ODE SOLVERS USING BAND-LIMITED APPROXIMATIONS

G. BEYLKIN* AND K. SANDBERG**

ABSTRACT. We use generalized Gaussian quadratures for exponentials to develop a new ODE solver. Nodes and weights of these quadratures are computed for a given bandlimit c and user selected accuracy ϵ, so that they integrate functions e^{ibx}, for all $|b| \leq c$, with accuracy ϵ. Nodes of these quadratures do not concentrate excessively near the end points of an interval as those of the standard, polynomial-based Gaussian quadratures. Due to this property, the usual implicit Runge-Kutta (IRK) collocation method may be used with a large number of nodes, as long as the method chosen for solving the nonlinear system of equations converges. We show that the resulting ODE solver is symplectic and demonstrate (numerically) that it is A-stable. We use this solver, dubbed Band-limited Collocation (BLC-IRK), for orbit computations in astrodynamics. Since BLC-IRK minimizes the number of nodes needed to obtain the solution, in this problem we achieve speed close to that of the traditional explicit multistep methods.

1. INTRODUCTION

Standard methods for solving ODEs, be that multistep or Runge-Kutta, are based on polynomial approximations of functions. However, both classical and recent results [35, 22, 34, 36, 2, 3, 4, 27] indicate that in many situations band-limited functions provide a better tool for numerical integration and interpolation of functions than the traditional polynomials. We construct a new method for solving the initial value problem for Ordinary Differential Equations (ODEs) using band-limited approximations and demonstrate certain advantages of such approach. As an example, we consider orbit computations in astrodynamics as a practical application for the new ODE solver as well as a gauge to ascertain its performance.

Using approximations of functions via exponentials instead of polynomials have been considered in e.g., [7, 12, 26] and, recently, in e.g., [17, 18, 13, 20, 21], where in [13, 20, 21] authors use band-limited exponentials with complex-valued exponents. In these papers the nodes are chosen to be equally spaced and, thus, such methods may be viewed as band-limited analogues of multistep schemes. As explained below, our method, dubbed Band-limited Collocation Implicit Runge-Kutta (BLC-IRK), uses unequally spaced nodes and is different from the earlier approaches as in e.g., [1, 8].

It is well-known that choosing between equally spaced and unequally spaced nodes on a specified time interval results in significantly different properties of ODE solvers. For example, polynomial-based multistep schemes have $\{\mathcal{R}e(z) \leq 0, \ z \in \mathbb{C}\}$ as the region of absolute stability (A-stable) only if their order does not exceed 2, the so-called Dahlquist barrier. In contrast, an A-stable implicit Runge-Kutta (IRK) scheme may be of arbitrary order. A class of A-stable IRK schemes uses the Gauss-Legendre quadrature nodes on each time interval and the order of such methods is 2ν, where ν is the number of nodes (see, e.g., [16]). A-stability assures that growth and decay of numerical solutions exactly mimics that of the analytic solutions of the test problem which, in turn, implies that the choice of step size involves only accuracy consideration.

Another property of interest, that of preservation of volume in the phase space, identifies symplectic integrators. Symplectic integrators preserve a particular conserved quantity of Hamiltonian systems as well as an approximate Hamiltonian. In problems of orbit determination, a symplectic integrator would maintain the correct orbit more or less indefinitely with the error accumulating only in a position along that orbit, thus closely reproducing a particular behavior of analytic solutions of nonlinear Hamiltonian systems. The IRK schemes which use the Gauss-Legendre nodes are symplectic (see, e.g., [16]).

While IRK schemes with the Gauss-Legendre nodes provide an excellent discretization of a system of ODEs, using many such nodes on a specified time interval is not practical. The nodes of the Gauss-Legendre quadratures (as well as any other polynomial based Gaussian quadratures) accumulate rapidly towards the end points of an interval. A heuristic reason for such accumulation is that these quadratures have to account for a possible rapid growth of polynomials near the boundary. If functions being approximated do not exhibit such behavior, then such concentration of nodes may not be needed. However, using equally spaced nodes as in the standard multistep methods, leads to less than ideal properties of the resulting schemes as mentioned above.

In this paper we demonstrate that, within IRK collocation methods, quadratures based on polynomials may be replaced by quadratures for band-limited exponentials. The nodes of these quadratures do not accumulate significantly toward the end points of an interval (a heuristic reason for an improved arrangement of nodes is that the exponentials do not grow anywhere within the interval). Our method addresses numerical integration of ODEs whose solutions are well approximated by band-limited exponentials. We note that band-limited exponentials have been successfully used in problems of wave propagation [3] (see also [29, 20]), where it is natural to approximate solutions by band-limited functions.

Unlike the classical Gaussian quadratures for polynomials that integrate exactly a subspace of polynomials up to a fixed degree, the Gaussian type quadratures for exponentials use a finite set of nodes to integrate an infinite set of functions, namely, $\left\{ e^{ibx} \right\}_{|b| \leq c}$ on the interval $|x| \leq 1$. As there is no way to accomplish this exactly, these quadratures are constructed so that all exponentials with $|b| \leq c$ are integrated with accuracy of at least ϵ, where ϵ is arbitrarily small but finite. Such quadratures were constructed in [2] and, via a different approach in [36] (see also [28]). As observed in [3], quadrature nodes of this type do not concentrate excessively toward the end of the interval. The density of nodes increases toward the end points of the interval only by a factor that depends on the desired accuracy ϵ but not on the overall number of nodes.

Using quadratures to integrate band-limited exponentials with bandlimit $2c$ and accuracy ϵ^2, we naturally arrive at a method for interpolation of functions with bandlimit c and accuracy ϵ (see [36, 2]). It turns out that the nodes and weights of quadratures to interpolate with accuracy $\epsilon \approx 10^{-15}$ can, in fact, be constructed using only the standard double precision machine arithmetic. However, generating the integration matrix for the new double precision BLC-IRK method requires using quadruple precision in the intermediate calculations. Importantly, once generated, the quadratures and the integration matrix are applied using only the standard double precision.

While analytically the classical Gauss-Legendre quadratures for polynomials are exact, in practice their accuracy is limited by the machine precision. By choosing (interpolation) accuracy $\epsilon \approx 10^{-15}$, our integrator is effectively "exact" within the double precision of machine arithmetic. Remarkably, using a particular construction of the integration matrix, we show that BLC-IRK method is (exactly) symplectic and, with high accuracy, A-stable. These results were unexpected and

3

indicate that properties of approximate quadratures for band-limited exponentials need to be explored further.

While IRK schemes require solving a system of nonlinear equations at each time step, it does not automatically imply that such schemes are always computationally more expensive than explicit schemes. In the environment where the cost of function evaluation is high, the balance between the necessary iteration with fewer nodes of an implicit scheme vs significantly greater number of nodes of an explicit scheme (but no iteration), may tilt towards an implicit scheme. In problems of astrodynamics, we use an additional observation that most iterations can be performed with an inexpensive (low fidelity) gravity model, making implicit schemes with a large number of nodes per time interval practical. We select the problem of orbit determination as an example where our approach is competitive with numerical schemes that are currently in use (see [5, 6]). We take advantage of the reduced number of function calls to the full gravity model in a way that appears difficult to replicate using alternative schemes.

In order to accelerate solving a system of nonlinear equations, we modify the scheme by explicitly exponentiating the linear part of the force term. For the problem of orbit computations this modification accelerates convergence of iterations by (effectively) making use of the fact that the system is of the second order. So far we did not study possible acceleration of iterations using spectral deferred correction approach as in [11, 24, 15, 19].

We start by providing background information on quadratures for band-limited functions in Section 2. We then describe BLC-IRK method in Section 3 (with some details deferred to Appendix). In Section 4 we detail our algorithm and provide examples.

2. Preliminaries: quadratures for band-limited functions

2.1. **Band-limited functions as a replacement of polynomials.** The quadratures constructed in [36, 2, 28] break with the conventional approach of using polynomials as the fundamental tool in analysis and computation. The approach based on polynomial approximations has a long tradition and leads to such notions as the order of convergence of numerical schemes, polynomial based interpolation, and so on. Recently, an alternative to polynomial approximations has been developed; it turns out that constructing quadratures for band-limited functions, e.g., exponentials e^{ibx}, with $|b| \leq c$, where c is the bandlimit,

in many cases leads to significant improvement in performance of algorithms for interpolation, estimation and solving partial differential equations [3, 29, 20].

2.2. **Bases for band-limited functions.** It is well-known that a function whose Fourier Transform has compact support can not have compact support itself unless it is identically zero. On the other hand, in physics duration of all signals is finite and their frequency response for all practical purposes is also band-limited. Thus, it is important to identify classes of functions which are essentially time and frequency limited. Towards this end, it is natural to analyze an operator whose effect on a function is to truncate it both in the original and the Fourier domains. Indeed, this has been the topic of a series of seminal papers by Slepian, Landau and Pollak, [35, 22, 23, 31, 32, 33, 34], where they observed (*inter alia*) that the eigenfunctions of such operator (see (2.2) below) are the Prolate Spheroidal Wave Functions (PSWFs) of classical Mathematical Physics.

While periodic band-limited functions may be expanded into Fourier series, neither the Fourier series nor the Fourier integral may be used efficiently for non-periodic functions on *intervals*. This motivates us to consider a class of functions (not necessarily periodic) admitting a representation via exponentials $\left\{e^{ibx}\right\}_{|b|\leq c}$, $x \in [-1,1]$, with a fixed parameter c (bandlimit). Following [2], let us consider the linear space of functions

$$\mathcal{E}_c = \left\{ u \in L^\infty([-1,1]) \ \mid \ u(x) = \sum_{k\in\mathbb{Z}} a_k e^{icb_k x} : \{a_k\}_{k\in\mathbb{Z}} \in l^1, \ b_k \in [-1,1] \right\}.$$

Given a finite accuracy ϵ, we represent the functions in $\mathcal{E}_c$ by a fixed set of exponentials $\{e^{ic\tau_k x}\}_{k=1}^M$, where M is as small as possible. It turns out that by finding quadrature nodes $\{\tau_k\}_{k=1}^M$ and weights $\{w_k\}_{k=1}^M$ for exponentials with bandlimit $2c$ and accuracy ϵ^2, we in fact obtain (with accuracy ϵ) a basis for $\mathcal{E}_c$ with bandlimit c [2].

The generalized Gaussian quadratures for exponentials are constructed in [2] (see [36] and [28] for different constructions), which we summarize as

Lemma 1. *For $c > 0$ and any $\epsilon > 0$, there exist nodes $-1 < \tau_1 < \tau_2 < \cdots < \tau_M < 1$ and corresponding weights $w_k > 0$, such that for any $x \in [-1,1]$,*

$$(2.1) \qquad \left| \int_{-1}^1 e^{ictx} \ dt - \sum_{k=1}^M w_k e^{ic\tau_k x} \right| < \epsilon,$$

5

where the number of nodes, $M = c/\pi + O\left(\log c\right)$, is (nearly) optimal. The nodes and weights maintain the natural symmetry, $\tau_k = -\tau_{M-k+1}$ and $w_k = w_{M-k+1}$.

Remark 2. The construction in [2] is more general and yields quadratures for band-limited exponentials integrated with a weight function. If the weight function is 1 as in Lemma 1, then the approach in [2] identifies the nodes of the generalized Gaussian quadratures in (2.1) as zeros of the *Discrete* Prolate Spheroidal Wave Functions (DPSWFs) [33], corresponding to small eigenvalues.

Next we consider band-limited functions,

$$\mathcal{B}_c = \{f \in L^2(\mathbb{R}) \mid \hat{f}(\omega) = 0 \text{ for } |\omega| \geq c\},$$

and briefly summarize some of the results in [35, 22, 23, 31, 34]. Let us define the operator $F_c: L^2[-1,1] \rightarrow L^2[-1,1]$,

$$(2.2) \qquad F_c(\psi)(\omega) = \int_{-1}^{1} e^{icx\omega}\psi(x)dx,$$

where $c > 0$ is the bandlimit. We also consider the operator $Q_c = \frac{c}{2\pi}F_c^*F_c$,

$$(2.3) \qquad Q_c(\psi)(y) = \frac{1}{\pi}\int_{-1}^{1}\frac{\sin(c(y-x))}{y-x}\psi(x)\ dx.$$

The eigenfunctions $\psi_0^c, \psi_1^c, \psi_2^c, \cdots$ of Q_c coincide with those of F_c, and the eigenvalues μ_j of Q_c are related to the eigenvalues λ_j of F_c as

$$(2.4) \qquad \mu_j = \frac{c}{2\pi}|\lambda_j|^2, \quad j = 0, 1, 2, \ldots.$$

While all $\mu_j < 1$, $j = 0, 1, \ldots$, for large c the first approximately $2c/\pi$ eigenvalues μ_j are close to 1. They are followed by $\mathcal{O}(\log c)$ eigenvalues which decay exponentially fast forming a transition region; the rest of the eigenvalues μ_j are very close to zero.

The key result in [35] states that there exists a strictly increasing sequence of real numbers $\gamma_0 < \gamma_1 \ldots$, such that ψ_j^c are eigenfunctions of the differential operator,

$$(2.5) \quad L_c\psi_j^c \equiv \left(-(1-x^2)\frac{d^2}{dx^2} + 2x\frac{d}{dx} + c^2x^2\right)\psi_j^c(x) = \gamma_j\psi_j^c(x).$$

The eigenfunctions of L_c have been known as the angular Prolate Spheroidal Wave Functions (PSWF) before the connection with (2.2) was discovered in [35] by demonstrating that L_c and F_c commute. We note that if $c \rightarrow 0$, then it follows from (2.5) that, in this limit, ψ_j^c become the Legendre polynomials. In many respects, PSWFs are strikingly

similar to orthogonal polynomials; they are orthonormal, constitute a Chebychev system, and admit a version of Gaussian quadratures [36].

Since the space $\mathcal{E}_c$ is dense in $\mathcal{B}_c$ (and vice versa) [2], we note that the quadratures in [36] may potentially be used for the purposes of this paper as well (the nodes of the quadratures in [36] and those used in this paper are close but are not identical). Importantly, given accuracy ϵ, the functions $\psi_0^c, \psi_1^c, \psi_2^c, \cdots, \psi_{M-1}^c$ may be used as a basis for interpolation on the interval $[-1, 1]$ with $\tau_1, \tau_2, \cdots, \tau_M$ as the interpolation nodes, provided that these are quadrature nodes constructed for the bandlimit $2c$ and accuracy ϵ^2. Given functions $\psi_0^c, \psi_1^c, \psi_2^c, \cdots, \psi_{M-1}^c$, we can construct an analogue of the Lagrange interpolating polynomials, $R_k^c(x) = \sum_{j=0}^{M-1} \alpha_{kj} \psi_j^c(x)$, $x \in [-1, 1]$, by solving

$$(2.6) \qquad \delta_{kl} = R_k^c(\tau_l) = \sum_{j=0}^{M-1} \alpha_{kj} \psi_j^c(\tau_l)$$

for the coefficients α_{kj}. The matrix $\psi_j^c(\tau_l)$ in (2.6) is well conditioned.

A well-known problem associated with the numerical use of orthogonal polynomials is concentration of their roots near the ends of the interval. Let us consider the ratio

$$(2.7) \qquad r(M, \epsilon) = \frac{\tau_2 - \tau_1}{\tau_{\lfloor M/2 \rfloor} - \tau_{\lfloor M/2 \rfloor - 1}},$$

where "$\lfloor M/2 \rfloor$" denotes the least integer part, and look at it as a function of M. Observing that the distance between nodes of Gaussian quadratures for exponentials changes monotonically from the middle of an interval toward its end points, and that the smallest distance occurs between the nodes closest to the end point, the ratio (2.7) may be used as a measure of node accumulation. For example, the distance between the nodes near the end points of the standard Gaussian quadratures for polynomials decreases as $\mathcal{O}(1/M^2)$, so that we have $r(M, \epsilon) = \mathcal{O}(1/M)$, where M is the number of nodes. In Figure 2.1 we illustrate the behavior of $r(M, \epsilon)$ for the nodes of quadratures for bandlimited exponentials. This ratio approaches a constant that depends on the accuracy ϵ but does not depend on the number of nodes.

Another important property of quadratures for exponentials emerges if we compare the critical sampling rate of a smooth *periodic* function, to that of smooth *non-periodic* function defined on an interval. Considering bandlimit c as a function of the number of nodes, M, and the desired accuracy ϵ, we observe that the oversampling factor,

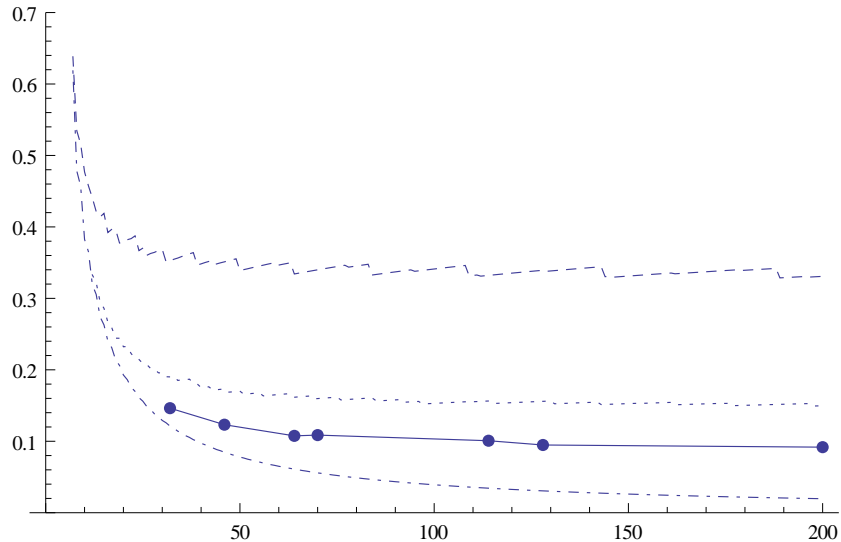$$\alpha(M, \epsilon) = \frac{\pi M}{c(M, \epsilon)} > 1,$$

7

FIGURE 2.1. The ratio $r(M, \epsilon)$ in (2.7) as a function of the number of nodes M and interpolation accuracy $\epsilon \approx 10^{-3.5}$ (top curve, dashed), $\epsilon \approx 10^{-8.5}$ (middle curve, dotted) and $\epsilon \approx 10^{-13}$ (middle curve, solid). The dots on the solid curve indicate the number of nodes of quadratures used in our numerical experiments. The bottom curve shows this ratio for the Gauss-Legendre nodes.

approaches 1 for large M. We recall that in the case of the Gaussian quadratures for polynomials, this oversampling factor approaches $\frac{\pi}{2}$ rather than 1 (see e.g. [14]).

2.3. **Interpolating bases for band-limited functions.** A basis of interpolating band-limited functions for the bandlimit c and accuracy ϵ plays the same role in the derivation of a system of nonlinear equations for solving ODEs as the bases of Lagrange interpolating polynomials defined on the Gauss-Legendre nodes. While (2.6) relies on available solutions of the differential equation (2.5), interpolating basis functions may also be obtained by solving the integral equation (2.2) (see [2, 28]).

We start by first constructing a quadrature for the bandlimit $2c > 0$ and accuracy threshold $\epsilon^2 > 0$, yielding M nodes $\{\tau_m\}_{m=1}^M$ and weights $\{w_m\}_{m=1}^M$. For the inner product of two functions $f, g \in \mathcal{E}_c$, we have

$$\left| \int_{-1}^1 f(t)g(t)dt - \sum_{m=1}^M w_m f(\tau_m)g(\tau_m) \right| \leq \epsilon^2.$$

Following [2], we discretize (2.2) using nodes $\{\tau_m\}_{m=1}^{M}$ and weights $\{w_m\}_{m=1}^{M}$ and obtain an algebraic eigenvalue problem,

$$(2.8) \qquad \sum_{l=1}^{M} w_l e^{ic\tau_m\tau_l} \mathbf{\Psi}_j(\tau_l) = \eta_j \mathbf{\Psi}_j(\tau_m).$$

The approximate PSWFs on $[-1, 1]$ are then defined consistent with (2.2) as

$$(2.9) \qquad \Psi_j(x) = \frac{1}{\eta_j} \sum_{l=1}^{M} w_l e^{icx\tau_l} \mathbf{\Psi}_j(\tau_l),$$

where η_j are the eigenvalues and $\mathbf{\Psi}_j(\tau_l)$ the eigenvectors in (2.8). Following [2], we then define the interpolating basis for band-limited functions as

$$(2.10) \qquad R_k(x) = \sum_{l=1}^{M} r_{kl} e^{ic\tau_l x}, \quad k = 1, \dots, M,$$

where

$$(2.11) \qquad r_{kl} = \sum_{j=1}^{M} w_k \mathbf{\Psi}_j(\tau_k) \frac{1}{\eta_j} \mathbf{\Psi}_j(\tau_l) w_l.$$

It is shown in [2] that the functions $R_k(x)$ are interpolating, $R_k(\tau_l) = \delta_{kl}$.

3. BLC-IRK method

3.1. **Discretization of Picard integral equation.** We consider the initial value problem for a system of ODEs,

$$\mathbf{y}' = \mathbf{f}(t, \mathbf{y}), \quad \mathbf{y}(0) = \mathbf{y_0},$$

or, equivalently,

$$(3.1) \qquad \mathbf{y}(t) = \mathbf{y_0} + \int_0^t \mathbf{f}(s, \mathbf{y}(s)) \, ds.$$

It is sufficient to discretize (3.1) on the interval $[0, t]$ since, by shifting the time variable, the initial condition may always be set at $t = 0$. We require

$$\mathbf{y}'(t\tau_j) = \mathbf{f}(t\tau_j, \mathbf{y}(t\tau_j)), \quad j = 1, \dots, M,$$

9

where $\{\tau_j\}_{j=1}^M$ are Gaussian nodes for band-limited exponentials on $[0, 1]$ (constructed for an appropriate bandlimit c and accuracy ϵ). We approximate

$$(3.2) \qquad \left\| \mathbf{f}(t\tau, \mathbf{y}(t\tau)) - \sum_{j=1}^M \mathbf{f}(t\tau_j, \mathbf{y}(t\tau_j)) R_j(\tau) \right\| \le \epsilon, \quad \tau \in [0, 1]$$

where $R_j(\tau)$ are interpolating basis functions associated with these quadratures and briefly described in Section 2.3 (see [2, 3] for details). Using (3.2), we replace $\mathbf{f}$ in (3.1) and evaluate $\mathbf{y}(t\tau)$ at the quadrature nodes yielding a nonlinear system,

$$
\begin{aligned}
(3.3) \qquad \mathbf{y}(t\tau_k) &= \mathbf{y_0} + t \sum_{j=1}^M \mathbf{f}(t\tau_j, \mathbf{y}(t\tau_j)) \int_0^{\tau_k} R_j(s) ds \\
&= \mathbf{y_0} + t \sum_{j=1}^M S_{kj} \mathbf{f}(t\tau_j, \mathbf{y}(t\tau_j)),
\end{aligned}
$$

where $S_{kj} = \int_0^{\tau_k} R_j(s) ds$ is the integration matrix and $k = 1, \dots M$. After solving for $\{\mathbf{y}(t\tau_j)\}_{j=1}^M$, we have from (3.1)

$$(3.4) \qquad \mathbf{y}(t) = \mathbf{y_0} + t \sum_{j=1}^M w_j \mathbf{f}(t\tau_j, \mathbf{y}(t\tau_j)),$$

where $\{w_j\}_{j=1}^M$ are the quadrature weights. The result is an implicit Runge-Kutta method (IRK) where the usual Gauss-Legendre quadratures are replaced by Gaussian quadratures for band-limited exponentials.

The nodes, weights, and the entries of the integration matrix are typically organized in the Butcher tableau,

$$\begin{array}{c|c} \tau & S \\ \hline & w^t \end{array}.$$

Unlike in the standard IRK method based on Gauss-Legendre quadratures, we solve (3.3) on a time interval containing a large number of quadrature nodes, since these nodes do not concentrate excessively near the end points. This implies that the interval $[0, t]$ may be selected to be large in comparison with the usual choices in RK methods.

3.2. **Exact Linear Part.** In many problems (including that of orbit computations in astrodynamics), the right hand side of the ODE, $\mathbf{f}(t, \mathbf{y})$, may be split into a linear and nonlinear part,

$$\mathbf{f}(t, \mathbf{y}(t)) = \mathbf{L}\mathbf{y}(t) + \mathbf{g}(t, \mathbf{y}(t)),$$

10

so that the integral equation (3.1) may be written as

$$(3.5) \qquad \mathbf{y}(t) = e^{tL}\mathbf{y_0} + \int_0^t e^{(t-s)\mathbf{L}}\mathbf{g}(s, \mathbf{y}(s)) \ ds.$$

If the operator $e^{t\mathbf{L}}$ can be computed efficiently, this formulation leads to savings when solving the integral equation iteratively.

We discretize (3.5) by using (3.2) and obtain

$$
\begin{aligned}
\mathbf{y}(t\tau_k) \ &= \ e^{t\tau_k \mathbf{L}}\mathbf{y_0} + t \sum_{j=1}^{M} e^{t(\tau_k - \tau_j)\mathbf{L}}\mathbf{g}(t\tau_j, \mathbf{y}(t\tau_j)) \int_0^{\tau_k} R_j(s) ds \\
(3.6) \qquad &= \ e^{t\tau_k \mathbf{L}}\mathbf{y_0} + t \sum_{j=1}^{M} S_{kj} e^{t(\tau_k - \tau_j)\mathbf{L}}\mathbf{g}(t\tau_j, \mathbf{y}(t\tau_j))
\end{aligned}
$$

where $S_{kj} = \int_0^{\tau_k} R_j(s)ds$. We note that (3.3) is a special case of (3.6) with $\mathbf{L} = 0$ and $\mathbf{g} = \mathbf{f}$.

3.3. **Symplectic integrators.** Following [30], let us introduce matrix $\mathcal{M} = \{m_{kj}\}_{k,j=1}^{M}$ for an M-stage IRK scheme,

$$(3.7) \qquad m_{kj} = w_k S_{kj} + w_j S_{jk} - w_k w_j,$$

where the weights $w = \{w_k\}_{k=1}^{M}$ and the integration matrix $S = \{S_{kj}\}_{k,j=1}^{M}$ define the Butcher's tableau for the method.

It is shown in [30] that

Theorem 3. *If matrix $\mathcal{M} = 0$ in (3.7), then an M-stage IRK scheme is symplectic.*

This condition, $\mathcal{M} = 0$, is satisfied for the Gauss-Legendre RK methods, see e.g. [9, 30]. We enforce this condition for BLC-IRK method by an $\mathcal{O}(\epsilon^2)$ modification of the weights and of the integration matrix. For convenience, in what follows, we consider the band-limited exponentials and integration matrix on the interval $[-1, 1]$ rather than on the interval $[0, 1]$ usually used for ODEs.

Proposition 4. *Let $\{\tau_j\}_{j=1}^{M}$ and $\{w_j\}_{j=1}^{M}$ be quadrature nodes and weights for the bandlimit $2c$ and accuracy ϵ^2. Consider interpolating basis functions on these quadrature nodes, $R_k(\tau)$, $R_k(\tau_j) = \delta_{kj}$, $k, j = 1, \ldots, M$, and define $F_k(\tau) = \int_{-1}^{\tau} R_k(s) \ ds$. Then we have*

$$(3.8) \qquad \left| \int_{-1}^{1} F_j(\tau) F_k'(\tau) \ d\tau - \sum_{l=1}^{M} w_l F_j(\tau_l) F_k'(\tau_l) \right| < \epsilon^2$$

11

or

$$\left| \int_{-1}^{1} \left(\int_{-1}^{\tau} R_j(s)\ ds \right) R_k(\tau)\ d\tau - w_k \int_{-1}^{\tau_k} R_j(s)\ ds \right| < \epsilon^2,$$

and

(3.9)
$$\left| \int_{-1}^{1} R_k(\tau)\ d\tau - \sum_{l=1}^{M} w_l R_k(\tau_l) \right| < \epsilon^2,$$

or

$$\left| \int_{-1}^{1} R_k(s)\ ds - w_k \right| < \epsilon^2.$$

Proof. The relations in (3.8) and (3.9) is the property of the quadrature, since the bandlimit of the product $F_j(\tau)F_k'(\tau)$ is less or equal to $2c$ and that of $R_k(\tau)$ is less or equal to c. Due to the interpolating property of $R_k(\tau)$, we have

(3.10)
$$\sum_{l=1}^{M} w_l F_j(\tau_l) F_k'(\tau_l) = \sum_{l=1}^{M} \left(\int_{-1}^{\tau_l} R_j(s)\ ds \right) w_l R_k(\tau_l) = w_k \int_{-1}^{\tau_k} R_j(s)\ ds$$

and

$$\sum_{l=1}^{M} w_l R_k(\tau_l) = w_k$$

Also, by definition,

$$\int_{-1}^{1} F_j(\tau) F_k'(\tau)\ d\tau = \int_{-1}^{1} \left(\int_{0}^{\tau} R_j(s)\ ds \right) R_k(\tau)\ d\tau,$$

and the result follows. $\qquad\square$

Theorem 5. *Let $\{\tau_j\}_{j=1}^{M}$ be quadrature nodes of the quadrature for the bandlimit $2c$ and accuracy ϵ^2 and $R_k(\tau)$, $R_k(\tau_j) = \delta_{kj}$, $k, j = 1, \ldots, M$, the corresponding interpolating basis. Let us define weights for the quadrature as*

(3.11)
$$w_k = \int_{-1}^{1} R_k(\tau) d\tau$$

and the integration matrix as

(3.12)
$$S_{kj} = \frac{\int_{-1}^{1} \left(\int_{-1}^{\tau} R_j(s)\ ds \right) R_k(\tau)\ d\tau}{w_k}, \quad k, j = 1, \ldots, M.$$

Then

(3.13)
$$w_k S_{kj} + w_j S_{jk} - w_k w_j = 0,$$

and the implicit scheme using these nodes and weights is symplectic.

12

Proof. Using Proposition 4, we observe that the weights defined in (3.11) are the same (up to accuracy ϵ^2) as those of the quadrature. The result follows by setting $F_k(\tau) = \int_{-1}^{\tau} R_k(\tau) \, d\tau$, $F_k'(\tau) = R_k(\tau)$ and integrating by parts to obtain

$$w_k S_{kj} + w_j S_{jk} - w_k w_j = \int_{-1}^{1} F_j(\tau) F_k'(\tau) \, d\tau + \int_{-1}^{1} F_k(\tau) F_j'(\tau) \, d\tau - w_k w_j$$

$$= F_j(1) F_k(1) - w_k w_j.$$

By the definition of the weights, we have $F_k(1) = w_k$ and, hence, $F_j(1) F_k(1) - w_k w_j = 0$. $\qquad\square$

3.4. **Construction of the integration matrix.** There are at least three approaches to compute the integration matrix. Two of them, presented in the Appendix, rely on Theorem 5 and differ in the construction of interpolating basis functions. In what appears to be a simpler approach, the integration matrix may also be obtained without computing interpolating basis functions explicitly and, instead, using a collocation condition derived below together with the symplectic condition (3.13).

We require that our method accurately solves the test problems

$$y' = ic\tau_m y, \quad y(-1) = e^{-ic\tau_m}, \quad m = 1, \ldots, M,$$

on the interval $[-1, 1]$, where τ_m are the nodes of the quadrature. Specifically, given solutions of these test problems, $y_m(t) = e^{ic\tau_m t}$, we require that (3.3) holds at the nodes $t = \tau_k$ with accuracy ϵ,

$$(3.14) \qquad \left| \frac{e^{ic\tau_m \tau_k} - e^{-ic\tau_m}}{ic\tau_m} - \sum_{j=1}^{M} S_{kj} e^{ic\tau_m \tau_j} \right| \leq \epsilon, \quad m, k = 1, \ldots, M.$$

We then obtain the integration matrix as the solution of (3.13) satisfying an approximate collocation condition (3.14).

We proceed by observing that (3.13) suggests that the integration matrix can be split into symmetric and antisymmetric part. Defining the symmetric part of the integration matrix as

$$(3.15) \qquad\qquad T_{kj} = \frac{w_k w_j}{w_k + w_j},$$

we set

$$(3.16) \qquad\qquad S_{kj} = T_{kj} + A_{kj} w_j,$$

and observe that it follows from (3.13) that A_{kj} is antisymmetric,

$$A_{kj} + A_{jk} = 0.$$

13

Using (3.16) and casting (3.14) as an equality, we obtain equations for the matrix entries A_{kj},

(3.17)
$$\sum_{j=1}^{M} A_{kj} w_j e^{ic\tau_m\tau_j} = \frac{e^{ic\tau_m\tau_k} - e^{-ic\tau_m}}{ic\tau_m} - \sum_{j=1}^{M} T_{kj} e^{ic\tau_m\tau_j}, \quad m, k = 1, \ldots, M.$$

Splitting the real and imaginary parts of the right hand side,

$$\frac{e^{ic\tau_m\tau_k} - e^{-ic\tau_m}}{ic\tau_m} - \sum_{j=1}^{M} T_{kj} e^{ic\tau_m\tau_j} = u_{km} + iv_{km},$$

we obtain

$$u_{km} = (\tau_k + 1) \operatorname{sinc} (c\tau_m(\tau_k + 1)/2) \cos (c\tau_m(\tau_k - 1)/2) - \sum_{j=1}^{M} T_{kj} \cos (c\tau_m\tau_j)$$

and, since $T_{kj} = T_{k(M-j+1)}$ due to the symmetry of the weights, we arrive at

$$v_{km} = (\tau_k + 1) \operatorname{sinc} (c\tau_m(\tau_k + 1)/2) \sin (c\tau_m(\tau_k - 1)/2) .$$

We also have

$$u_{km} = \sum_{j=1}^{M} A_{kj} w_j \cos (c\tau_m\tau_j) , \quad v_{km} = \sum_{j=1}^{M} A_{kj} w_j \sin (c\tau_m\tau_j) .$$

Since matrices $\cos (c\tau_m\tau_j)$ and $\sin (c\tau_m\tau_j)$ are rank deficient, we choose to combine these equations

(3.18)
$$u_{km} + v_{km} = \sum_{j=1}^{M} A_{kj} w_j (\cos (c\tau_m\tau_j) + \sin (c\tau_m\tau_j)) .$$

The number of unknowns in (3.18) is $M(M - 1)/2$ since the matrix A is antisymmetric. Instead of imposing additional conditions due to antisymmetry of A, we proceed by solving (3.18) using quadruple precision (since this system is ill-conditioned). We find matrix $\tilde{A}$ and discover that, while $S_{kj} = T_{kj} + \tilde{A}_{kj} w_j$ makes (3.14) into an equality, the matrix $\tilde{A}$ is not antisymmetric. We then enforce anti-symmetry by setting $A_{kj} = -A_{jk} = \left(\tilde{A}_{kj} - \tilde{A}_{jk} \right) /2$ and $S_{kj} = T_{kj} + A_{kj} w_j$. We then verify that the matrix S satisfies the inequality (3.14).

Remark 6. The fact that integration matrix satisfies (3.13) and the inequality (3.14) indicates that, perhaps by a slight modification of nodes and weights of the quadrature, it might be possible to satisfy (3.13) and (3.14) with $\epsilon = 0$.

3.5. A-stability of the BLC-IRK method.

As shown in e.g. [16, Section 4.3], in order to ascertain stability of an IRK method, it is sufficient to consider the rational function

$$(3.19) \qquad r(z) = 1 + z\mathbf{w}^t(I - zS)^{-1}\mathbf{1},$$

where S is the integration matrix, $\mathbf{w}$ is a vector of weights and $\mathbf{1}$ is a vector with all entries set to 1, and verify that $|r(z)| \leq 1$ in the left half of the complex plane, $\mathcal{R}e\,(z) \leq 0$. This function is an approximation of the solution e^{zt} at $t = 1$ of the test problem

$$y' = zy, \quad y(0) = 1$$

computed via (3.3) and (3.4) on the interval $[0, 1]$. If all poles of $r(z)$ have a positive real part, then it is sufficient to verify this inequality only on the imaginary axis, $z = iy$, $y \in \mathbb{R}$. In fact, it may be possible to show that $r(z)$ is unimodular on imaginary axis, $|r(iy)| = 1$, for $y \in \mathbb{R}$. Implicit Runge-Kutta methods based on Gauss-Legendre nodes are A-stable (see e.g [16]) and, indeed, for these methods $r(z)$ is unimodular on imaginary axis.

Given an $M \times M$ matrix S with M_1 complex eigenvalues and M_2 real eigenvalues implies that the function $r(z)$ in (3.19) has $2M_1 + M_2 = M$ poles. If this function is unimodular on the imaginary axis then it is easy to show that it has a particular form,

$$(3.20) \qquad r(z) = \prod_{k=1}^{M_1} \frac{z + \overline{\lambda}_k^{-1}}{z - \lambda_k^{-1}} \frac{z + \lambda_k^{-1}}{z - \overline{\lambda}_k^{-1}} \prod_{k'=1}^{M_2} \frac{z + \lambda_{k'}^{-1}}{z - \lambda_{k'}^{-1}}.$$

Currently, we do not have an analytic proof of A-stability of BLC-IRK method; instead we verify (3.20) numerically. We compute eigenvalues of the integration matrix to obtain the poles of $r(z)$ and check that all eigenvalues have a positive real part separated from zero. For example, the integration matrix for the BLC-IRK method with 64 nodes (bandlimit $c = 17\pi$) has all eigenvalues with real part larger than $0.7 \cdot 10^{-3}$(see Figure 3.1). One way to check that $r(z)$ has the form (3.20) is to compute $r(-\overline{\lambda}_k^{-1})$ for complex valued and $r(-\lambda_k^{-1})$ for real valued eigenvalues in order to observe if these are its zeros. In fact, it is the case with high (quadruple) precision.

One can argue heuristically that since a rational function with M poles has at most $2M$ real parameters (since matrix S is real its eigenvalues appear in complex conjugate pairs) and since, by construction, $r(iy)$ for $|y| \leq c$ is an accurate approximation to e^{iy} (which is obviously unimodular), $r(z)$ is then unimodular. It remains to show it rigorously; a possible proof may depend on demonstrating a conjecture in Remark 6.
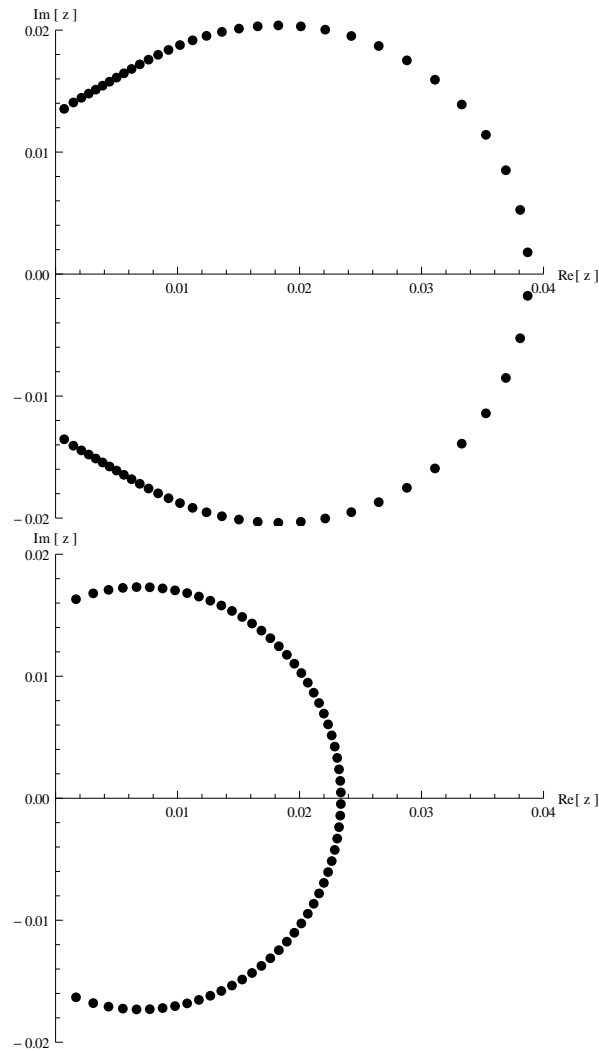
FIGURE 3.1. Eigenvalues (computed using quadruple precision) of the integration matrix for BLC-IRK scheme with 64 nodes corresponding to the bandlimit 17π and, for comparison, eigenvalues of the integration matrix of the standard IRK scheme 64 Gauss-Legendre nodes.

4. APPLICATIONS

4.1. **Algorithm.** We use a (modified) fixed point iteration to solve (3.6). These equations are formulated on a large time interval in comparison with the polynomial-based IRK schemes since we do not have to deal with the excessive concentration of nodes near the end points.

Thus, the only constraint on the size of the interval is the requirement that the (standard) fixed point iteration for (3.6) converges .

Let N_{it} denote the number of iterations, which can either be set to a fixed number or be determined adaptively. Labeling the intermediate solutions in the iteration scheme as $\mathbf{y}^{(n)}$, $n = 1, \ldots, N_{it}$, we have

(1) Initialize $y^{(1)}(t\tau_m) = \mathbf{y_0}$, $m = 1, \ldots, M$.
(2) **For** $n = 1, \ldots, N_{it}$
    For $k = 1, \ldots, M$

   (a) Update the solution at the node k:

$$\mathbf{y}^{(n)}(t\tau_k) = e^{t\tau_k \mathbf{L}}\mathbf{y_0} + t\sum_{j=1}^{M} S_{kj}\, e^{t(\tau_k - \tau_j)\mathbf{L}}\mathbf{g}(t\tau_j, \mathbf{y}^{(n)}(t\tau_j))$$

   (b) Update the right hand side at the node k: $\mathbf{g}(t\tau_k, y^{(n)}(t\tau_k))$

We note that the updated value of $y^{(n)}(t\tau_k)$ is used in the computation at the next node τ_{k+1} within the same iteration n. This modification of the standard fixed point iteration is essential for a faster convergence.

Remark 7. Although we currently apply the integration matrix directly, using a large time interval and, consequently, a large number of nodes per interval, opens a possibility of developing fast algorithms for this purpose. Such algorithms may be faster than the direct application of the matrix only for a sufficiently large matrix size and are typically less efficient than the direct method if the size is relatively small. Since we may choose many nodes, it makes sense to ask if the integration matrix of an BLC-IRK type method may be applied in $\mathcal{O}(M)$ or $\mathcal{O}(M \log M)$ operations rather than $\mathcal{O}(M^2)$. We mention an example of an algorithm for this purpose using the partitioned low rank (PLR) representation (as it was described in e.g., [3]) but leave open a possibility of more efficient approaches.

4.2. **Problem of Orbit Determination.** Let us consider the spherical harmonic model of a gravitational potential of degree N,

$$(4.1) \qquad V^{(N)}(r, \theta, \lambda) = \frac{\mu}{r}\left(1 + \sum_{n=2}^{N}\left(\frac{R}{r}\right)^{-n} Y_n(\theta, \lambda)\right),$$

with

$$(4.2) \qquad Y_n(\theta, \lambda) = \sum_{m=0}^{n} \bar{P}_n^m(\sin \theta)(\bar{C}_{nm}\cos(m\lambda) + \bar{S}_{nm}\sin(m\lambda)),$$

where $\bar{P}_n^m$ are normalized associated Legendre functions and $\bar{C}_{nm}$ and $\bar{S}_{nm}$ are normalized gravitational coefficients. In case of the Earth's

gravitational model, μ is the Earth's gravitational constant and R is chosen to be the Earth's equatorial radius. Choosing the Cartesian coordinates, we write $V^{(N)}(\mathbf{r})$, $\mathbf{r} = (x, y, z)$, assuming that the values $V^{(N)}(\mathbf{r})$ are evaluated via (4.1) by changing from the Cartesian to the spherical coordinates, $r = \sqrt{x^2 + y^2 + z^2}$, $\theta = \arcsin(z/r)$ and $\lambda = \arctan(y/x)$.

We formulate the system of ODEs in the Cartesian coordinates and denote the solution as $\mathbf{r}(t) = (x(t), y(t), z(t))$. Setting $\mathbf{G}^{(N)}(\mathbf{r}) = \nabla V^{(N)}(\mathbf{r})$, we consider the initial value problem

(4.3)

$$\frac{d^2}{dt^2}\mathbf{r}(t) = -\mathbf{G}^{(N)}(\mathbf{r}(t)), \quad \mathbf{r}(0) = \mathbf{r}_0 = \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix}, \quad \mathbf{r}'(0) = \mathbf{v}_0 = \begin{pmatrix} x_0' \\ y_0' \\ z_0' \end{pmatrix}.$$

We observe that the first few terms of the Earth's gravitational models are large in comparison with the rest of the model terms. For example, in EGM96 [25], the only non-zero coefficients for $Y_2(\theta, \lambda)$ are $\bar{C}_{20}$, $\bar{C}_{22}$ and $\bar{S}_{22}$, where $\bar{C}_{20} \approx -0.48 \cdot 10^{-3}$, $\bar{C}_{22} \approx 0.24 \cdot 10^{-5}$, and $\bar{S}_{22} \approx -0.14 \cdot 10^{-5}$, whereas the coefficients of the terms $Y_n(\theta, \lambda)$ with $n \geq 3$ are less than $0.14 \cdot 10^{-5}$. For this reason it makes sense to split the force as

$$\mathbf{G}^{(N)}(\mathbf{r}) = \mathbf{G}^{(2)}(\mathbf{r}) + \left(\mathbf{G}^{(N)}(\mathbf{r}) - \mathbf{G}^{(2)}(\mathbf{r})\right)$$

and use only $\mathbf{G}^{(2)}(\mathbf{r})$ in most of the iterations (since using the full model, $\mathbf{G}^{(N)}(\mathbf{r})$, may be expensive).

We first use the gravity model of degree $N = 2$ on a large portion of an orbit (e.g., $1/2$ of a period) to solve the system of nonlinear equations via fixed point iteration. Once the approximate solution $\tilde{\mathbf{r}}(t)$ to

$$\frac{d^2}{dt^2}\tilde{\mathbf{r}}(t) = -\mathbf{G}^{(2)}(\tilde{\mathbf{r}}(t)), \quad \tilde{\mathbf{r}}(0) = \mathbf{r}_0 = \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix}, \quad \tilde{\mathbf{r}}'(0) = \mathbf{v}_0 = \begin{pmatrix} x_0' \\ y_0' \\ z_0' \end{pmatrix},$$

is obtained, we then access the full gravity model $\mathbf{G}^{(N)}(\tilde{\mathbf{r}}(t\tau_j))$ to evaluate the forces at the nodes τ_j which, by now, are located close to their correct positions. We continue iteration (without accessing the full gravity model again) to adjust the orbit. This results in an essentially correct trajectory. At this point we may (and currently do) access the full gravity model $\mathbf{G}^{(N)}$ one more time to evaluate the gravitational force and perform another iteration. Thus, we access the full gravity model at most twice per node while the number of nodes is substantially lower than in traditional methods.

Next, let us write the orbit determination problem in a form that conforms with the algorithm in Section 4.1. Effectively, we make use of the fact that system (4.3) is of the second order. We define the six component vector

$$\mathbf{u}(t) = \left[\begin{array}{c} \mathbf{r}(t) \\ \mathbf{r}'(t) \end{array} \right] = \left[\begin{array}{c} \mathbf{r}(t) \\ \mathbf{v}(t) \end{array} \right],$$

where $\mathbf{r}'(t) = \mathbf{v}(t)$ is the velocity, and the matrix

$$\mathbf{L} = \left(\begin{array}{cc} \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} \end{array} \right),$$

where $\mathbf{I}$ is 3×3 identity matrix. We have

$$(4.4) \qquad \frac{d}{dt} \left[\begin{array}{c} \mathbf{r}(t) \\ \mathbf{v}(t) \end{array} \right] = \mathbf{L} \left[\begin{array}{c} \mathbf{r}(t) \\ \mathbf{v}(t) \end{array} \right] + \left[\begin{array}{c} \mathbf{0} \\ -\mathbf{G}^{(N)}\left(\mathbf{r}(t)\right) \end{array} \right],$$

and the orbit determination problem is now given by (3.5) with appropriate forces as follow from (4.4). Using (3.5) accelerates convergence of the fixed point iteration in our scheme.

4.3. **Example.** We present an example of using our method. An extensive study of the method for applications in astrodynamics may be found in [6] (see also [5]) and here we simply demonstrate that our scheme allows computations on large time intervals and requires relatively few evaluations of the full gravity model. Since the cost of evaluating the full (high-degree) gravity model is substantial, this results in significant computational savings.

As an example, we simulate an orbit with initial condition

$$\mathbf{r}_{|t=0} = \left(\begin{array}{c} x_0 \\ y_0 \\ z_0 \end{array} \right) = \left(\begin{array}{r} 2284.060 \\ 6275.400 \\ 4.431 \end{array} \right) \text{ (km)}$$

and

$$\frac{d\mathbf{r}}{dt}_{|t=0} = \mathbf{v}_0 = \left(\begin{array}{r} -5.947 \\ 2.164 \\ 0 \end{array} \right) \text{ (km/s),}$$

and propagate it for 86,000 seconds (approximately 1 day). We use 22 time intervals and, on each interval, quadratures with 74 nodes. Hence, on average, this corresponds to time distance between nodes of approximately 53 seconds. For the full gravitational model we use a 70 degree spherical harmonics model WGS84 [10].

Using the 8th-order Gauss-Jackson integration scheme with very fine sampling (one second time step), we generate the reference solution. We selected the Gauss-Jackson method since it is often used for orbit

computations in astrodynamics; we refer to [6, 5] for a more detailed discussion on the issue of generating reference solutions.

We then compute the orbit trajectory using the algorithm from Section 4.1 adopted to the problem of orbit propagation as described in Section 4.2 and compare the result with the reference solution. Achieving an error of less than 5 cm at the final time, we need 6512 evaluations of the reduced (3-term) gravitational model, and 3256 evaluations of the full gravitational model.

5. Conclusions

We have constructed an implicit, symplectic integrator that has speed comparable to explicit multistep integrators currently used for orbit computation. The key difference with the traditional IRK method is that our scheme uses quadratures for band-limited exponentials rather than the traditional Gaussian quadratures constructed for the orthogonal Legendre polynomials. The nodes of quadratures for band-limited exponentials do not concentrate excessively towards the end points of an interval thus removing a practical limit on the number of nodes used within each time interval.

6. Appendix

In both approaches described below we use nodes of generalized Gaussian quadratures for exponentials $\{\tau_l\}_{l=1}^M$ constructed in [2] (see Lemma 1). Some of the steps may require extended precision to yield accurate results.

6.1. **Computing integration matrix using exact PSWFs.** In this approach we assume that the solutions $\psi_j^c(x)$ and the eigenvalues λ_j satisfying

$$(6.1) \qquad \left(F_c\psi_j^c\right)(x) = \int_{-1}^1 e^{icxy}\psi_j^c(y)dy = \lambda_j\psi_j^c(x),$$

where F_c is defined in (2.2), are available. We use (2.6) and the matrix of values of PSWFs at the nodes, $\psi_j^c(\tau_l)$, to compute coefficients α_{kj}, so that we have

$$R_k^c(\tau) = \sum_{j=0}^{M-1} \alpha_{kj}\psi_j^c(\tau), \quad k = 1,\dots M.$$

We then compute weights using (3.11),

$$w_k = \int_{-1}^{1} R_k^c(x)dx = \sum_{j=0}^{M-1} \alpha_{kj} \int_{-1}^{1} \psi_j^c(x)dx = \sum_{j=0}^{M-1} \alpha_{kj}\lambda_j\psi_j^c(0).$$

Next we define

$$K_l^c(x) = \int_{-1}^{x} R_l^c(s)ds = \sum_{j=0}^{M-1} \alpha_{lj} \int_{-1}^{x} \psi_j^c(s)ds = \sum_{j=0}^{M-1} \alpha_{lj}\Phi_j^c(x),$$

where

$$(6.2) \qquad \Phi_j^c(x) = \int_{-1}^{x} \psi_j^c(s)ds.$$

In order to compute the integration matrix (3.12), we need to evaluate

$$w_k S_{kl} = \int_{-1}^{1} K_l^c(x)R_k^c(x)dx = \sum_{j,j'=0}^{M-1} \alpha_{lj}\alpha_{kj'} \int_{-1}^{1} \Phi_j^c(x)\psi_{j'}^c(x)dx = \sum_{j,j'=0}^{M-1} \alpha_{lj}\alpha_{kj'}I_{jj'},$$

where

$$(6.3) \qquad I_{jj'} = \int_{-1}^{1} \Phi_j^c(x)\psi_{j'}^c(x)dx = \int_{-1}^{1} \Phi_j^c(x)\frac{d}{dx}\Phi_{j'}^c(x)dx.$$

We have

Proposition 8. *If j and j' are both even, then*

$$(6.4) \qquad I_{jj'} = I_{j'j} = \frac{1}{2}\lambda_j\lambda_{j'}\psi_j^c(0)\psi_{j'}^c(0).$$

If j and j' are both odd, then

$$(6.5) \qquad I_{jj'} = 0.$$

If j is even and j' is odd, then

$$(6.6) \qquad I_{jj'} = -I_{j'j},$$

$$(6.7) \qquad I_{jj'} = \frac{\lambda_{j'}}{ic\lambda_j} \int_{-1}^{1} \psi_j^c(y)\frac{\psi_{j'}^c(y)}{y}dy$$

and
(6.8)
$$I_{j'j} = \frac{\lambda_j}{ic\lambda_{j'}} \left(\int_{-1}^{1} \psi_j^c(y)\frac{\psi_{j'}^c(y)}{y}dy - 2\psi_j^c(0)\int_0^1 \frac{\psi_{j'}^c(y)}{y}dy + ic\psi_j^c(0)\overline{\lambda}_{j'}\int_0^1 \psi_{j'}^c(y)dy \right).$$

We use (6.7) if $|\lambda_{j'}| < |\lambda_j|$, (6.8) otherwise.

Proof. Integrating (6.3) by parts, we obtain

$$(6.9) \quad I_{jj'} + I_{j'j} = \Phi_j^c(1)\Phi_{j'}^c(1) - \Phi_j^c(-1)\Phi_{j'}^c(-1) = \lambda_j \lambda_{j'} \psi_j^c(0) \psi_{j'}^c(0)$$

and, since $\psi_j(0) = 0$ if j is odd (due to parity of PSWFs), we arrive at (6.5) and (6.6).

Using (6.2) and (6.1), we have

$$\Phi_j^c(x) = \frac{1}{\lambda_j} \int_{-1}^{1} \left(\int_{-1}^{x} e^{icys} ds \right) \psi_j^c(y) dy = \frac{1}{\lambda_j} \int_{-1}^{1} \frac{e^{icyx} - e^{-icy}}{icy} \psi_j^c(y) dy,$$

and, thus,

$$
\begin{aligned}
I_{jj'} &= \frac{1}{\lambda_j} \int_{-1}^{1} \left[\int_{-1}^{1} \frac{e^{icyx} - e^{-icy}}{icy} \psi_j^c(y) dy \right] \psi_{j'}^c(x) dx \\
(6.10) \quad &= \frac{\lambda_{j'}}{ic\lambda_j} \left(\int_{-1}^{1} \psi_j^c(y) \frac{\psi_{j'}^c(y)}{y} dy - \psi_{j'}^c(0) \int_{-1}^{1} \frac{\psi_j^c(y)}{y} e^{-icy} dy \right).
\end{aligned}
$$

It follows from (6.10) that if j is even and j' is odd (so that $\psi_{j'}^c(0) = 0$), we obtain (6.7) and

$$I_{j'j} = \frac{\lambda_j}{ic\lambda_{j'}} \left(\int_{-1}^{1} \psi_j^c(y) \frac{\psi_{j'}^c(y)}{y} dy - \psi_j^c(0) \int_{-1}^{1} \frac{\psi_{j'}^c(y)}{y} e^{-icy} dy \right).$$

Introducing

$$u(x) = \int_{-1}^{1} \frac{\psi_{j'}^c(y)}{y} e^{-icyx} dy,$$

we have

$$u'(x) = -ic \int_{-1}^{1} \psi_{j'}^c(y) e^{-icyx} dy = -ic\overline{\lambda}_{j'} \psi_{j'}^c(x)$$

so that

$$u(x) = u(a) - ic\overline{\lambda}_{j'} \int_{a}^{x} \psi_{j'}^c(s) ds.$$

Setting $x = 1$ and $a = 0$, we obtain

$$
\begin{aligned}
\int_{-1}^{1} \frac{\psi_{j'}^c(y)}{y} e^{-icy} dy &= \int_{-1}^{1} \frac{\psi_{j'}^c(y)}{y} dy - ic\overline{\lambda}_{j'} \int_{0}^{1} \psi_{j'}^c(y) dy \\
&= 2 \int_{0}^{1} \frac{\psi_{j'}^c(y)}{y} dy - ic\overline{\lambda}_{j'} \int_{0}^{1} \psi_{j'}^c(y) dy
\end{aligned}
$$

and arrive at (6.8). $\qquad \square$

22

6.2. Computing integration matrix using approximate PSWFs.
If the interpolating basis for band-limited functions is defined via (2.10), then the coefficients r_{kl} are obtained using

$$(6.11) \qquad \delta_{km} = R_k(\tau_m) = \sum_{l=1}^{M} r_{kl} e^{ic\tau_l \tau_m}$$

by inverting the matrix $E = \{e^{ic\tau_l \tau_m}\}_{l,m=1,\dots M}$. We have

$$K_k(x) = \int_{-1}^{x} R_k(s)ds = \sum_{l=1}^{M} r_{kl} \frac{e^{ic\tau_l x} - e^{-ic\tau_l}}{ic\tau_l}$$

and compute

$$
\begin{aligned}
w_k S_{kl} &= \int_{-1}^{1} K_l(x) R_k(x) dx \\
&= \sum_{j,j'=1,\dots M} r_{kj} r_{lj'} \int_{-1}^{1} e^{ic\tau_j x} \frac{e^{ic\tau_{j'} x} - e^{-ic\tau_{j'}}}{ic\tau_{j'}} dx \\
&= \sum_{j,j'=1,\dots M} r_{kj} r_{lj'} G_{jj'},
\end{aligned}
$$

where

$$G_{jj'} = 2\frac{\operatorname{sinc}\left(c\left(\tau_j + \tau_{j'}\right)\right) - e^{-ic\tau_{j'}} \operatorname{sinc}\left(c\tau_j\right)}{ic\tau_{j'}}.$$

Thus, we have

$$w_k S_{kl} = \left(E^{-1} G E^{-1}\right)_{kl}.$$

References

[1] G.V. Berghe, H. D. Meyer, M. V. Daele, and T. V. Hecke. Exponentially fitted Runge–Kutta methods. *Journal of Computational and Applied Mathematics*, 125(1):107–115, 2000.
[2] G. Beylkin and L. Monzón. On generalized Gaussian quadratures for exponentials and their applications. *Appl. Comput. Harmon. Anal.*, 12(3):332–373, 2002.
[3] G. Beylkin and K. Sandberg. Wave propagation using bases for bandlimited functions. *Wave Motion*, 41(3):263–291, 2005.
[4] J. P Boyd, G. Gassner, and B. A Sadiq. The nonconvergence of h-refinement in prolate elements. *Journal of Scientific Computing*, 57:1–18, 2013.
[5] B.K. Bradley, B.A. Jones, G. Beylkin, and P. Axelrad. A new numerical integration technique in astrodynamics. In *22nd Annual AAS/AIAA Space Flight Mechanics Meeting, Charleston, SC, Jan. 29 - Feb. 2*, 2012.
[6] B.K. Bradley, B.A. Jones, G. Beylkin, K. Sandberg, and P. Axelrad. Bandlimited Implicit Runge-Kutta Integration for Astrodynamics. *Celestial Mechanics and Dynamical Astronomy*, 2013. submitted.

[7] P. Brock and F. J. Murray. The use of exponential sums in step by step integration. *Mathematical Tables and Other Aids to Computation*, 6(38):63–78, 1952.

[8] H. V. de Vyver. Frequency evaluation for exponentially fitted Runge–Kutta methods. *Journal of computational and applied mathematics*, 184(2):442–463, 2005.

[9] K. Dekker and J.G. Verwer. *Stability of the Runge-Kutta methods for stiff nonlinear differential equations*. North-Holland, Amsterdam, 1984.

[10] Dept. of Defense World Geodetic System. Defense Mapping Agency Technical Report. Technical report, 1987. DMA TR 8350.2.

[11] A. Dutt, L. Greengard, and V. Rokhlin. Spectral deferred correction methods for ordinary differential equations. *BIT*, 40(2):241–266, 2000.

[12] W. Gautschi. Numerical integration of ordinary differential equations based on trigonometric polynomials. *Numerische Mathematik*, 3(1):381–397, 1961.

[13] A. Glaser and V. Rokhlin. A new class of highly accurate solvers for ordinary differential equations. *Journal of Scientific Computing*, 38(3):368–399, 2009.

[14] D. Gottlieb and S. A. Orszag. *Numerical analysis of spectral methods: theory and applications*. Society for Industrial and Applied Mathematics, Philadelphia, Pa., 1977. CBMS-NSF Regional Conference Series in Applied Mathematics, No. 26.

[15] J. Huang, J. Jia, and M. Minion. Accelerating the convergence of spectral deferred correction methods. *J. Comput. Phys.*, 214(2):633–656, 2006.

[16] A. Iserles. *A first course in the numerical analysis of differential equations*. Cambridge University Press, 1996.

[17] L. G. Ixaru, M. Rizea, G. V. Berghe, and H. D. Meyer. Weights of the exponential fitting multistep algorithms for first-order ODEs. *Journal of computational and applied mathematics*, 132(1):83–93, 2001.

[18] L.G. Ixaru, Vanden B. G., and H. De Meyer. Frequency evaluation in exponential fitting multistep algorithms for ODEs. *Journal of Computational and Applied Mathematics*, 140(1):423–434, 2002.

[19] J. Jia and J. Huang. Krylov deferred correction accelerated method of lines transpose. *Journal of Computational Physics*, 227(3):1739–1753, 2008.

[20] W. Y. Kong and V. Rokhlin. A new class of highly accurate differentiation schemes based on the prolate spheroidal wave functions. *Appl. Comput. Harmon. Anal.*, 33(2):226–260, 2012. doi:10.1016/j.acha.2011.11.005.

[21] D. Kushnir and V. Rokhlin. A Highly Accurate Solver for Stiff Ordinary Differential Equations. *SIAM Journal on Scientific Computing*, 34(3):A1296–A1315, 2012.

[22] H. J. Landau and H. O. Pollak. Prolate spheroidal wave functions, Fourier analysis and uncertainty II. *Bell System Tech. J.*, 40:65–84, 1961.

[23] H. J. Landau and H. O. Pollak. Prolate spheroidal wave functions, Fourier analysis and uncertainty III. *Bell System Tech. J.*, 41:1295–1336, 1962.

[24] A. T. Layton and M. L. Minion. Implications of the choice of quadrature nodes for Picard integral deferred corrections methods for ordinary differential equations. *BIT*, 45(2):341–373, 2005.

[25] F.G. Lemoine, S.C. Kenyon, J.K. Factor, R.G. Trimmer, N.K. Pavlis, D.S. Chinn, C.M. Cox, S.M. Klosko, S.B. Luthcke, M.H. Torrence, et al. The development of the joint NASA GSFC and the National Imagery and Mapping Agency (NIMA) geopotential model EGM96. *NASA*, (19980218814), 1998.

[26] M. Mäkelä, O. Nevanlinna, and A. H Sipilä. Exponentially fitted multistep methods by generalized Hermite-Birkhoff interpolation. *BIT Numerical Mathematics*, 14(4):437–451, 1974.

[27] A. Osipov, V. Rokhlin, and H. Xiao. *Prolate Spheroidal Wave Functions of Order Zero*. Springer, 2013.

[28] M. Reynolds, G. Beylkin, and L. Monzón. On generalized Gaussian quadratures for bandlimited exponentials. *Appl. Comput. Harmon. Anal.*, 34:352–365, 2013.

[29] K. Sandberg and K.J. Wojciechowski. The EPS method: A new method for constructing pseudospectral derivative operators. *J. Comp. Phys.*, 230(15):5836–5863, 2011.

[30] J.M. Sanz-Serna. Runge-Kutta schemes for Hamiltonian systems. *BIT*, v. 28:877–883, 1988.

[31] D. Slepian. Prolate spheroidal wave functions, Fourier analysis and uncertainty IV. Extensions to many dimensions; generalized prolate spheroidal functions. *Bell System Tech. J.*, 43:3009–3057, 1964.

[32] D. Slepian. Some asymptotic expansions for prolate spheroidal wave functions. *J. Math. and Phys.*, 44:99–140, 1965.

[33] D. Slepian. Prolate spheroidal wave functions, Fourier analysis and uncertainty V. The discrete case. *Bell System Tech. J.*, 57:1371–1430, 1978.

[34] D. Slepian. Some comments on Fourier analysis, uncertainty and modeling. *SIAM Review*, 25(3):379–393, 1983.

[35] D. Slepian and H. O. Pollak. Prolate spheroidal wave functions, Fourier analysis and uncertainty I. *Bell System Tech. J.*, 40:43–63, 1961.

[36] H. Xiao, V. Rokhlin, and N. Yarvin. Prolate spheroidal wavefunctions, quadrature and interpolation. *Inverse Problems*, 17(4):805–838, 2001.

*Department of Applied Mathematics, University of Colorado at Boulder, 526 UCB, Boulder, CO 80309-0526, **Computational Solutions, Inc., 1800 30th Street, Suite 210B, Boulder, Colorado 80301