulation A using first-order Adams-Bashforth algorithm ("AB1"). Universal variable ("UV") and series ("S") solutions are shown for the Sims-Flanagan ("S&F") model; Jacobi ("J") and series solutions are shown for the Stark model.

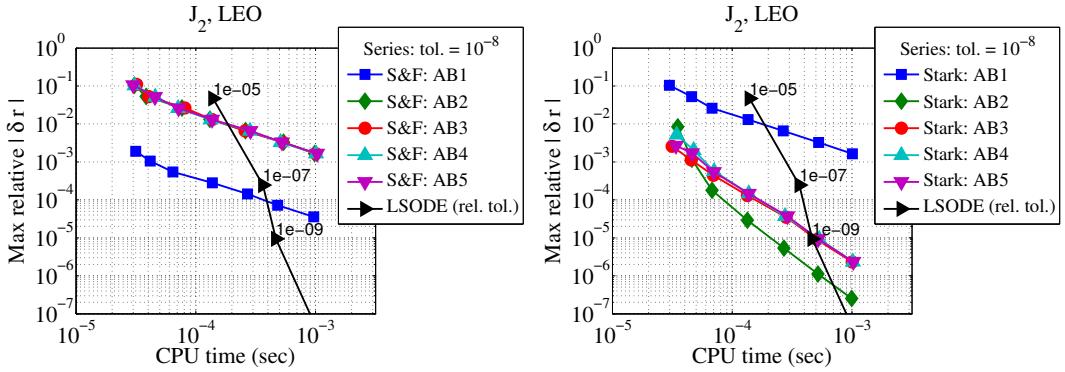


(a) (b)

Figure 6: Accuracy over three orbital periods for Simulation B.

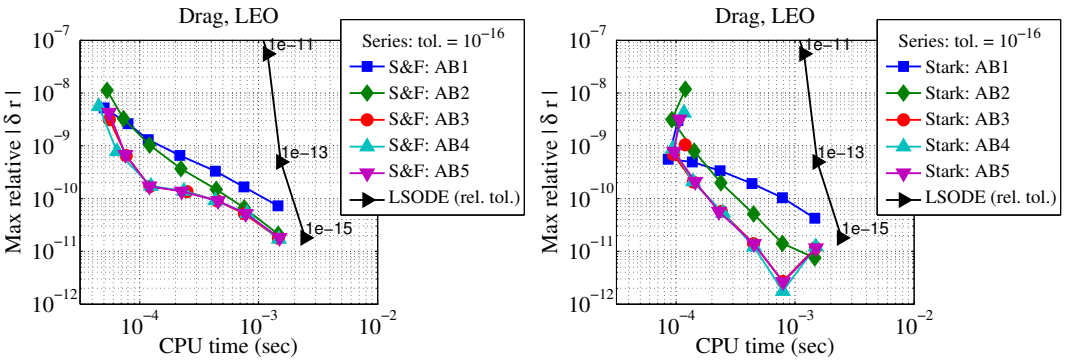
In order to yield a simple, direct comparison between the two models, propagation is performed using time as the independent variable. The Stark model takes into account a continuously applied disturbing acceleration between nodes, while the Sims-Flanagan model does not. The result is that the same change in eccentric anomaly results in different changes in time in the Stark and Sims-Flanagan models; the use of constant time steps circumvents this issue. This choice is justified when the propagated orbits are not appreciably eccentric, as is the case in the simulations presented here.

†To improve the readability of Figures 7 and 8, only the Taylor series solutions are shown. Because increasing the number of steps used in the Adams-Bashforth algorithm does not increase the number of propagations required, the speed comparisons between the analytical and series solutions shown in Figures 5 and 6 remain valid.



(a) Sims-Flanagan model. (b) Stark model.

Figure 7: Accuracy over three orbital periods for Simulation A series solutions compared to classic numerical integration.



(a) Sims-Flanagan model. (b) Stark model.

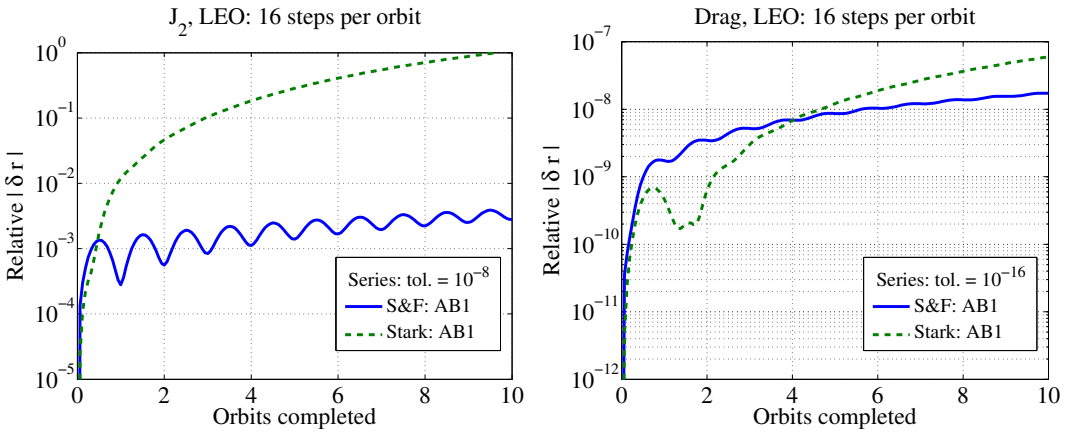
Figure 8: Accuracy over three orbital periods for Simulation B series solutions compared to classic numerical integration.

The order of the series solutions used to perform both the Stark and Kepler propagations is allowed to vary based on a tolerance placed on the magnitude of the estimated position truncation error. The tolerance is selected empirically to be smaller than the error introduced by the approximations of the models. For Simulation A, the tolerance is 10^{-8}, and, for Simulation B, the tolerance is 10^{-16}.

Computational speed is assessed in the manner described in the analysis of a single step in the Stark model. Accuracy is once again computed using a comparison to an LSODE-integrated solution in quadruple precision with a relative accuracy tolerance of 10^{-25}. However, since multiple steps are taken in this case, the reported inaccuracy is the maximum magnitude of position difference between the evaluated model and the LSODE model at any step of the propagation.

As may be expected, both Figures 5 and 6 show that accuracy improves as the number of steps per orbit increases. Additionally, the superior efficiency of the series solution for the Stark model ("Stark: S") relative to the Jacobi solution ("Stark: J") is evident. On the other hand, the universal variable analytical ("S&F: UV") and series ("S&F: S") solutions used to solve the Kepler problem

in the Sims-Flanagan model require comparable computation time. For both the Kepler and Stark propagations, the efficiency of the series solution increases relative to the analytical solution as the number of steps per orbit increases because fewer series terms are required to achieve the prescribed accuracy tolerance.



(a) Simulation A: J_2 perturbation.

(b) Simulation B: drag perturbation.

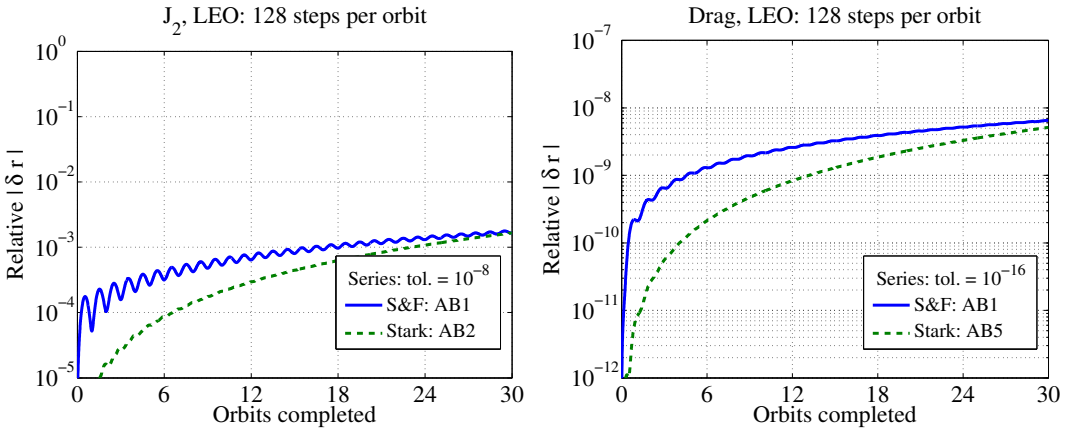
Figure 9: Accuracy as a function of propagation time for series solutions. Note different vertical axis scalings are used due to differences in perturbation magnitudes.

The accuracy characteristics of the two simulations are significantly different. From Figure 5, the Sims-Flanagan model is clearly more accurate than the Stark model when the same number of steps per orbit is used in each model for Simulation A. Meanwhile, Figure 6 shows that, for Simulation B, the two solution methods tend to produce results of comparable accuracy, with the Stark model actually more accurate than the Sims-Flanagan model. However, further testing suggests that the apparent superiority of the Stark model for Simulation B is an artifact of the chosen propagation time; as shown in Figure 9b, soon after the propagation surpasses three orbital periods, the Sims-Flanagan model becomes more accurate than the Stark model. From both Figures 9a and 9b, the accumulation of error in the Sims-Flanagan model is shown to occur more slowly than in the Stark model. It is hypothesized that the inaccuracy of the Stark model when using the one-step Adams-Bashforth method is caused by the fact that the modeled disturbing acceleration always lags the true disturbing acceleration. On the other hand, each $\Delta \boldsymbol{v}$ in the Sims-Flanagan model is applied in an instantaneously correct direction, even though the disturbing acceleration is not applied continuously.

The use of higher-order Adams-Bashforth methods in conjunction with the Sims-Flanagan model produces inconsistent results: The accuracy of the propagation decreases in Simulation A (Figure 7a), but increases in Simulation B (Figure 8a). Meanwhile, the use of higher-order Adams-Bashforth methods consistently and significantly improves the accuracy of the Stark model (Figures 7b and 8b). It is hypothesized that the reason for this discrepancy is that the use of higher-order Adams-Bashforth methods allows the disturbing acceleration vector used in each step of the Stark model to more closely resemble an "average" disturbing acceleration over the step. In the Sims-Flanagan model, on the other hand, the disturbing acceleration at a node i determined using the one-step Adams-Bashforth method may already be thought of as a similar type of average disturbing acceleration over the interval $(t_i - \Delta t/2) < t < (t_i + \Delta t/2)$. The instantaneous $\Delta \boldsymbol{v}$ at t_i may

therefore be considered analogous to midpoint-rule integration over $(t_i - \Delta t/2) < t < (t_i + \Delta t/2)$ in addition to its connection to forward Euler integration over the time span $t_i < t < t_{i+1}$. This comparison does not apply to the Stark method because the entirety of the disturbing acceleration is not applied instantaneously. Once again, it is noted that more complete testing is necessary before general conclusions may be drawn.

Also shown in Figures 7 and 8 are data points corresponding to LSODE numerical integration with varied relative tolerance parameter. In general, for an analytical approximation to be worthwhile in computation, the method must achieve an accuracy superior to classic numerical integration while requiring less computation time. Comparison to a single integration algorithm is not intended to constitute a thorough study of the effectiveness of the Sims-Flanagan and Stark models compared to all classic numerical integrators. Nevertheless, it is noted that, particularly for the low-magnitude atmospheric drag perturbation modeled in Simulation B (Figure 8), both the Sims-Flanagan and Stark models produce results that indicate their use is justified in situations in which rapid computation is more important than high accuracy.



(a) Simluation A: J_2 perturbation.

(b) Simulation B: drag perturbation.

Figure 10: Accuracy as a function of propagation time for series solutions. Note different vertical axis scalings are used due to differences in perturbation magnitudes.

Figure 10 shows the 128-step-per-orbit accuracies for Simulation A (Figure 10a) and Simulation B (Figure 10b) as functions of propagation time when multistep Adams-Bashforth methods are used with the Stark propagation.* The use of higher-order Adams-Bashforth methods is seen to delay the time at which the first-order Sims-Flanagan model becomes more accurate than the Stark model (compare to Figure 9). Nevertheless, the first-order Sims-Flanagan model still exhibits a slower error accumulation rate than the multistep Stark models as the propagation time increases.

CONCLUSION

Three recently introduced methods of solving the Stark problem – one based on Jacobi elliptic and related functions, one based on Weierstrass elliptic and related functions, and one based on Taylor series expansions – are compared for the bounded motion case. In order to provide a fair

*The order of the Adams-Bashforth plotted for the Stark model for each simulation is that which is most accurate after three orbital periods.

computational comparison to the published series solution routines, Fortran routines to solve the Jacobi and Weierstrass methods are implemented. In the current implementation, the Jacobi formulation is found to be more efficient than the Weierstrass formulation. The series method is found to be the fastest of the three for series orders up to approximately 18. Thus, the analytical solutions are preferable only for long-duration propagations for which truncation error is unacceptably large for series of moderate order. However, the series solution is found to give computational savings for propagations of up to approximately 80 deg of eccentric anomaly if the series is expanded in eccentric anomaly, even for significantly eccentric orbits.

The Stark model is also compared against the Sims-Flanagan model and classic numerical integration in the propagation of orbits with continuously time-varying perturbations. The use of a method based on the Adams-Bashforth numerical integration algorithm to compute disturbing accelerations for the Sims-Flanagan and Stark models is proposed and implemented. The method uses disturbing acceleration information from previous propagation steps to calculate a higher-order approximation of the effects of the perturbation between two steps. The increase in computation time compared to a first-order approximation is minimal because no additional propagations are required. Results are presented for two example simulations.

For both simulations, the Stark model is found to be more accurate than the Sims-Flanagan model over short propagation times. However, the Sims-Flanagan model is found to be superior in limiting the growth of error as propagation time increases. For the Stark model, the use of multistep Adams-Bashforth methods is found to improve accuracy for suitably small step sizes, while, for the Sims-Flanagan model, results are found to be inconsistent. It is emphasized that these results are limited to the example simulations presented in this paper. A more rigorous and thorough investigation is necessary to determine generally the effectiveness of the Adams-Bashforth methods introduced in this study.

There are found to be regions of the accuracy vs. efficiency space for the two example simulations in which both the Sims-Flanagan and Stark models provide superior performance compared to the classic numerical integrator LSODE. This region is seen to expand as the magnitude of the time-varying perturbation decreases. These results provide justification for the widespread use of the Sims-Flanagan model – and motivation for increased utilization of the Stark model – for efficiently approximating perturbed Keplerian motion. Further, the application of methods like the Adams-Bashforth algorithm introduced in this paper have the potential to improve the accuracy of both of these analytical approximations without sacrificing efficiency. The analytical models may then provide rapid approximations that more closely resemble true dynamics.

ACKNOWLEDGEMENTS

The authors would like to thank Ricardo León Restrepo Gomez for fruitful conversations regarding the Adams-Bashforth method. The authors would also like to thank Gregory Lantoine and Nicholas Bradley for their assistance in the implementation of the Jacobi formulation of the Stark problem and Etienne Pellegrini for his assistance in implementing the series formulation of the Stark problem.

REFERENCES

[1] J. A. Sims and S. N. Flanagan, "Preliminary Design of Low-Thrust Interplanetary Missions," *AAS/AIAA Astrodynamics Specialist Conference and Exhibit*, Girdwood, AK, Aug 1999.

[2] J. A. Sims, P. Finlayson, E. Rinderle, M. Vavrina, and T. Kowalkowski, "Implementation of a Low-Thrust Trajectory Optimization Algorithm for Preliminary Design," *AAS/AIAA Astrodynamics Specialist Conference and Exhibit*, Keystone, CO, Aug 2006.

[3] C. H. Yam and J. M. Longuski, "Reduced Parameterization for Optimization of Low-Thrust Gravity-Assist Trajectories: Case Studies," *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, Keystone, CO, August 2006.

[4] J. Stark, "Beobachtungen ber den Effekt des elektrischen Feldes auf Spektrallinien. I. Quereffekt. (Observations on the Effect of the Electric Field on Spectral Lines. I. Transverse Effect.)," *Annalen der Physik (Annals of Physics)*, Vol. 43, 1914, pp. 965–983.

[5] J. L. Lagrange, *Mécanique Analytique*. Paris: Courcier, 1788.

[6] G. Lantoine and R. P. Russell, "Complete closed-form solutions of the Stark problem," *Celestial Mechanics and Dynamical Astronomy*, Vol. 109, No. 4, 2011, pp. 333–366. DOI 10.1007/s10569-010-9331-1.

[7] V. V. Beletsky, *Essays on the Motion of Celestial Bodies*. Basel, Switzerland: Birkhäuser Verlag, 2001.

[8] Y. Isayev and A. Kunitsyn, "To the Problem of Satellite's Perturbed Motion Under the Influence of Solar Radiation Pressure," *Celestial Mechanics and Dynamical Astronomy*, Vol. 6, 1972, pp. 44–51.

[9] D. Rufer, "Trajectory Optimization by Making Use of the Closed Solution of Constant Thrust-Acceleration Motion," *Celestial Mechanics and Dynamical Astronomy*, Vol. 14, 1976, pp. 91–103.

[10] U. Kirchgraber, "A Problem of Orbital Dynamics, Which is Separable in KS-Variables," *Celestial Mechanics and Dynamical Astronomy*, Vol. 4, 1971, pp. 340–347.

[11] E. Pellegrini, R. P. Russell, and V. Vittaldev, "F and G Taylor Series Solutions to the Stark and Kepler Problems with Sundman Transformations," *Celestial Mechanics and Dynamical Astronomy*, Submitted: 2013. To appear.

[12] F. Biscani and D. Izzo, "The Stark problem in the Weierstrassian formalism," *Monthly Notices of the Royal Astronomical Society*, Submitted: June 2013.

[13] G. Lantoine and R. Russell, "The Stark Model: An Exact, Closed-Form Approach to Low-Thrust Trajectory Optimization," *21st International Symposium on Space Flight Dynamics*, 2009.

[14] G. Lantoine, *A Methodology for Robust Optimization of Low-Thrust Trajectories in Multi-Body Environments*. PhD thesis, Georgia Institute of Technology, December 2010.

[15] K. F. Sundman, "Mémoire sur le problème des trois corps," *Acta Mathematica*, Vol. 36, Dec. 1912, pp. 105 – 179.

[16] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions With Formulas, Graphs, and Mathematical Tables*. Courier Dover Publications, 1972. pp. 569 – 580, 649.

[17] R. R. Bate, D. D. Mueller, and J. E. White, *Fundamentals of Astrodynamics*. Dover Publications, Inc., 1971. pp. 177 – 226.

[18] K. Radhakrishnan and A. Hindmarsh, "Description and Use of LSODE , the Livermore Solver for Ordinary Differential Equations," NASA Reference Publication 1327, NASA, 1993.

[19] W. E. Boyce and R. C. DiPrima, *Elementary Differential Equations and Boundary Value Problems*. John Wiley & Sons, Inc., eighth ed., 2005. pp. 462 – 467.

[20] M. M. Berry and L. M. Healy, "Implementation of Gauss-Jackson Integration for Orbit Propagation," *The Journal of the Astronautical Sciences*, Vol. 52, July - September 2004, pp. 331 – 357.

[21] M. M. Berry, *A Variable-Step Double-Integration Multi-Step Integrator*. PhD thesis, Virginia Polytechnic Institute and State University, 2004.

[22] M. H. Kaplan, *Modern Spacecraft Dynamics & Control*. John Wiley & Sons, 1976. p. 301.