Report no. 09/02

Fast, numerically stable computation of oscillatory integrals with stationary points

Sheehan Olver

Oxford University Computing Laboratory, Wolfson Building, Parks Road, Oxford, UK

Sheehan.Olver@sjc.ox.ac.uk

Abstract We present a numerically stable way to compute oscillatory integrals of the form $\int_{-1}^{1} f(x)\mathrm{e}^{\mathrm{i}\omega g(x)}\,\mathrm{d}x$. For each additional frequency, only a small, well-conditioned linear system with a Hessenberg matrix must be solved, and the amount of work needed decreases as the frequency increases. Moreover, we can modify the method for computing oscillatory integrals with stationary points. This is the first stable algorithm for oscillatory integrals with stationary points which does not lose accuracy as the frequency increases and does not require deformation into the complex plane.

Keywords GMRES, highly oscillatory integrals, quadrature.

Oxford University Computing Laboratory
Numerical Analysis Group
Wolfson Building
Parks Road
Oxford, England OX1 3QD

May, 2009

1. Introduction

We are interested in solving the oscillatory integral

$$\mathcal{I}_\omega[f] = \int_{-1}^{1} f(x) \mathrm{e}^{\mathrm{i}\omega g(x)} \, \mathrm{d}x,$$

where $f \in \mathcal{C}^1[-1,1]$ and $g \in \mathcal{C}^r[-1,1]$ has a single stationary point of order $r-1$ at zero: $0 = g'(0) = \cdots = g^{(r-1)}(0)$, $g^{(r)}(0) \neq 0$ and $g'(x) \neq 0$ for $x \neq 0$. If the integral is over another domain $[a,b]$ with finitely many stationary points, then it can be written in terms of integrals of the form $\mathcal{I}_\omega[f]$.

Many methods have been developed recently for computing $\mathcal{I}_\omega[f]$, with an emphasis on the asymptotic decay as $\omega \to \infty$. High asymptotic orders is achieved in Filon-type methods [**11**] and moment-free Filon-type methods [**29**, **21**] by interpolating the derivatives of f at the stationary point and endpoints of the interval (i.e., zero and ± 1). Alternatively, derivatives can be avoided by allowing the interpolation points to approach the the critical points as ω increases [**12**]. Interpolating at such points along with Chebyshev points ensures convergence [**17**]. Unfortunately, it is unknown how to construct these methods in a numerically stable manner with such a choice of points, or how to choose the interpolation points to optimize the order of convergence.

Numerical steepest descent [**10**, **7**] simultaneously achieves high asymptotic order and numerical stability. We can remove the oscillations from the integrand by deforming the contour of integration along the steepest descent path. A standard quadrature method can then be used; in particular, generalized Laguerre quadrature. This method achieves roughly twice the asymptotic order as the other high asymptotic order methods for the same number of function evaluations. Unfortunately, it requires that the integral be analytic and have at most exponential growth in the complex plane, and complicated singularities in the complex plane can prevent convergence. In addition, the method we develop has the interesting property that it converges even when ω is complex or small in magnitude — it becomes equivalent to Clenshaw–Curtis quadrature [**4**, **27**] at $\omega = 0$ — whereas numerical steepest descent can fall pray to Stokes' phenomena [**18**] and the error blows up as $\omega \to 0$ for a fixed number of sample points.

When g is free of stationary points (i.e., $r = 1$), The Levin collocation method [**14**] can be computed in a stable fashion using a TSVD decomposition [**15**], however, it does not achieve high asymptotic orders and the amount of computation does not decrease significantly as ω increases. Levin-type methods [**24**] achieve high asymptotic order but are not numerically stable. Differential GMRES [**20**, **19**] is stable and achieves high asymptotic order, but requires derivatives. We could avoid derivatives by implementing differential GMRES in the `chebfun` system [**2**], which is very similar to the method we construct for the case without stationary points. Unfortunately, its adaptivity is a significant hindrance to performance, and errors

due to removable singularities prevent the use of such a method for integrals with stationary points.

We will construct a method $\mathcal{Q}^{\mathrm{M}}_{\omega,n,m}[f]$ to approximate $\mathcal{I}_\omega[f]$ which trades high asymptotic order in exchange for numerical stability and fast computation. Moreover, it preserves the most important property (from a computational perspective) of high asymptotic order: the computational cost needed to achieve a certain accuracy decreases as ω increases. Indeed, if n is sufficiently large, we obtain an effective asymptotic order of $\mathcal{O}\left(\omega^{-m-1}\right)$ — the method decays at this rate to until an error on the order of machine epsilon is obtained — using only $\mathcal{O}(mn \log n)$ operations. In addition, we reuse the exact same values of f for every choice of ω; thus, less computation is required if the same integral is evaluated with multiple values of ω.

For the case without stationary points, $\mathcal{Q}^{\mathrm{M}}_{\omega,n,m}[f]$ can be viewed as a way of computing the Levin collocation method. Suppose $u \in \mathcal{C}^1[-1,1]$ is a particular solution to the differential equation

$$\mathcal{L}u = f \qquad \text{for} \qquad \mathcal{L} = \mathcal{D} + \mathrm{i}\omega g', \tag{1.1}$$

where $\mathcal{D} : \mathcal{C}^1[-1,1] \to \mathcal{C}^0[-1,1]$ is the differentiation operator, so that $\mathcal{L}u = u' + \mathrm{i}\omega g' u$. It follows that

$$\mathcal{I}_\omega[f] = \mathcal{I}_\omega[\mathcal{L}u] = \int_{-1}^{1} (u\mathrm{e}^{\mathrm{i}\omega g})' \,\mathrm{d}x = u(1)\mathrm{e}^{\mathrm{i}\omega g(1)} - u(-1)\mathrm{e}^{\mathrm{i}\omega g(-1)}. \tag{1.2}$$

Thus finding a particular solution to (1.1) allows us to calculate $\mathcal{I}_\omega[f]$. The idea behind the Levin collocation method is to use collocation to approximate a particular solution to (1.1). In other words, for a basis $\psi_1, \ldots, \psi_n$ and sequence of collocation points $x_1, \ldots, x_n$ we find a function $u = \sum_{k=1}^{n} c_k \psi_k$ which satisfies

$$\mathcal{L}u(x_1) = f(x_1), \ldots, \mathcal{L}u(x_1) = f(x_n).$$

The method we construct uses GMRES [**26**, **25**] on the resulting linear system to compute the coefficients $\boldsymbol{c} = (c_1, \ldots, c_n)^\top$. By multiplying the linear system on the right appropriately, we obtain a shifted linear system. This allows us to reuse the same Arnoldi factorization for every value of ω [**6**]. Moreover, the rate of convergence of the GMRES algorithm increases as ω increases; hence, we need fewer iterations of GMRES when ω is large to achieve a similar accuracy.

In Section 2 we set up the Levin collocation method in more detail and demonstrate its numerical problems. In Section 3 we describe the GMRES algorithm and use it to construct $\mathcal{Q}^{\mathrm{M}}_{\omega,n,m}[f]$. We also detail the properties of shifted linear systems which our method exploits. In Section 4 we prove convergence, using a connection between $\mathcal{Q}^{\mathrm{M}}_{\omega,n,m}[f]$ and differential GMRES. In Section 5, we construct a method $\mathcal{Q}^{\mathrm{M}}_{\omega,n,m}[f]$ for the case where g has stationary points and demonstrate the effectiveness of the method by computing the Bessel function J_ν

for integer ν throughout the complex plane. Finally, in Section 6 we demonstrate numerically the loss of convergence as $n \to \infty$. Despite these problems, the method still achieves higher accuracy and takes less work than other methods for computing the Levin collocation method.

2. Levin collocation method

For a sequence of n points $\boldsymbol{x} = (x_1, \ldots, x_n)^\top$ and $f \in \mathcal{C}^0[-1,1]$, we define $f(\boldsymbol{x})$ as the vector of f applied to each component of $\boldsymbol{x}$: $f(\boldsymbol{x}) = (f(x_1), \ldots, f(x_n))^\top$. Then, for a quasimatrix (a row vector whose columns are functions who belong to $\mathcal{C}^0[-1,1]$) $\boldsymbol{\psi} = (\psi_1, \ldots, \psi_n)$, we define

$$\boldsymbol{\psi}(\boldsymbol{x}) = (\psi_1(\boldsymbol{x}), \ldots, \psi_n(\boldsymbol{x})) = \begin{pmatrix} \psi_1(x_1) & \cdots & \psi_n(x_1) \\ \vdots & \ddots & \vdots \\ \psi_1(x_n) & \cdots & \psi_n(x_n) \end{pmatrix}.$$

Thus, given a differentiable quasimatrix $\boldsymbol{\psi}$ and collocation points $\boldsymbol{x}$, we can rephrase the Levin collocation method as finding a function $u_n = \boldsymbol{\psi}\boldsymbol{c}$ such that $\mathcal{L}u_n(\boldsymbol{x}) = f(\boldsymbol{x})$. The coefficients $\boldsymbol{c}$ are thus found by solving the linear system

$$\mathcal{L}\boldsymbol{\psi}(\boldsymbol{x})\boldsymbol{c} = f(\boldsymbol{x}), \tag{2.1}$$

where applying a differential operator to a quasimatrix is equivalent to applying the operator to each column. We then obtain a Levin collocation method:

$$\mathcal{Q}_\omega^{\mathrm{L}}[f] = u_n(1)\mathrm{e}^{\mathrm{i}\omega g(1)} - u_n(-1)\mathrm{e}^{\mathrm{i}\omega g(-1)} = \left[\boldsymbol{\psi}(1)\mathrm{e}^{\mathrm{i}\omega g(1)} - \boldsymbol{\psi}(-1)\mathrm{e}^{\mathrm{i}\omega g(-1)}\right]\boldsymbol{c}$$

for

$$\boldsymbol{c} = (\mathcal{L}\boldsymbol{\psi}(\boldsymbol{x}))^{-1} f(\boldsymbol{x}).$$

The asymptotic order of Levin collocation methods is known, and a higher asymptotic order is achieved when we collocate at the endpoints.

Theorem 2.1 [**14**, **24**] *Suppose that the columns of $\boldsymbol{\psi}$ are independent of ω and form a Chebyshev set and g' does not vanish. If $g \in \mathcal{C}^2[-1,1]$ and $f \in \mathcal{C}^1[-1,1]$, then*

$$\mathcal{Q}_\omega^{\mathrm{L}}[f] = \mathcal{O}\left(\omega^{-1}\right), \qquad \omega \to \infty.$$

If $g \in \mathcal{C}^3[-1,1]$, $f \in \mathcal{C}^2[-1,1]$, $x_1 = -1$ and $x_n = 1$, then

$$\mathcal{Q}_\omega^{\mathrm{L}}[f] = \mathcal{O}\left(\omega^{-2}\right), \qquad \omega \to \infty.$$

The obvious choice for the columns of $\boldsymbol{\psi}$ are the Chebyshev polynomials of the first kind [1] — i.e., $\boldsymbol{\psi}^{\mathrm{T}} = (T_0, \ldots, T_{n-1})$ — with $\boldsymbol{x}$ equal to the Chebyshev–Lobatto points

$$\boldsymbol{x}^{\mathrm{T}} = \begin{pmatrix} -1 \\ \cos \pi \frac{n-2}{n-1} \\ \cos \pi \frac{n-3}{n-1} \\ \vdots \\ \cos \pi \frac{2}{n-1} \\ \cos \pi \frac{1}{n-1} \\ 1 \end{pmatrix}.$$

This choice of nodes includes the endpoints, which ensure that a higher asymptotic order is achieved. We define the discrete inverse Chebyshev transform as $\mathcal{T}^{-1} = \boldsymbol{\psi}^{\mathrm{T}}(\boldsymbol{x}^{\mathrm{T}})$, which is equivalent to the Chebyshev–Vandermonde matrix. $\mathcal{T}^{-1}\boldsymbol{c}$ maps the coefficients $\boldsymbol{c}$ of a Chebyshev series to the values the series takes at $\boldsymbol{x}^{\mathrm{T}}$. The Chebyshev transform is then $\mathcal{T}$, which maps $\boldsymbol{f} = f(\boldsymbol{x}^{\mathrm{T}})$ to the coefficients of the nth degree Chebyshev series which interpolates f at $\boldsymbol{x}^{\mathrm{T}}$. Both $\mathcal{T}$ and $\mathcal{T}^{-1}$ applied to a vector can be evaluated in $\mathcal{O}(n \log n)$ operations using the fast cosine transform.

Let $\mathbf{D}$ map the coefficients in a Chebyshev series to those of its derivatives, i.e., $\boldsymbol{\psi}^{\mathrm{T}\prime} = \boldsymbol{\psi}^{\mathrm{T}}\mathbf{D}$. We can define $\mathbf{D}$ by $\mathbf{D}\boldsymbol{c} = (d_1, \ldots, d_n)^{\top}$, where (cf. [9])

$$\begin{aligned} d_n &= 0, \\ d_{n-1} &= 2nc_n, \\ d_k &= d_{k+2} + 2(k+1)c_{k+1}, \qquad k = n-2, n-3, \ldots, 2, \\ d_1 &= \frac{d_3}{2} + c_2. \end{aligned}$$

It follows that

$$u_n'(\boldsymbol{x}^{\mathrm{T}}) = \boldsymbol{\psi}^{\mathrm{T}\prime}(\boldsymbol{x}^{\mathrm{T}})\boldsymbol{c} = \boldsymbol{\psi}^{\mathrm{T}}(\boldsymbol{x}^{\mathrm{T}})\mathbf{D}\boldsymbol{c} = \mathcal{T}^{-1}\mathbf{D}\boldsymbol{c},$$

and the linear system (2.1) is now

$$\mathbf{L}\boldsymbol{c} = \boldsymbol{f} \quad \text{for} \quad \mathbf{L} = \mathcal{L}\boldsymbol{\psi}^{\mathrm{T}}(\boldsymbol{x}^{\mathrm{T}}) = \mathcal{T}^{-1}\mathbf{D} + \mathrm{i}\omega G\mathcal{T}^{-1} \quad \text{and} \quad G = \operatorname{diag}(g'(\boldsymbol{x}^{\mathrm{T}})). \tag{2.2}$$

Applying $\mathbf{D}$ to a vector clearly takes $\mathcal{O}(n)$ operations; thence, applying $\mathbf{L}$ to a vector takes $\mathcal{O}(n \log n)$ operations.

Definition 2.2 We use curly capitals for differential linear operators and the Chebyshev transform, capitals for matrices operating on function values and bold capitals for matrices operating on Chebyshev series' coefficients. I denotes the identity operator in all spaces. Finally, in the definition of operators, any constant ω denotes ωI.

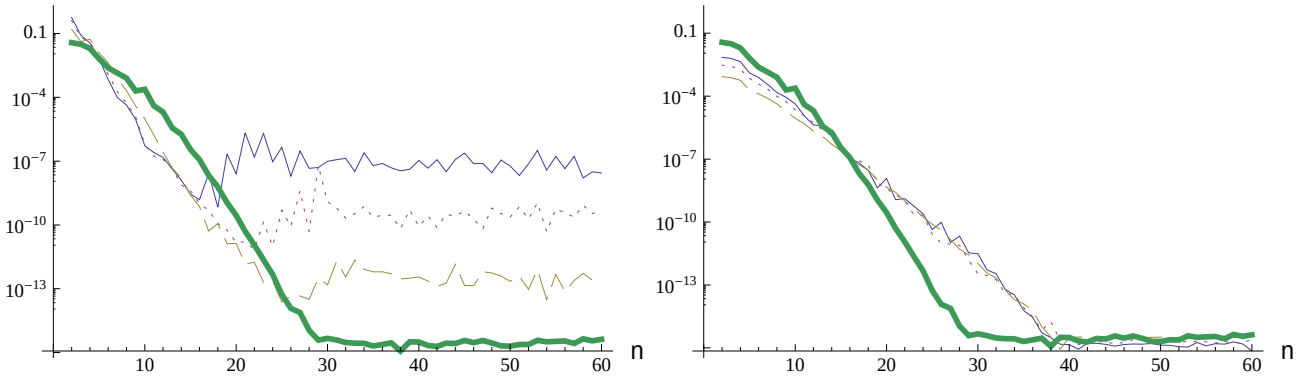

Figure 1: The error in approximating $\int_{-1}^{1} \frac{1}{x^2+1} \mathrm{e}^{\mathrm{i}\omega \sin\left(x+\frac{1}{4}\right)} \,\mathrm{d}x$ by $\mathcal{Q}_{\omega,n}^{\mathrm{T}}[f]$ for $\omega = 0.1$ (left, plain), 1 (left, dotted), 3 (left, dashed), 10 (both, thick), 30 (right, plain), 50 (right, dotted) and 100 (right, dashed).

Definition 2.3 The interpolation operator $\boldsymbol{e}(x)$ is defined by $\boldsymbol{e}(x)^\top \boldsymbol{f} = \boldsymbol{\psi}^{\mathrm{T}}(x)\mathcal{T}\boldsymbol{f}$, i.e., $\boldsymbol{e}(x)^\top \boldsymbol{f}$ is the polynomial which interpolates the data $\boldsymbol{f}$ at the points $\boldsymbol{x}^{\mathrm{T}}$. Thus $\boldsymbol{e}(\boldsymbol{x}^{\mathrm{T}}) = I$, or in particular, $\boldsymbol{e}(-1) = \boldsymbol{e}_1$ and $\boldsymbol{e}(1) = \boldsymbol{e}_n$ (which motivates the choice of notation).We use the barycentric formula [3] for intermediate values of x.

Note that $u_n(x) = \boldsymbol{\psi}^{\mathrm{T}}(x)\boldsymbol{c} = \boldsymbol{e}(x)^\top \mathcal{T}^{-1}\boldsymbol{c} = \boldsymbol{e}(x)^\top \mathcal{T}^{-1}\mathbf{L}^{-1}\boldsymbol{f}$. We thus obtain the Levin–Chebyshev collocation method:

Algorithm 2.4 Levin–Chebyshev collocation method

Given a function $f \in \mathcal{C}^0[-1,1]$, $g \in \mathcal{C}^1[-1,1]$ and integer n, compute $\mathcal{Q}_{\omega,n}^{\mathrm{T}}[f]$ as follows:

1: Let $\boldsymbol{f} = f(\boldsymbol{x}^{\mathrm{T}})$;

2: Construct $\mathbf{L} = \mathcal{T}^{-1}\mathbf{D} + \mathrm{i}\omega G\mathcal{T}^{-1}$ in matrix form;

3: Define

$$\mathcal{Q}_{\omega,n}^{\mathrm{T}}[f] = u_n(1)\mathrm{e}^{\mathrm{i}\omega g(1)} - u_n(-1)\mathrm{e}^{\mathrm{i}\omega g(-1)} = \left(\mathrm{e}^{\mathrm{i}\omega g(1)}\boldsymbol{e}(1)^\top - \mathrm{e}^{\mathrm{i}\omega g(-1)}\boldsymbol{e}(-1)^\top\right)\mathcal{T}^{-1}\mathbf{L}^{-1}\boldsymbol{f}.$$

In this algorithm, we needed to compute $\mathbf{L}^{-1}\boldsymbol{f}$. For fixed n, $\mathbf{L}$ is well-conditioned for sufficiently large ω, however, the condition number does increase as $n \to \infty$. We demonstrate this issue in Figure 1, where the integral

$$\mathcal{I}_\omega[f] = \int_{-1}^{1} \frac{1}{x^2+1} \mathrm{e}^{\mathrm{i}\omega \sin\left(x+\frac{1}{4}\right)} \,\mathrm{d}x,$$

is approximated by $\mathcal{Q}_{\omega,n}^{\mathrm{T}}[f]$ and $\mathbf{L}^{-1}\boldsymbol{f}$ is computed using Gaussian elimination. The numerical instability at low frequencies of the algorithm is evident.

3. GMRES

In [15], $\mathbf{L}^{-1}\boldsymbol{f}$ was computed using TSVD, resulting in the delaminating quadrature method. Indeed, numerical experiments suggest that the error caused by numerical instability is reduced if an algorithm based on LAPACK is used. As an alternative, we will use GMRES, which will not suffer from the same problems with stability as Gaussian elimination. Moreover, this approach has the remarkable property that the method converges faster when ω is large. In other words, for a fixed value n, less work is required for large ω to compute $\mathbf{L}^{-1}\boldsymbol{f}$.

We now describe the GMRES algorithm. The Krylov subspace is defined as follows:

Definition 3.1 For $\boldsymbol{f} \in \mathbb{C}^n$ and $M \in \mathbb{C}^{n\times n}$, the Krylov subspace is

$$\mathcal{K}_m[M, \boldsymbol{f}] = \operatorname{span}\left\{\boldsymbol{f}, M\boldsymbol{f}, \ldots, M^{m-1}\boldsymbol{f}\right\}.$$

Assume that $\boldsymbol{f}$ is not identically zero and that $\mathcal{K}_m[M, \boldsymbol{f}]$ has dimension m. Then Arnoldi iteration generates $n \times m$ matrices $Q_m = (\boldsymbol{q}_1, \ldots, \boldsymbol{q}_m)$ and $(m+1) \times m$ Hessenberg matrices

$$H_m = \begin{pmatrix} h_{1,1} & h_{1,2} & \cdots & h_{1,m} \\ h_{2,1} & h_{2,2} & \ddots & \vdots \\ & \ddots & \ddots & h_{m-1,m} \\ & & h_{m,m-1} & h_{m,m} \\ & & & h_{m+1,m} \end{pmatrix}$$

with the following properties:

- $(\boldsymbol{q}_1, \ldots, \boldsymbol{q}_m)$ form an orthonormal basis of $\mathcal{K}_m[M, \boldsymbol{f}]$;
- $\boldsymbol{q}_1 = \boldsymbol{f}/\left\|\boldsymbol{f}\right\|$ and $MQ_m = Q_{m+1}H_m$.

The algorithm typically used is essentially the Gram–Schmidt process — see [28] for details of the algorithm — applied to the matrix $(\boldsymbol{q}_1, \ldots, \boldsymbol{q}_m, M\boldsymbol{q}_m)$. Thus Arnoldi iteration does not require M in matrix form; rather, it only requires the ability to apply M to a fixed vector. The benefit in our case is that the operator M can be applied to a vector in $\mathcal{O}(n \log n)$, where applying the operator in matrix form would take $\mathcal{O}\left(n^2\right)$ operations. An alternative to this algorithm is Arnoldi–Householder iteration, cf. [25]. This version ensures that the columns of Q_m remain orthonormal for large n, as opposed to the Gram–Schmidt process which eventually loses orthogonality. In this case,

$$\boldsymbol{q}_1 = -\mathrm{e}^{-\mathrm{i}\arg e_1^\top \boldsymbol{f}} \frac{\boldsymbol{f}}{\left\|\boldsymbol{f}\right\|}.$$

In this paper, we use the Gram–Schmidt version for simplicity.

GMRES is an algorithm which finds an element $\boldsymbol{v}_m \in \mathcal{K}_m[M, \boldsymbol{f}]$ such that $\|M\boldsymbol{v}_m - f\|$ is minimal. Since the columns of Q_m span $\mathcal{K}_m[M, \boldsymbol{f}]$, we can write $\boldsymbol{v}_m = Q_m\boldsymbol{d}_m$, hence

$$\|M\boldsymbol{v}_m - \boldsymbol{f}\| = \|MQ_m\boldsymbol{d}_m - \|\boldsymbol{f}\|\, q_1\| = \|Q_{m+1}(H_m\boldsymbol{d}_m - \|\boldsymbol{f}\|\, \boldsymbol{e}_1)\| = \|H_m\boldsymbol{d}_m - \|\boldsymbol{f}\|\, \boldsymbol{e}_1\|\,.$$

Thus we need only find the least squares solution which minimizes $\|H_m\boldsymbol{d}_m - \|\boldsymbol{f}\|\, \boldsymbol{e}_1\|$. Thus the standard GMRES algorithm is as follows:

Algorithm 3.2 [26] GMRES

Given a vector $\boldsymbol{f} \in \mathbb{C}^n$ and operator $M \in \mathbb{C}^{n\times n}$, compute $\boldsymbol{v}_m$ as follows:

1: Compute Q_m and H_m using Arnoldi iteration on M and $\boldsymbol{f}$;

2: Use least squares to find $\boldsymbol{d}_m \in \mathbb{C}^m$ which minimizes the norm

$$\|H_m\boldsymbol{d}_m - \|f\|\, \boldsymbol{e}_1\|\,;$$

3: Define

$$\boldsymbol{v}_m = Q_m\boldsymbol{d}_m.$$

In [8], it was noted that GMRES performs especially well for shifted linear operators of the form

$$M = M_0 + \mathrm{i}\omega,$$

where $M_0 \in \mathbb{C}^{n\times n}$ is independent of ω. This is explained via the following theorems, which imply that we we can reuse the Arnoldi process applied to the case $\omega = 0$ for all other values of ω, the resulting least squares system is well-conditioned and GMRES obtains a high asymptotic order for $\omega \to \infty$.

Theorem 3.3 [6] *Suppose that Arnoldi iteration with $M_0 \in \mathbb{C}^{n\times n}$ and $\boldsymbol{f} \in \mathbb{C}^{n\times n}$ returns $\tilde{Q}_m$ and $\tilde{H}_m$. Then Arnoldi iteration applied to the operator $M = M_0 + \mathrm{i}\omega$ returns $Q_m = \tilde{Q}_m$ and the Hessenberg matrix*

$$H_{m,\omega} = \tilde{H}_m + \mathrm{i}\omega I_{m+1,m} \qquad \text{for the } (m+1)\times m \text{ matrix} \qquad I_{m+1,m} = \begin{pmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{pmatrix}.$$

Corollary 3.4 [19] *Any condition number $\kappa(H_{m,\omega}) \to 1$ as $\omega \to \infty$.*

Corollary 3.5 [19]

$$\|M\boldsymbol{v}_m - \boldsymbol{f}\| = \mathcal{O}\big(\omega^{-m}\big) \qquad \text{as} \qquad \omega \to \infty.$$

To take advantage of these results, we must premultiply $\mathbf{L}$ so that it takes the form $M_0 + \mathrm{i}\omega$. Since we still are assuming that g' does not vanish, this is trivial: we use the operator

$$M = \mathbf{L}\mathcal{T}G^{-1} = DG^{-1} + \mathrm{i}\omega, \tag{3.1}$$

where D is the Chebyshev differentiation matrix

$$D = \mathcal{T}^{-1}\mathbf{D}\mathcal{T}.$$

We then want to solve

$$M\boldsymbol{v} = \boldsymbol{f}. \tag{3.2}$$

The coefficients $\boldsymbol{c}$ are equal to $\mathcal{T}G^{-1}\boldsymbol{v}$, hence now

$$u_n(x) = \boldsymbol{e}(x)^\top \boldsymbol{c} = \boldsymbol{e}(x)^\top \mathcal{T}^{-1}\mathcal{T}G^{-1}\boldsymbol{v} = \boldsymbol{e}(x)^\top G^{-1}\boldsymbol{v}.$$

Applying GMRES to solve (3.2) results in the following algorithm:

Algorithm 3.6 GMRES–Levin collocation method

Given a function $f \in \mathcal{C}^0[-1,1]$ and $g \in \mathcal{C}^1[-1,1]$ such that g' does not vanish, compute $\mathcal{Q}^{\mathrm{M}}_{\omega,n,m}[f]$ as follows:

1: Precompute Q_m and $\tilde{H}_m$ using Arnoldi iteration with DG^{-1} and $\boldsymbol{f} = f\left(\boldsymbol{x}^{\mathrm{T}}\right)$;

2: Construct $H_{m,\omega} = \tilde{H}_m + \mathrm{i}\omega I_{m+1,m}$;

3: Use least squares to find $\boldsymbol{d}_m \in \mathbb{C}^m$ which minimizes the norm

$$\left\| H_{m,\omega}\boldsymbol{d}_m - \|\boldsymbol{f}\|\,\boldsymbol{e}_1 \right\|;$$

4: For $\boldsymbol{v}_m = Q_m\boldsymbol{d}_m$, define

$$\begin{aligned}\mathcal{Q}^{\mathrm{M}}_{\omega,n,m}[f] &= \left(\mathrm{e}^{\mathrm{i}\omega g(1)}\boldsymbol{e}(1)^\top - \mathrm{e}^{\mathrm{i}\omega g(-1)}\boldsymbol{e}(-1)^\top\right) G^{-1}\boldsymbol{v}_m \\ &= \frac{\mathrm{e}^{\mathrm{i}\omega g(1)}}{g'(1)}\boldsymbol{e}_n^\top\boldsymbol{v}_m - \frac{\mathrm{e}^{\mathrm{i}\omega g(-1)}}{g'(-1)}\boldsymbol{e}_1^\top\boldsymbol{v}_m.\end{aligned}$$

Q.E.D.

We now demonstrate $\mathcal{Q}^{\mathrm{M}}_{\omega,n,m}[f]$ in Figure 2 on the example from Figure 1. Here we fix $n = 40$, and investigate the behaviour as m increases. For the low frequencies (.1,1 and 3), we achieve a significantly more accurate approximation than in Figure 1. Note that Algorithm 3.6 requires $\mathcal{O}(mn\log n)$ operations, whereas Algorithm 2.4 with Gaussian elimination requires $\mathcal{O}\left(n^3\right)$ operations. Thus for all frequencies we achieve roughly machine precision with less computation.

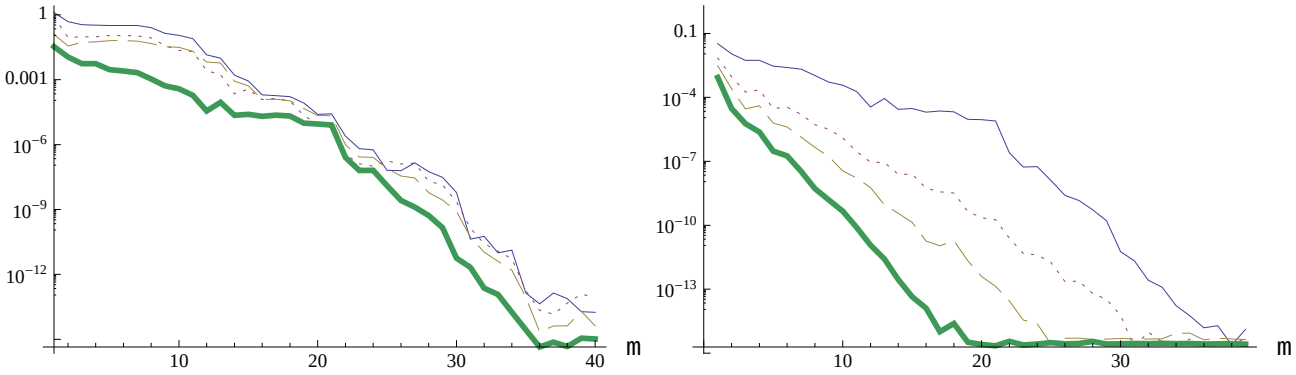

Figure 2: The error in approximating $\int_{-1}^{1} \frac{1}{x^2+1} \mathrm{e}^{\mathrm{i}\omega \sin\left(x+\frac{1}{4}\right)} \,\mathrm{d}x$ by $\mathcal{Q}^{\mathrm{M}}_{\omega,40,m}[f]$ for $\omega = 0.1$ (left, plain), 1 (left, dotted), 3 (left, dashed), 10 (both, thick), 30 (right, plain), 50 (right, dotted) and 100 (right, dashed).

In the following theorem we prove the asymptotic order of this method as $\omega \to \infty$. This was the motivation for using Chebyshev–Lobatto points, rather than the standard Chebyshev points: interpolation at the endpoints increases the asymptotic order from $\mathcal{O}\left(\omega^{-1}\right)$ to $\mathcal{O}\left(\omega^{-2}\right)$.

Theorem 3.7

$$\mathcal{I}_\omega[f] - \mathcal{Q}^{\mathrm{M}}_{\omega,n,m}[f] = \mathcal{O}\left(\omega^{-2}\right) \qquad as \qquad \omega \to \infty.$$

Proof: From (1.2), (2.2) and (3.1) we find that

$$\mathcal{Q}^{\mathrm{M}}_{\omega,n,m}[f] = \mathcal{I}_\omega[\mathcal{L}\boldsymbol{e}(x)^\top G^{-1}\boldsymbol{v}_m] = \mathcal{I}_\omega[\boldsymbol{e}(x)^\top \boldsymbol{L}\mathcal{T}G^{-1}\boldsymbol{v}_m] = \mathcal{I}_\omega[\boldsymbol{e}(x)^\top M\boldsymbol{v}_m];$$

hence,

$$\mathcal{I}_\omega[f] - \mathcal{Q}^{\mathrm{M}}_{\omega,n,m}[f] = \mathcal{I}_\omega[\boldsymbol{e}(x)^\top(\boldsymbol{f} - M\boldsymbol{v}_m)] + \mathcal{I}_\omega[f - \boldsymbol{e}(x)^\top \boldsymbol{f}].$$

Corollary 2.1 in [**24**] states that if $p_\omega, p'_\omega = \mathcal{O}\left(\omega^{-\alpha}\right)$ and $0 = p_\omega(\pm 1) = \cdots = p_\omega^{(s-1)}(\pm 1)$, then

$$\mathcal{I}_\omega[p_\omega] = \mathcal{O}\left(\omega^{-s-\alpha-1}\right).$$

Thus, with $p_\omega = f - \boldsymbol{e}(x)^\top \boldsymbol{f}$, $\alpha = 0$ and $s = 1$ we find that $\mathcal{I}_\omega[f - \boldsymbol{e}(x)^\top \boldsymbol{f}] = \mathcal{O}\left(\omega^{-2}\right)$. Furthermore, from Corollary 3.5 we know that $\boldsymbol{f} - M\boldsymbol{v}_m = \mathcal{O}\left(\omega^{-m}\right)$; hence, with $s = 0$ and since $\boldsymbol{e}(x)^\top(\boldsymbol{f} - M\boldsymbol{v}_m), \boldsymbol{e}(x)^\top D(\boldsymbol{f} - M\boldsymbol{v}_m) = \mathcal{O}\left(\omega^{-m}\right)$, $\mathcal{I}_\omega[\boldsymbol{e}(x)^\top(\boldsymbol{f} - M\boldsymbol{v}_m)] = \mathcal{O}\left(\omega^{-m-1}\right)$, which completes the proof.

Q.E.D.

The proof of this theorem explains the behaviour of the error in Figure 2. The error behaves like $\mathcal{O}\big(\omega^{-m}\big)$, $\omega \to \infty$, until the error is dominated by $\mathcal{I}_\omega[f - \boldsymbol{e}(x)^\top \boldsymbol{f}]$, i.e., the error in interpolating f at n Chebyshev–Lobatto points. Thus if n is large enough that f is resolved to machine precision, then the effective asymptotic order is $\mathcal{O}\big(\omega^{-m}\big)$.

4. Differential GMRES and convergence

There is an alternate way to view the method we have just constructed. In [**19**], a particular solution to (1.1) was found by applying GMRES not to a discretization of the differential equation, but, rather, to the differential equation itself. In other words, for $f, g \in \mathcal{C}^\infty[-1,1]$ and

$$\mathcal{M} = \mathcal{D}\frac{1}{g'} + \mathrm{i}\omega,$$

we find a function v_m in the Krylov subspace

$$\mathcal{K}_m[\mathcal{M}, f] = \operatorname{span}\left\{f, \mathcal{M}f, \ldots, \mathcal{M}^{m-1}f\right\}$$

that minimizes some seminorm

$$\|\mathcal{M}v_m - f\|.$$

The algorithm is essentially the same as Algorithm 3.2, where now an infinite-dimensional Arnoldi iteration uses a semi-inner product $\langle\cdot,\cdot\rangle$ to produce a quasimatrix $Q_m^{\mathcal{M}} = (q_1, \ldots, q_m)$ and Hessenberg matrix $H_m \in \mathbb{C}^{(m+1)\times m}$ such that (assuming $\|f\|$ is nonzero)

$$q_1 = \frac{f}{\|f\|} \qquad \text{and} \qquad \mathcal{M}Q_m^{\mathcal{M}} = Q_{m+1}^{\mathcal{M}} H_m.$$

Then we determine $\boldsymbol{d}_m^{\mathcal{M}}$ by minimizing the finite-dimensional norm

$$\left\|H_m \boldsymbol{d}_m^{\mathcal{M}} - \|f\|\,\boldsymbol{e}_1\right\|,$$

resulting in the method

$$\mathcal{Q}_{\omega,m}^{\mathcal{M}}[f] = \frac{\mathrm{e}^{\mathrm{i}\omega g(1)}}{g'(1)} Q_m^{\mathcal{M}}(1)\boldsymbol{d}_m^{\mathcal{M}} - \frac{\mathrm{e}^{\mathrm{i}\omega g(-1)}}{g'(-1)} Q_m^{\mathcal{M}}(-1)\boldsymbol{d}_m^{\mathcal{M}}.$$

When $g'(x) = x$, the GMRES–Levin collocation method can be thought of as replacing f by its n-degree Chebyshev polynomial $\boldsymbol{e}(x)^\top \boldsymbol{f}$, and apply differential GMRES to $\boldsymbol{e}(x)^\top \boldsymbol{f}$:

$$\mathcal{Q}_{\omega,n,m}^{\mathrm{M}}[f] = \mathcal{Q}_{\omega,m}^{\mathcal{M}}[\boldsymbol{e}(x)^\top \boldsymbol{f}].$$

This connection will allow us to prove convergence. Note that these convergence proofs only hold in the idealized case where there is no round-off error, see Section 6 for problems associated with letting n increase. The first result is for the convergence of differential GMRES:

Theorem 4.1 [19] *Suppose there exists a simply connected open set U in the complex plane containing zero such that*

$$w(t) = \frac{f(g^{-1}(t))}{g'(g^{-1}(t))} \mathrm{e}^{\mathrm{i}\omega t}$$

can be analytically continued throughout $U + [-1, 1]$. If there exists a sequence of balls $\{B_1, B_2, \ldots\} \subset U$ such that $\|w\|_{L_\infty[B_k+[-1,1]]} \to 0$ as $k \to \infty$, then

$$\|\mathcal{M} v_m - f\|_T \to 0 \qquad as \qquad m \to \infty.$$

.

When ω is real, we obtain a simpler condition: for $\epsilon > 0$ and a curve $p(t)$ such that $p(0) = -\epsilon$ and $\operatorname{Im} p(t) \to +\infty$, if

$$\frac{f(g^{-1}(t))}{g'(g^{-1}(t))}$$

can be analytically continued along $p(t) + [-1 - \epsilon, 1 + \epsilon]$ with at most exponential growth, then

$$\|\mathcal{M} v_m - f\|_T \to 0 \qquad \text{as} \qquad m \to \infty$$

for sufficiently large ω.

The following lemma proves that the error in the finite dimensional GMRES algorithm approaches the error for differential GMRES as $n \to \infty$.

Lemma 4.2 *Suppose that $f \in \mathcal{C}^{m+2}[-1, 1]$ and $g \in \mathcal{C}^\infty[-1, 1]$. Then*

$$\limsup_{n\to\infty} \|M \boldsymbol{v}_m - \boldsymbol{f}\| \le \|\mathcal{M} v_m - f\|_T .$$

Proof: We use the notation that Q_m denotes the orthonormal basis returned by Arnoldi iteration with M and $\boldsymbol{f}$ and $\boldsymbol{v}_m$ the resulting GMRES approximation. Let $Q_m^{\mathcal{M}} = (q_1, \ldots, q_m)$ be a quasimatrix whose columns form an orthonormal basis for $\mathcal{K}_m[\mathcal{M}, f]$, which is produced by Arnoldi iteration with the Chebyshev inner product. We can then write $v_m = Q_m^{\mathcal{M}} \boldsymbol{d}_m^{\mathcal{M}}$. From the definition of GMRES we know that

$$\|M \boldsymbol{v}_m - \boldsymbol{f}\| \le \left\| M Q_m \boldsymbol{d}_m^{\mathcal{M}} - \boldsymbol{f} \right\| .$$

But

$$\begin{aligned} \left\| M Q_m \boldsymbol{d}_m^{\mathcal{M}} - \boldsymbol{f} \right\| &= \left\| \boldsymbol{e}(x)^\top M Q_m \boldsymbol{d}_m^{\mathcal{M}} - \boldsymbol{e}(x)^\top \boldsymbol{f} \right\|_T \\ &\le \left\| \boldsymbol{e}(x)^\top M Q_m \boldsymbol{d}_m^{\mathcal{M}} - \mathcal{M} Q_m^{\mathcal{M}} \boldsymbol{d}_m^{\mathcal{M}} \right\|_T + \left\| \mathcal{M} Q_m^{\mathcal{M}} \boldsymbol{d}_m^{\mathcal{M}} - f \right\|_T \\ &\quad + \left\| f - \boldsymbol{e}(x)^\top \boldsymbol{f} \right\|_T . \end{aligned} \tag{4.1}$$

From the theory of polynomial interpolation, we know that, as $n \to \infty$,

$$\begin{aligned} \boldsymbol{e}(x)^\top \boldsymbol{f} &\to f(x), \\ \boldsymbol{e}(x)^\top M \boldsymbol{f} &\to \mathcal{M} f(x), \\ &\vdots \\ \boldsymbol{e}(x)^\top M^m \boldsymbol{f} &\to \mathcal{M}^m f(x). \end{aligned}$$

It follows immediately that the last norm in (4.1) goes to zero. We now show that the first norm goes to zero, or in other words $\boldsymbol{e}(x)^\top M Q_m \to \mathcal{M} Q_m$. Since $\|\boldsymbol{f}\| \to \|f\|_T$, it is evident that

$$\boldsymbol{e}(x)^\top M^k \boldsymbol{q}_1 = \boldsymbol{e}(x)^\top M^k \frac{\boldsymbol{f}}{\|\boldsymbol{f}\|} \to \frac{\mathcal{M}^k f}{\|f\|_T} = \mathcal{M}^k q_1 \qquad \text{for} \qquad k = 0, \ldots, m.$$

Now assume that

$$\boldsymbol{e}(x)^\top M^k \boldsymbol{q}_j \to \mathcal{M}^k q_j \qquad \text{for} \qquad k = 0, \ldots, m - j + 1.$$

From the Arnoldi algorithm [**25**] we know that $\boldsymbol{q}_{j+1}$ is equal to the normalization of (using $*$ to denote the conjugate transpose)

$$M\boldsymbol{q}_j - (\boldsymbol{q}_1^* M \boldsymbol{q}_j)\boldsymbol{q}_1 - \cdots - (\boldsymbol{q}_j^* M \boldsymbol{q}_j)\boldsymbol{q}_j$$

But

$$\begin{aligned} &\boldsymbol{e}(x)^\top M^k \left[M\boldsymbol{q}_j - (\boldsymbol{q}_1^* M \boldsymbol{q}_j)\boldsymbol{q}_1 - \cdots - (\boldsymbol{q}_j^* M \boldsymbol{q}_j)\boldsymbol{q}_j \right] \\ &\quad \to \mathcal{M}^k \left[\mathcal{M} q_j - \langle q_1, \mathcal{M} q_j \rangle_T \, q_1 - \cdots - \langle q_j, \mathcal{M} q_j \rangle_T \, q_j \right] \end{aligned} \qquad \text{for} \qquad k = 0, \ldots, m - j.$$

Normalizing both sides shows us that

$$\boldsymbol{e}(x)^\top M^k \boldsymbol{q}_{j+1} \to \mathcal{M}^k q_{j+1} \qquad \text{for} \qquad k = 0, \ldots, m - j.$$

Thence $\boldsymbol{e}(x)^\top M Q_m \to \mathcal{M} Q_m^{\mathcal{M}}$. It follows that

$$\limsup_{n \to \infty} \|M\boldsymbol{v}_m - \boldsymbol{f}\| \le \left\| \mathcal{M} Q_m^{\mathcal{M}} \boldsymbol{d}_m^{\mathcal{M}} - f \right\|_T = \|\mathcal{M} v_m - f\|_T \,.$$

Q.E.D.

Combining the preceding two lemmas we obtain a proof of convergence:

Theorem 4.3 *Assuming the conditions of Theorem 4.1 apply, we can choose integers n_m such that*

$$\mathcal{Q}^{\mathrm{M}}_{\omega, n_m, m}[f] \to \mathcal{I}_\omega[f] \qquad \text{as} \qquad m \to \infty.$$

Proof: Note that

$$\begin{aligned}\frac{1}{\sqrt{2}}\left|\mathcal{Q}^{\mathrm{M}}_{\omega,n,m}[f]-\mathcal{I}_\omega[f]\right| &= \frac{1}{\sqrt{2}}\left|\mathcal{I}_\omega[\boldsymbol{e}(x)^\top M\boldsymbol{v}_m - f]\right| \\ &\le \left\|\boldsymbol{e}(x)^\top M\boldsymbol{v}_m - f\right\|_2 \le \left\|\boldsymbol{e}(x)^\top M\boldsymbol{v}_m - f\right\|_T \\ &\le \|M\boldsymbol{v}_m - \boldsymbol{f}\| + \left\|\boldsymbol{e}(x)^\top\boldsymbol{f} - f\right\|_T \\ &\le \|\mathcal{M}v_m - f\|_T + \left\|\boldsymbol{e}(x)^\top\boldsymbol{f} - f\right\|_T + e_{m,n},\end{aligned}$$

where $e_{m,n} = \left|\|\mathcal{M}v_m - f\|_T - \|M\boldsymbol{v}_m - \boldsymbol{f}\|\right|$. Due to Theorem 4.1, we can choose m large enough so that $\|\mathcal{M}v_m - f\|_T < \epsilon$. We can also choose $n = n_m$ large enough so that $\left\|\boldsymbol{e}(x)^\top\boldsymbol{f} - f\right\|_T$ and e_{m,n_m} are also less than ϵ, which completes the proof. *Q.E.D.*

5. Stationary points

We now consider the case where g' vanishes at zero. In the following discussion, we assume that g itself also vanishes at zero. This is justified, as

$$\int_{-1}^{1} f\mathrm{e}^{\mathrm{i}\omega g}\,\mathrm{d}x = \mathrm{e}^{\mathrm{i}\omega g(0)}\int_{-1}^{1} f\mathrm{e}^{\mathrm{i}\omega[g-g(0)]}\,\mathrm{d}x.$$

To motivate the method, we first approach the problem in the differential GMRES context. Thus we want to multiply $\mathcal{L} = \mathcal{D} + \mathrm{i}\omega g'$ on the right so it takes the form $\mathcal{M} = \mathcal{M}_0 + \mathrm{i}\omega$ for some linear operator $\mathcal{M}_0 : \mathcal{C}^\infty[-1,1] \to \mathcal{C}^\infty[-1,1]$. In this case, the operator $\mathcal{D}\frac{1}{g'}$ introduces a singularity, hence fails the requirement that $\mathcal{M}_0$ must map $\mathcal{C}^\infty[-1,1]$ to $\mathcal{C}^\infty[-1,1]$. Instead, we will premultiply the operator so that the resulting singularity is removable.

Suppose we can find functions $\phi_{\omega,0},\ldots,\phi_{\omega,r-2} \in \mathcal{C}^\infty[-1,1]$ such that

$$\mathcal{D}^j\mathcal{L}\phi_k(0) = \delta_{k,j}, \qquad j = 0,\ldots,r-2,$$

where $\delta_{k,j}$ is the Kronecker delta and each $\mathcal{L}\phi_k$ is independent of ω (which we emphasize by dropping the dependence on ω in the notation). We can then define the operator

$$\mathcal{R}v = v(0)\mathcal{L}\phi_0 + \cdots + v^{(r-2)}(0)\mathcal{L}\phi_{r-2}.$$

When $g(x) = x^r$, our choice of $\phi_{\omega,k}$ will satisfy $\mathcal{L}\phi_k = \frac{x^k}{k!}$, and hence this reduces to the first $r-1$ terms of the Taylor series of v.

Now consider the operator

$$\mathcal{P}v = \frac{v - \mathcal{R}v}{g'} + \mathrm{i}\omega(v(0)\phi_{\omega,0} + \ldots + v^{(r-2)}(0)\phi_{\omega,r-2}),$$

For any $v \in \mathcal{C}^\infty[-1,1]$, $(v - \mathcal{R}v)/g'$ has a removable singularity at zero, thus

$$\begin{aligned}\mathcal{M}v = \mathcal{L}\mathcal{P}v &= \mathcal{D}\left[\frac{v-\mathcal{R}v}{g'}\right] + \mathrm{i}\omega g' \frac{v-\mathcal{R}v}{g'} + \mathrm{i}\omega(v(0)\mathcal{L}\phi_0 + \ldots + v^{(r-2)}(0)\mathcal{L}\phi_{r-2}) \\ &= \mathcal{D}\left[\frac{v-\mathcal{R}v}{g'}\right] + \mathrm{i}\omega v \end{aligned} \tag{5.1}$$

does indeed map $\mathcal{C}^\infty[-1,1]$ to $\mathcal{C}^\infty[-1,1]$. As before, solving $\mathcal{M}v = f$ allows us to compute $\mathcal{I}_\omega[f]$:

$$\mathcal{I}_\omega[f] = \mathcal{I}_\omega[\mathcal{M}v] = \mathcal{I}_\omega[\mathcal{L}\mathcal{P}v] = \mathcal{P}v(1)\mathrm{e}^{\mathrm{i}\omega g(1)} - \mathcal{P}v(-1)\mathrm{e}^{\mathrm{i}\omega g(-1)}.$$

The remaining question is how to construct the functions ϕ_k. One choice is found in [**21**], which we generalize here to be valid in the complex plane. Define

$$\begin{aligned}\sqrt[r]{g}\,(x) &= g^{(r)}(0)^{\frac{1}{r}} x \left(\frac{g(x)}{g^{(r)}(0)x^r}\right)^{\frac{1}{r}}, \qquad x \neq 0, \\ \sqrt[r]{g}\,(0) &= 0.\end{aligned}$$

The notation is justified since

$$\sqrt[r]{g}\,(x)^r = g^{(r)}(0)x^r \frac{g(x)}{g^{(r)}(0)x^r} = g(x).$$

Since $g(x) = \frac{g^{(r)}(0)}{r!}x^r + \mathcal{O}\left(x^{r+1}\right)$, $\frac{g(x)}{g^{(r)}(0)x^r} > 0$ near zero, hence $\sqrt[r]{g}$ is an analytic function in $[-1,1]$ as long as $g(x)$ only vanishes at zero. When $g(x) > 0$ for $x > 0$, $\sqrt[r]{g}$ can be viewed as an analytic continuation of $g(x)^{1/r}$ from $(0,1]$ to $[-1,1]$.

We now define

$$\tilde{\phi}_{\omega,k}(x) = \mathrm{e}^{-\mathrm{i}\omega g(x)} \int_0^{\sqrt[r]{g}\,(x)} t^k \mathrm{e}^{\mathrm{i}\omega t^r}\,\mathrm{d}t.$$

For all k and r, the moment in this expression can be determined from the definition of the derivative of the incomplete Gamma function, cf. [**1**], giving us:

$$\int_0^x t^k \mathrm{e}^{\mathrm{i}\omega t^r}\,\mathrm{d}t = \frac{(-\mathrm{i}\omega x^r)^{\frac{-1-k}{r}} x^{1+k}}{r}\left[\Gamma\left(\frac{1+k}{r}\right) - \Gamma\left(\frac{1+k}{r}, -\mathrm{i}\omega x^r\right)\right].$$

(All branch cuts are the standard choices). Alternatively, when $r = 2$ we only need the first moment, which can be expressed in terms of the error function:

$$\int_0^x \mathrm{e}^{\mathrm{i}\omega t^2}\,\mathrm{d}t = \frac{1+\mathrm{i}}{2}\sqrt{\frac{\pi}{2\omega}}\,\mathrm{erf}\,\mathrm{e}^{\frac{3\pi}{4}\mathrm{i}}\sqrt{\omega}\,x. \qquad [\mathbf{1}]$$

Differentiating we obtain

$$\begin{aligned}\sqrt[r]{g}\,'(x) &= g^{(r)}(0)^{\frac{1}{r}}\left[\left(\frac{g(x)}{g^{(r)}(0)x^r}\right)^{\frac{1}{r}} + \frac{1}{r}x\left(\frac{g(x)}{g^{(r)}(0)x^r}\right)^{\frac{1}{r}-1}\left(\frac{g'(x)}{g^{(r)}(0)x^r} - r\frac{g(x)}{g^{(r)}(0)x^{r-1}}\right)\right]\\ &= \frac{1}{rg}\sqrt[r]{g}\,g'.\end{aligned}$$

It follows that

$$\begin{aligned}\mathcal{L}\tilde{\phi}_k &= \tilde{\phi}'_{\omega,k} + \mathrm{i}\omega g'\tilde{\phi}_{\omega,k}\\ &= \mathrm{e}^{-\mathrm{i}\omega g}\left[-\mathrm{i}\omega g'\int_0^{\sqrt[r]{g}} t^k\mathrm{e}^{\mathrm{i}\omega t^r}\,\mathrm{d}t + \sqrt[r]{g}^{\,k}\,\sqrt[r]{g}\,'\mathrm{e}^{\mathrm{i}\omega\sqrt[r]{g}^{\,r}} + \mathrm{i}\omega g'\int_0^{\sqrt[r]{g}} t^k\mathrm{e}^{\mathrm{i}\omega t^r}\,\mathrm{d}t\right]\\ &= \sqrt[r]{g}^{\,k+1}\frac{g'}{rg},\end{aligned}$$

which is, as desired, independent of ω. A straightforward calculation shows that $\mathcal{D}^j\mathcal{L}\tilde{\phi}_k(0) = 0$ for $j < k$, and $\mathcal{D}^j\mathcal{L}\tilde{\phi}_j(0) \neq 0$. Thus we can define $\phi_{\omega,k}$ as follows:

$$\begin{aligned}\phi_{\omega,r-2} &= \frac{\tilde{\phi}_{\omega,r-2}}{\mathcal{D}^{r-2}\mathcal{L}\tilde{\phi}_{r-2}(0)},\\ \phi_{\omega,r-3} &= \frac{1}{\mathcal{D}^{r-3}\mathcal{L}\tilde{\phi}_{r-3}(0)}\left[\tilde{\phi}_{\omega,r-3} - \tilde{\phi}^{(r-2)}_{\omega,r-3}(0)\phi_{r-2}\right],\\ \phi_{\omega,r-4} &= \frac{1}{\mathcal{D}^{r-4}\mathcal{L}\tilde{\phi}_{r-4}(0)}\left[\tilde{\phi}_{\omega,r-4} - \tilde{\phi}^{(r-3)}_{\omega,r-4}(0)\phi_{\omega,r-3} - \tilde{\phi}^{(r-2)}_{\omega,r-4}(0)\phi_{r-2}\right],\\ &\vdots\\ \phi_{\omega,0} &= \frac{1}{\mathcal{L}\tilde{\phi}_0(0)}\left[\tilde{\phi}_{\omega,0} - \sum_{k=1}^{r-2}\tilde{\phi}^{(k)}_{\omega,0}(0)\phi_{\omega,k}\right].\end{aligned}$$

With this in hand, we can define a differential GMRES algorithm for integrals with stationary points:

Algorithm 5.1 Differential GMRES with stationary points

Given $f, g \in \mathcal{C}^\infty[-1,1]$ such that $g'(0) = g''(0) = \ldots = g^{(r-1)}(0)$, $g^{(r)}(0) \neq 0$, $g'(x) \neq 0$ and $g(x) - g(0) \neq 0$ for $0 < |x| \leq 1$, and a semi-inner product $\langle\cdot,\cdot\rangle$ defined over $\mathcal{K}_m[\mathcal{M}, f]$, compute $\mathcal{Q}^{\mathcal{M}}_{\omega,m}[f]$ as follows:

1: If $g(0) \neq 0$, then run the algorithm with $g - g(0)$ in place of g to determine $\mathcal{Q}^{\mathcal{M}}_{\omega,m}[f]$, and return

$$\mathrm{e}^{\mathrm{i}\omega g(0)}\mathcal{Q}^{\mathcal{M}}_{\omega,m}[f];\ \text{otherwise,}$$

2: Run Arnoldi iteration on $\mathcal{M}_0 = \mathcal{D}\frac{1}{g'}[I - \mathcal{R}]$ and f;

3: Define $H_{m,\omega} = \tilde{H}_m + \mathrm{i}\omega I_{m+1,m}$;

4: Use least squares to find $\boldsymbol{d}_m^{\mathcal{M}} \in \mathbb{C}^m$ which minimizes the norm

$$\left\| H_{m,\omega}\boldsymbol{d}_m^{\mathcal{M}} - \|f\|\,\boldsymbol{e}_1 \right\|;$$

5: Define $v = Q_m^{\mathcal{M}}\boldsymbol{d}_m^{\mathcal{M}}$ and

$$\begin{aligned}\mathcal{Q}_{\omega,m}^{\mathcal{M}}[f] &= \mathrm{e}^{\mathrm{i}\omega g(1)}\mathcal{P}v(1) - \mathrm{e}^{\mathrm{i}\omega g(-1)}\mathcal{P}v(-1) \\ &= \mathrm{e}^{\mathrm{i}\omega g(1)}\frac{v(1) - \mathcal{R}v(1)}{g'(1)} - \mathrm{e}^{\mathrm{i}\omega g(-1)}\frac{v(-1) - \mathcal{R}v(-1)}{g'(-1)} \\ &\quad + \mathrm{i}\omega\boldsymbol{\mu}_\omega^\top \begin{pmatrix} v(0) \\ \vdots \\ v^{(r-2)}(0) \end{pmatrix}\end{aligned}$$

for

$$\boldsymbol{\mu}_\omega = \begin{pmatrix} \phi_{\omega,0}(1)\mathrm{e}^{\mathrm{i}\omega g(1)} - \phi_{\omega,0}(-1)\mathrm{e}^{\mathrm{i}\omega g(-1)} \\ \vdots \\ \phi_{\omega,r-2}(1)\mathrm{e}^{\mathrm{i}\omega g(1)} - \phi_{\omega,r-2}(-1)\mathrm{e}^{\mathrm{i}\omega g(-1)} \end{pmatrix}.$$

There is a significant problem with this algorithm, even if we assume that derivatives of f and g are easily computable. The removable singularity introduced is a numerical and logistical nightmare. Using the L_2 inner product computed with an adaptive quadrature scheme will fail, as near the removable singularity round-off error explodes. If we instead use a semi-inner product based on Gaussian or Clenshaw–Curtis quadrature with a fixed number of sample points, we are still left with the problem of computing $Q_m^{\mathcal{M}}$ and its derivatives at zero. This can be accomplished with L'Hopital's rule, but is too cumbersome to be practical.

To avoid these issues, we will convert the problem to a finite-dimensional version *à la* Section 2 and Section 3. We assume the first r derivatives of g are still available, and construct $\phi_{\omega,0}, \ldots, \phi_{\omega,r-2}$ as we did before.

Note that, for $\boldsymbol{v} \in \mathbb{C}^n$, we define R by

$$\begin{aligned}(\mathcal{R}\boldsymbol{e}(x)^\top\boldsymbol{v})(\boldsymbol{x}^{\mathrm{T}}) &= (\mathcal{L}\phi_0(\boldsymbol{x}^{\mathrm{T}})\boldsymbol{e}(0)^\top + \ldots + \mathcal{L}\phi_{r-2}(\boldsymbol{x}^{\mathrm{T}})\boldsymbol{e}(0)^\top D^{r-2})\boldsymbol{v} \\ &= R\boldsymbol{v}.\end{aligned}$$

At first glance, a finite-dimensional version of (5.1) seems clear:

$$M \stackrel{?}{=} DG^{-1}[I - R] + \mathrm{i}\omega.$$

Indeed, when n is even, this is the definition we use. When n is odd, however, G becomes singular, as the $((n+1)/2,(n+1)/2)$ entry of G is zero. Thus we must utilize L'Hopital's rule to determine the $((n+1)/2,(n+1)/2)$ entry of $M\boldsymbol{v}$. Define

$$\tilde{G} = \operatorname{diag}\left(g'(-1),\ldots,g'(\boldsymbol{e}_{\frac{n}{2}-1}^{\top}\boldsymbol{x}^{\mathrm{T}}),1,g'(\boldsymbol{e}_{\frac{n}{2}+1}^{\top}\boldsymbol{x}^{\mathrm{T}})\ldots,g'(1)\right).$$

Then, for $\boldsymbol{v} = v(\boldsymbol{x}^{\mathrm{T}})$,

$$\mathcal{M}v(\boldsymbol{x}^{\mathrm{T}}) \approx \left(D\tilde{G}^{-1}[I-R] + \mathrm{i}\omega\right)\boldsymbol{v}$$

is valid for every entry except the $(n+1)/2$ entry. At zero, we utilize L'Hopital's rule:

$$\frac{v(0)-Rv(0)}{g'} = \frac{\mathcal{D}^{r-1}(v(0)-\mathcal{R}v(0))}{g^{(r)}(0)} \approx \frac{1}{g^{(r)}(0)}D^{r-1}(I-R)\boldsymbol{v}.$$

Thus

$$\mathcal{M}v(\boldsymbol{x}^{\mathrm{T}}) \approx M\boldsymbol{v}$$

for

$$M = D\left[\tilde{G}^{-1} + \frac{1}{g^{(r)}(0)}\boldsymbol{e}(0)\boldsymbol{e}(0)^{\top}D^{r-1}\right](I-R) + \mathrm{i}\omega,$$

where we use $\boldsymbol{e}(0)$ to denote $\boldsymbol{e}_{(n+1)/2}$. This motivates the following algorithm:

Algorithm 5.2 GMRES–Levin collocation method for stationary points

Given $f \in \mathcal{C}^0[-1,1]$ and $g \in \mathcal{C}^r[-1,1]$ such that $g'(0) = g''(0) = \ldots = g^{(r-1)}(0)$, $g^{(r)}(0) \neq 0$, $g'(x) \neq 0$ and $g(x)-g(0) \neq 0$ for $0 < |x| \leq 1$, compute $\mathcal{Q}^{\mathrm{M}}_{\omega,n,m}[f]$ as follows:

1: If $g(0) \neq 0$, then run the algorithm with $g-g(0)$ in place of g to determine $\mathcal{Q}^{\mathrm{M}}_{\omega,n,m}[f]$, and return

$$\mathrm{e}^{\mathrm{i}\omega g(0)}\mathcal{Q}^{\mathrm{M}}_{\omega,n,m}[f];\ \text{otherwise,}$$

2: Define

$$M_0 = \begin{cases} DG^{-1}(I-R) & \text{if } n \text{ is even,} \\ D\left[\tilde{G}^{-1} + \frac{1}{g^{(r)}(0)}\boldsymbol{e}(0)\boldsymbol{e}(0)^{\top}D^{r-1}\right](I-R) & \text{otherwise;} \end{cases}$$

3: Precompute Q_m and $\tilde{H}_m$ using Arnoldi iteration with M_0 and $\boldsymbol{f} = f\left(\boldsymbol{x}^{\mathrm{T}}\right)$;

4: Define $H_{m,\omega} = \tilde{H}_n + \mathrm{i}\omega I_{n+1,n}$;

5: Use least squares to find $\boldsymbol{d}_m \in \mathbb{C}^m$ which minimizes the norm

$$\left\|H_{m,\omega}\boldsymbol{d}_m - \|\boldsymbol{f}\|\,\boldsymbol{e}_1\right\|;$$

6: For $\boldsymbol{v}_m = Q_m \boldsymbol{d}_m$ and $\boldsymbol{\mu}_\omega$ as defined in Algorithm 5.1, define

$$\mathcal{Q}^{\mathrm{M}}_{\omega,n,m}[f] = \left(\frac{-\mathrm{e}^{\mathrm{i}\omega g(-1)}}{g'(-1)}, 0, \ldots, 0, \frac{\mathrm{e}^{\mathrm{i}\omega g(1)}}{g'(1)}\right)(I-R)\boldsymbol{v}_m + \mathrm{i}\omega \boldsymbol{\mu}_\omega^\top \begin{pmatrix} \boldsymbol{e}(0)^\top \\ \boldsymbol{e}(0)^\top D \\ \vdots \\ \boldsymbol{e}(0)^\top D^{r-2} \end{pmatrix} \boldsymbol{v}_m.$$

Q.E.D.

As before we only need to compute the Arnoldi iteration and evaluate the integrand once for all values of ω. There is now, however, the additional work of computing $\boldsymbol{\mu}_\omega$, which depends on computing the incomplete Gamma function or error function. Both of these can be computed efficiently, using methods described in [**5**, **16**].

Assuming that $\mathcal{Q}^{\mathcal{M}}_{\omega,m}[f] \to \mathcal{I}_\omega[f]$, similar logic to Section 4 should prove the convergence of this method. Unfortunately, a convergence prove for $\mathcal{Q}^{\mathcal{M}}_{\omega,m}[f]$ is unknown, and we leave it as an open problem.

Computing Bessel functions

We will employ this the method to compute the Bessel function throughout the complex plane. When ν is an integer, we have the integral representation

$$\begin{aligned}
\mathrm{J}_\nu(z) &= \frac{1}{\pi}\int_0^\pi \cos(\nu t - z\sin t)\,\mathrm{d}t \qquad [\mathbf{1}] \\
&= \frac{1}{2}\int_{-1}^1 \cos\left(\nu\pi\frac{x+1}{2} - z\sin\pi\frac{x+1}{2}\right)\mathrm{d}x \\
&= \frac{1}{4}\int_{-1}^1 \left[\mathrm{e}^{\mathrm{i}\left(\nu\pi\frac{x+1}{2} - z\sin\pi\frac{x+1}{2}\right)} + \mathrm{e}^{-\mathrm{i}\left(\nu\pi\frac{x+1}{2} - z\sin\pi\frac{x+1}{2}\right)}\right]\mathrm{d}x \\
&= \frac{\mathrm{e}^{\mathrm{i}z - \mathrm{i}\nu\pi/2}}{4}\mathcal{I}_{-z}[f_\nu] + \frac{\mathrm{e}^{-\mathrm{i}z + \mathrm{i}\nu\pi/2}}{4}\mathcal{I}_z[f_\nu],
\end{aligned}$$

where $f_\nu(x) = \mathrm{e}^{-\mathrm{i}\nu\pi\frac{x}{2}}$ and

$$\mathcal{I}_z[f] = \int_{-1}^1 f(x)\mathrm{e}^{\mathrm{i}z\left(1-\sin\pi\frac{x+1}{2}\right)}\,\mathrm{d}x.$$

Thus we approximate $\mathrm{J}_\nu(z)$ by

$$\mathrm{J}^M_{\nu,n,m}(z) = \frac{\mathrm{e}^{\mathrm{i}z - \mathrm{i}\nu\pi/2}}{4}\mathcal{Q}^{\mathrm{M}}_{-z,n,m}[f_\nu] + \frac{\mathrm{e}^{-\mathrm{i}z + \mathrm{i}\nu\pi/2}}{4}\mathcal{Q}^{\mathrm{M}}_{z,n,m}[f_\nu],$$

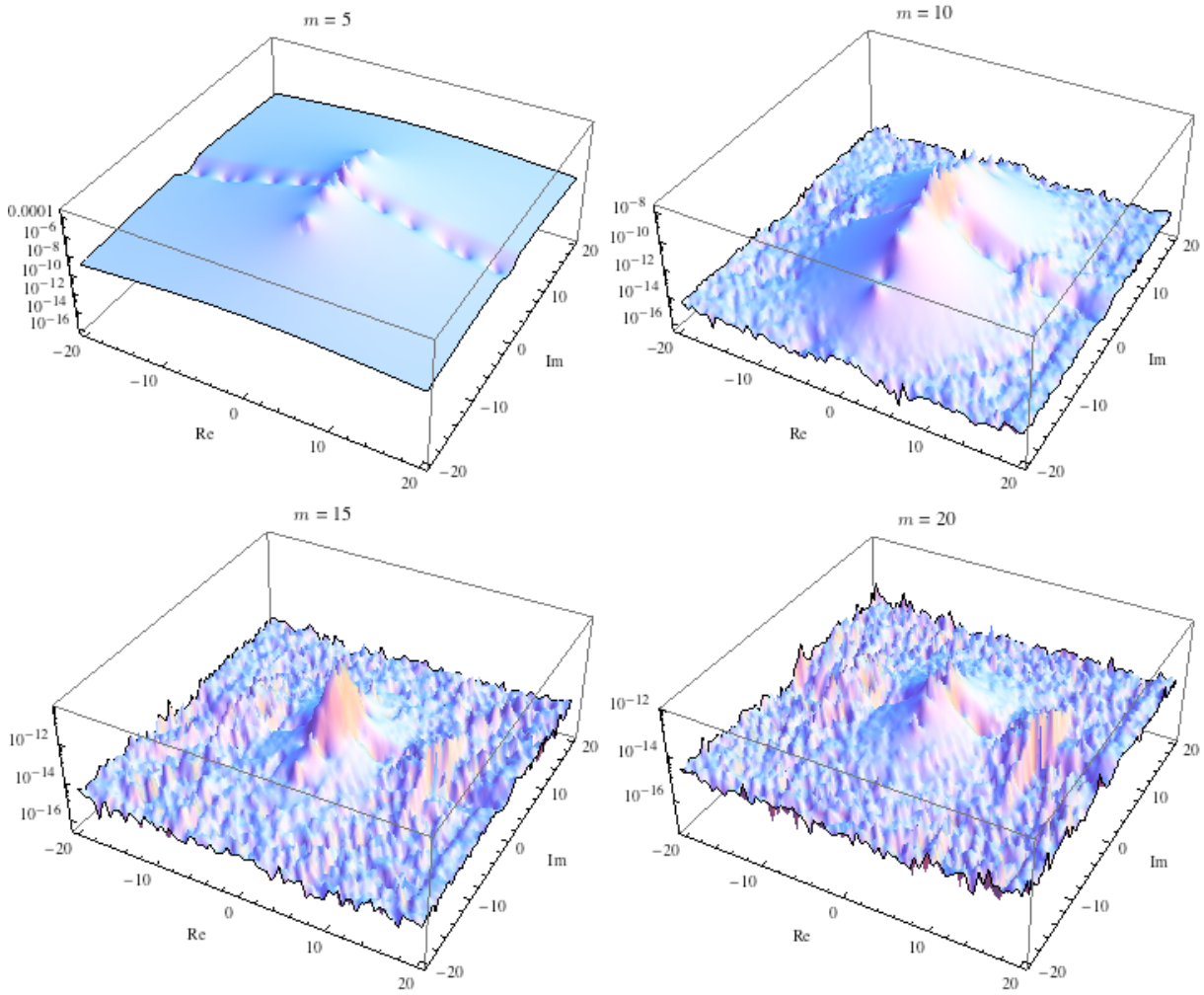

Figure 3: Error in approximating $\mathrm{J}_1(z)$ by $\mathrm{J}^M_{1,25,m}(z)$ for complex z and four choices of m. The error is the absolute error when $|\mathrm{J}_1(z)| < 1$ and the relative error otherwise.

Note that for fixed ν, n and m, the Arnoldi process need only be performed once to compute $\mathrm{J}^M_{\nu,n,m}(z)$ for all values of z throughout the complex plane, though we do need to compute the error function for each additional choice of z.

Figure 3 demonstrates the method for approximating $\mathrm{J}_1(z)$. Note that we are only using 25 function evaluations, and the method converges to machine precision throughout the complex plane.

Because numerical methods for approximating Bessel functions have been developed and refined considerably, other methods should be faster, see [**16**] for a list of publications. Such methods are typically based on a combination of asymptotic expansions, recurrence relationships and contour integration in the complex plane [**9**].

Remark: As a rough idea of the speed of $\mathrm{J}^M_{\nu,n,m}(z)$, if the Arnoldi iteration is precomputed

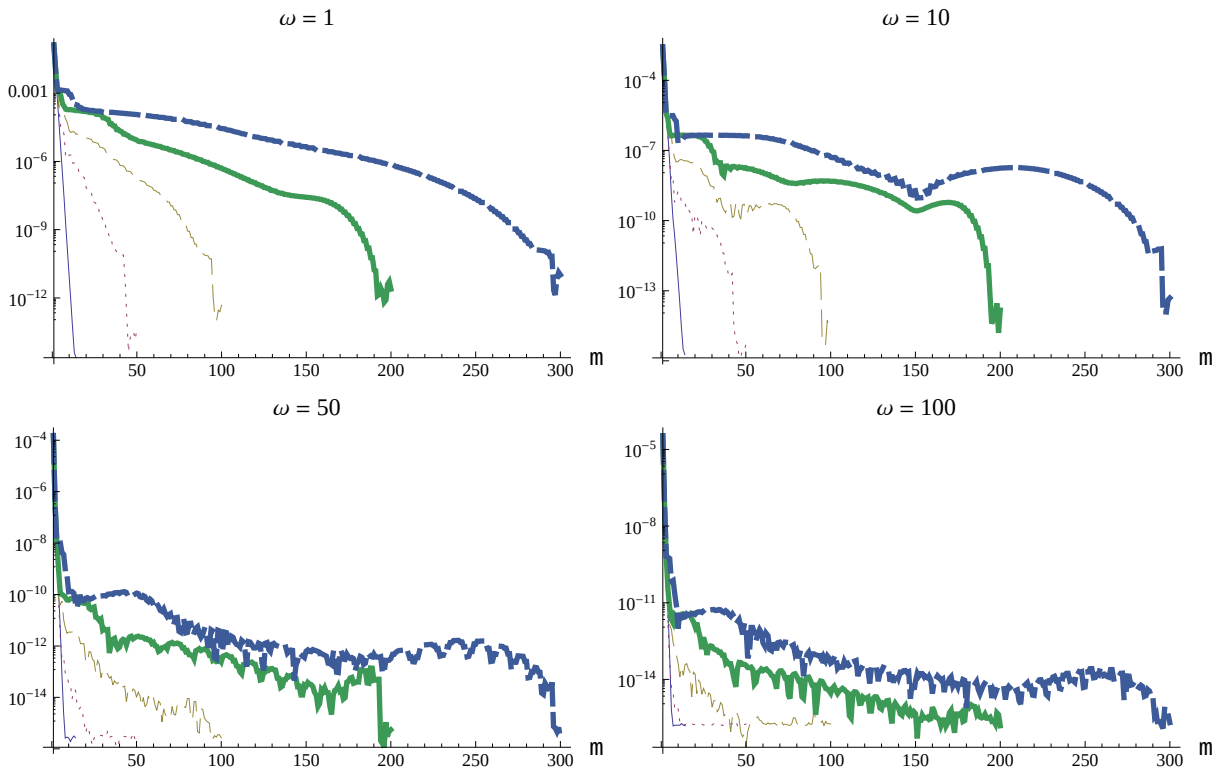

Figure 4: The error in approximating $\int_{-1}^{1} \frac{1}{x+2}\,\mathrm{e}^{\mathrm{i}\omega x}\,\mathrm{d}x$ by $\mathcal{Q}^{\mathcal{M}}_{\omega,m}[f]$ (plain) and $\mathcal{Q}^{\mathrm{M}}_{\omega,n,m}[f]$, for $n = 50$ (plain), 100 (dotted), 200 (dashed) and 400 (thick), for four choices of ω.

then the method takes about 0.01 seconds on my 2005 PowerBook G4 in MATHEMATICA 6.0 for each additional z, using the built-in routine to compute the error functions. For comparison, the built-in routine for computing J_1 typically takes about 0.001 seconds.

6. Numerical problems for large degree polynomials

We will now briefly investigate how the method breaks down as n becomes large. In Figure 4, we approximate the integral

$$\int_{-1}^{1} \frac{1}{x+2}\,\mathrm{e}^{\mathrm{i}\omega x}\,\mathrm{d}x$$

using $\mathcal{Q}^{\mathrm{M}}_{\omega,n,m}[f]$. It can clearly be seen in this figure that the number of Arnoldi iterations required increases as n increases. On the other hand, in the infinite-dimensional case (i.e., $n = \infty$), differential GMRES converges nicely.

This phenomenon can be explained heuristically by round-off error in the DCT. As touched on in Section 4, the method can be considered as replacing f by the polynomial $p_n = \boldsymbol{e}(x)^{\top}\boldsymbol{f}$, and applying the differential GMRES method to p. But as n becomes large,

the nth Chebyshev coefficient of p_n will not go to zero, but rather to something on the order of machine epsilon. As $n \to \infty$, T_n blows up in the complex plane, and in the limit, Theorem 4.1 will not apply.

It should be emphasized that these errors are not caused by the fact that the derivatives of p_n do not accurately approximate the derivatives of f. Successfully evaluating $\mathcal{I}_\omega[p_n]$ is sufficient to approximate $\mathcal{I}_\omega[f]$, and the former does not depend on f itself. Replacing Chebyshev series by another approximant which does not blow up in the complex plane for large n should hopefully resolve the issue. An alternative being investigate is to use a Chebyshev–Padé approximation, as rationals decay in the complex plane.

We remark that, even with these problems, the GMRES–Levin collation method (Algorithm 3.6) with $n = m$ is still significantly more accurate than the Levin–Chebyshev collocation method (Algorithm 2.4) with Gaussian elimination. Furthermore, it only takes $\mathcal{O}\left(n^2 \log n\right)$ operations; hence, it is more efficient than either Gaussian elimination or the TSVD approach of [**15**].

7. Conclusion

We have constructed a numerically stable method for computing highly oscillatory integrals with or without stationary points, which does not require knowledge of the derivatives of f. We retain the most important computational property of the asymptotic methods: high frequency requires less work. Applying this algorithm to the computation of Bessel functions resulted in a scheme which, in some sense, converges everywhere in the complex plane.

In the future, we hope to generalize these results for computing multivariate integrals with stationary points over general domains. A multivariate oscillatory integral without stationary points over a domain Ω can be expressed as an oscillatory integral with an exponential oscillator over the boundary of Ω [**23**]. If we tried to combine the results of Section 5 with this approach, we would obtain an integral over the boundary in terms of the incomplete Gamma function. Thus to generalize the method for bivariate integrals, we would need to be able to compute highly oscillatory univariate integrals whose kernel is the incomplete Gamma function. How to accomplish this requires investigation.

We could apply this method to other oscillatory kernels. A Levin collocation method has been constructed for oscillatory integrals whose kernel satisfies an ODE [**13**, **22**]. Generalizing Algorithm 3.6 for the case where the domain of integration does not contain a turning point of the ODE would be straightforward. Perhaps a generalization of Algorithm 5.2 can be constructed for kernels with turning points. For kernels which satisfy second order ODEs, we would most likely have to replace the incomplete Gamma function with a special function that satisfies

$$\phi'' + \omega^2 x^r \phi = x^k \qquad k = 0, \ldots, r-1.$$

When $r = 1$, these are known as Scorer's functions [**18**].

References

[**1**] Abramowitz, M. and Stegun, I., *Handbook of Mathematical Functions*, National Bureau of Standards Appl. Math. Series, #55, U.S. Govt. Printing Office, Washington, D.C., 1970.

[**2**] Battles, Z. and Trefethen, L.N., An extension of MATLAB to continuous functions and operators, *SIAM J. Sci. Comput.* **25** (2004), 1743–1770.

[**3**] Berrut, J.-P. and Trefethen, L.N., Barycentric lagrange interpolation, *SIAM Review* **46** (2004), 501–517.

[**4**] Clenshaw, C. W. and Curtis, A. R., A method for numerical integration on an automatic computer, *Numer. Math.* **2** (1960), 197–205.

[**5**] Cody, W. J., *An Overview of Software Development*, Lecture Notes in Mathematics, 506, Numerical Analysis, Dundee G.A. Watson (ed.), Springer-Verlag, Berlin, 1976.

[**6**] Datta, B.N. and Saad, Y., Arnoldi methods for large Sylvester-like observer matrix equations and an associated algorithm for partial pole assignment, *Linear Algebra Appl* **154–156** (1991), 225–244.

[**7**] Deaño, A. and Huybrechs, D., Complex Gaussian quadrature of oscillatory integrals, *Numer. Math.*, to appear.

[**8**] Frommer, A. and Glässner, U., Restarted GMRES for shifted linear systems, *SIAM J. Sci. Comput.* **19** (1998), 15–26.

[**9**] Gil, A., Segura, J. and Temme, N.M., *Numerical Methods for Special Functions*, SIAM, 2007.

[**10**] Huybrechs, D. and Vandewalle, S., On the evaluation of highly oscillatory integrals by analytic continuation, *SIAM J. Num. Anal.* **44** (2006), 1026–1048.

[**11**] Iserles, A. and Nørsett, S.P., Efficient quadrature of highly oscillatory integrals using derivatives, *Proceedings Royal Soc. A* **461** (2005), 1383–1399.

[**12**] Iserles, A. and Nørsett, S.P., On quadrature methods for highly oscillatory integrals and their implementation, *BIT* **44** (2004), 755–772.

[**13**] Levin, D., Analysis of a collocation method for integrating rapidly oscillatory functions, *J. Comput. Appl. Maths* **78** (1997), 131–138.

[**14**] Levin, D., Procedures for computing one and two-dimensional integrals of functions with rapid irregular oscillations, *Math. Comp.* **38** (1982), no. 158 531–538.

[**15**] Jianbing, L., XueSong, W. and Tao, W., A universal solution to one-dimensional oscillatory integrals, *Sci China Ser F-Inf Sci* **51** (2008), 1614–1622.

[**16**] Lozier, D.W. and Olver, F.W.J., Numerical evaluation of special functions, *in*: *AMS Proceedings of Symposia in Applied Mathematics*, ed: W. Gautschi, 1994, 79–125.

[**17**] Melenk, J.M., On the convergence of Filon quadrature, *J. Comput. Appl. Maths*, to appear.

[**18**] Olver, F.W.J., *Asymptotics and Special Functions*, Academic Press, New York, 1974.

[**19**] Olver, S., Shifted GMRES for oscillatory integrals, preprint, NA-08/19, Computing Laboratory, Oxford Unversity.

[**20**] Olver, S., GMRES for the differentiation operator, preprint, NA-08/06, Computing Laboratory, Oxford University.

[**21**] Olver, S., Moment-free numerical approximation of highly oscillatory integrals with stationary points, *Euro. J. Appl. Math.* **18** (2007), 435–447.

[**22**] Olver, S., Numerical approximation of vector-valued highly oscillatory integrals, *BIT* **47** (2007), 637–655.

[**23**] Olver, S., On the quadrature of multivariate highly oscillatory integrals over non-polytope domains, *Numer. Math.* **103** (2006), 643–665.

[**24**] Olver, S., Moment-free numerical integration of highly oscillatory functions, *IMA J. Num. Anal.* **26** (2006), 213–227.

[**25**] Saad, Y., *Iterative Methods for Sparse Linear Systems*, SIAM, 2003.

[**26**] Saad, Y. and Schultz, M.H., GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems., *SIAM J. Sci. Stat. Comput.* **7** (1986), 856–869.

[**27**] Trefethen, L.N., Is Gauss quadrature better than Clenshaw-Curtis?, *SIAM Review* **50** (2008), 67–87.

[**28**] Trefethen, L.N. and Bau III, D., *Numerical Linear Algebra*, SIAM, 1997, 191–192.

[**29**] Xiang, S., Efficient Filon-type methods for $\int_a^b f(x)\mathrm{e}^{\mathrm{i}\omega g(x)}\,\mathrm{d}x$, *Numer. Math.* **105** (2007), 633–658.