SIAM J. SCI. COMPUT.
Vol. 41, No. 5, pp. A2836–A2849


# IMPROVED RESOLUTION OF BOUNDARY LAYERS FOR SPECTRAL COLLOCATION*

CONOR MCCOID† AND MANFRED R. TRUMMER†

**Abstract.** We propose a new algorithm to improve the accuracy of spectral methods for singularly perturbed two-point boundary value problems. Driscoll and Hale [*J. Numer. Anal.*, 36 (2016), pp. 108–132] suggest resampling as an alternative to row replacement when including boundary conditions. Testing this with an iterated sine-transformation [T. Tang and M. R. Trummer [*SIAM J. Sci. Comput.*, 17 (1996), pp. 430–438] designed for boundary layers reveals artificial boundary conditions imposed by the transformation. The transformation is regularized to prevent this. The new regularized sine-transformation is employed to solve boundary value problems with and without resampling. It shows superior accuracy provided the regularization parameter is chosen from an optimal range.

**Key words.** spectral methods, singular perturbation, boundary value problems, coordinate transformation, Chebyshev collocation, boundary layers

**AMS subject classifications.** 65N35, 65L10

**DOI.** 10.1137/18M1211015

## 1. Introduction.

We are investigating spectral collocation methods for solving two-point boundary value problems. In particular, we are interested in singularly perturbed problems whose solutions exhibit thin boundary layers.

Including boundary conditions in spectral collocation methods poses a series of issues. One straightforward way to include boundary conditions is to remove rows of the spectral collocation differentiation matrix and replace them with the boundary conditions [7]. This leads to an important question: Which rows should be removed? Standard practice is to replace the first and last rows for second-order ODEs, but no standard practice exists for any other order.

Driscoll and Hale [7] suggest a resampling method to deal with the ambiguity of removing rows from ODE matrices. This involves evaluating the underlying interpolating polynomial at a new set of points. The cardinality of this new set of points is equal to the cardinality of the set of Chebyshev points minus the order of the ODE. While this incurs a certain amount of round-off error, it eliminates the need to remove rows; instead, the boundary conditions can be directly concatenated.

We now seek to apply this resampling procedure to problems involving boundary layers. To resolve boundary layers it is often beneficial to use a coordinate transformation. This causes collocation points to become very close. The goal of this paper is to see if resampling can be used alongside a coordinate transformation to both resolve the boundary layers and eliminate the ambiguity of row removal. Specifically, we examine the iterated sine-transformation from [11].

For comparison we also consider rational spectral collocation (RSC) [1] combined with the coordinate transformations presented here. Like resampling, the RSC

*Submitted to the journal's Methods and Algorithms for Scientific Computing section September 4, 2018; accepted for publication (in revised form) June 10, 2019; published electronically September 12, 2019.

https://doi.org/10.1137/18M1211015

**Funding:** This work was funded by the Canadian National Science and Engineering Research Council (NSERC) Discovery grant RGPIN 05758.

†Department of Mathematics, Simon Fraser University, Burnaby, V5A 1S6, BC, Canada (cmccoid@sfu.ca, mrt@math.sfu.ca).

method changes the differentiation operator to work on a new set of points. It is used frequently with customized transformations to improve accuracy for certain problems; see, for example, [6].

The methods mentioned above are by no means the only numerical approaches to solving singularly perturbed two-point boundary value problems. For example, [9] proposes a method based on interval subdivision and solving an integral formulation of the problem, and [10] reports an efficiently implemented method computing the Chebyshev expansion coefficients.

**2. Resampling.** The resampling method of Driscoll and Hale [7] is explained here briefly for the case of Chebyshev–Gauss–Lobatto spectral collocation. Let $N$ be an integer and $X = \{x_0, x_1, \ldots, x_N\}$ the set of Chebyshev nodes

$$x_k = \cos\left(\frac{\pi k}{N}\right), \quad 0 \le k \le N.$$

Let $U$ be a vector of length $N+1$ representing a function $u(x)$ evaluated at the Chebyshev nodes. This vector uniquely defines a polynomial $u_N(x)$ of degree at most $N$ that interpolates $u(x)$ at these nodes.

If the matrix $R$ resamples this vector at points $\{y_j\}_{j=0}^{M}$, then the result is the interpolating polynomial evaluated on these points:

$$RU = \begin{bmatrix} u_N(y_0) \\ \vdots \\ u_N(y_M) \end{bmatrix}.$$

Note that the vector $RU$ uniquely defines a polynomial of degree at most $M$. For $M < N$ this means we are removing information stored in $u_N(x)$. Driscoll and Hale refer to this as downsampling.

The matrix $R$ can be formed with only four lines of code. We refer to Driscoll and Hale for this code and describe briefly how it arises.

The interpolating polynomial $u_N(x)$ can be expressed in the barycentric form [5]

$$u_N(x) = \frac{\sum_{k=0}^{N} \frac{c_k u(x_k)}{x - x_k}}{\sum_{k=0}^{N} \frac{c_k}{x - x_k}}, \tag{2.1}$$

where $x_k$ is the $k$th Chebyshev node and $c_k$ is the $k$th barycentric weight, defined as

$$c_k = \begin{cases} 1/2 & k = 0, \\ (-1)^k & k = 1, \ldots, N-1, \\ (-1)^N/2 & k = N. \end{cases} \tag{2.2}$$

Evaluating $u_N(x)$ at the point $y_j$ can then be expressed as the following vector inner product:

$$u_N(y_j) = \frac{1}{\sum_{k=0}^{N} \frac{c_k}{x - x_k}} \begin{bmatrix} \frac{c_0}{y_j - x_0} & \cdots & \frac{c_N}{y_j - x_N} \end{bmatrix} \begin{bmatrix} u(x_0) \\ \vdots \\ u(x_N) \end{bmatrix}.$$

As the column vector in the above equation is the vector $U$, the row vector represents

the $j$th row of the matrix $R$. Thus, the matrix $R$ can be defined elementwise as

$$R_{jk} = \begin{cases} \frac{\frac{c_k}{y_j - x_k}}{\sum_{i=0}^{N} \frac{c_i}{y_j - x_i}} & y_j \neq x_k \\ 1 & y_j = x_k. \end{cases} \tag{2.3}$$

Driscoll and Hale use Chebyshev points of the first kind as the resampling points [7]:

$$y_j = \cos\left(\frac{(2j+1)\pi}{2(M+1)}\right), \quad 0 \leq j \leq M.$$

These points do not include the endpoints $x = \pm 1$. The examples in this paper focus on second-order equations, which require two boundary conditions. We choose $M = N - 2$ for these problems.

The points $x_k$ and $y_j$ are closest for $j = k = 0$. This difference is equal to $1 - \cos(\pi/2(N-1)) = \mathcal{O}\left(1/N^2\right)$. Higham [8] suggests the round-off error in each $R_{jk}$ is $\mathcal{O}(N)$.

**2.1. Application.** Consider using Chebyshev collocation to solve the second-order differential equation

$$\epsilon u''(x) + p(x)u'(x) + q(x)u(x) = f(x), \qquad x \in [-1, +1].$$

This reduces the problem to solving

$$\epsilon u_N''(x) + p(x)u_N'(x) + q(x)u_N(x) = f(x), \quad x \in X.$$

This ultimately leads to $N + 1$ coupled linear equations.

In addition to the equation, we require two boundary conditions for the problem to be well-posed. Standard methodology recommends removing two of the aforementioned linear equations and replacing them with these boundary conditions. Alternatively, we can use resampling in one of two approaches.

First, rather than evaluating the equation at the Chebyshev nodes, we can evaluate it directly at the resampling points $\{y_j\}_{j=0}^{M}$:

$$\epsilon u_N''(y_j) + p(y_j)u_N'(y_j) + q(y_j)u_N(y_j) = f(y_j),$$

which gives $M + 1$ coupled linear equations. Taking $M = N - 2$ will allow a straightforward concatenation of two boundary conditions. The process to find the values of $u_N(y_j)$ has already been described. For $u_N'(y_j)$ and $u_N''(y_j)$ it is a simple matter to resample $DU$ and $D^2U$:

$$\begin{bmatrix} u_N'(y_0) \\ \vdots \\ u_N'(y_M) \end{bmatrix} = RDU, \quad \begin{bmatrix} u_N''(y_0) \\ \vdots \\ u_N''(y_M) \end{bmatrix} = RD^2U.$$

Second, we can resample the linear equations after they have been calculated on the Chebyshev nodes

$$R\left(\epsilon D^2 + PD + Q\right)U = RF,$$

where $P$ and $Q$ are diagonal matrices with entries equal to $p(x_k)$ and $q(x_k)$, respectively, and $F$ is a column vector with entries $f(x_k)$.

Note that the first term, $\epsilon RD^2U$, is the same in both approaches. The remaining terms are different. On the right-hand side, the vector $RF$ is a resampling of the polynomial $f_N(x)$ that interpolates the function $f(x)$ on the Chebyshev nodes. Likewise, $RQU$ is a resampling of the polynomial $(qu)_N(x)$, which interpolates the product $q(x)u(x)$ on the Chebyshev nodes. The same is true of $RPDU$, $(pu')_N(x)$, and $p(x)u'_N(x)$.

Since the second approach resamples the same equations used in the standard method, there is likely to be the most agreement between these two. Which of the two approaches is more accurate is an open question and may vary from problem to problem. Driscoll and Hale recommend the second approach.

**3. Iterated sine-transformation.** The two-point boundary value problem described in subsection 2.1 has at least one boundary layer for small values of $\epsilon$. As such, we would like a number of collocation points in the boundary layer. To do so, one can use a coordinate stretching transformation, such as the iterated sine-transformation introduced in [11].

If we look at the problem on the transformed coordinate $y(x)$ (a function of the old coordinate $x$), then the equation becomes

$$\epsilon v''(y) + \left(\frac{p(x)}{y'(x)} + \epsilon\frac{y''(x)}{y'(x)^2}\right)v'(y) + \frac{q(x)}{y'(x)^2}v(y) = \frac{f(x)}{y'(x)^2},$$

where $v(y) = v(y(x)) = u(x)$. To apply a Chebyshev collocation method to this problem, restrict $y$ to the Chebyshev nodes. Thus, the equation can be discretized as

$$(\epsilon D^2 + PD + Q)V = F,$$

where $D$ is the standard Chebyshev differentiation matrix and $P$, $Q$, and $F$ are discretizations of $\frac{p(x)}{y'(x)} + \epsilon\frac{y''(x)}{y'(x)^2}$, $\frac{q(x)}{y'(x)^2}$, and $\frac{f(x)}{y'(x)^2}$, respectively.

Note that these functions are evaluated on the coordinate $x$. It is then necessary to calculate the values $x$ must take to match $y$. That is, we need to know $x_k$ such that $y_k = y(x_k)$ is the $k$th Chebyshev node.

The iterated sine-transformation is defined as [11]

$$g_0(y) = y, \quad g_m(y) = \sin\left(\frac{\pi}{2}g_{m-1}(y)\right),$$

where $m$ is the number of times the transformation is applied iteratively.

The mapping is bijective for all $m \geq 0$ [11]. Moreover, if $y_k$ is the $k$th Chebyshev node, then

$$g_m(y_0) - g_m(y_1) = g_m(y_{N-1}) - g_m(y_N) = \frac{8}{\pi^2}\left(\frac{\pi^2}{4N}\right)^{2^{m+1}}(1 + \mathcal{O}\left(1/N^2\right)), \tag{3.1}$$

and so the distance between points near the boundary shrinks exponentially with each iteration.

The coordinate $x$ can then be represented as $x(y) = g_m(y)$. The point $x_k$ at which to evaluate $p(x)$, $q(x)$, and $f(x)$ is the $m$th transformation of $y_k$, the $k$th Chebyshev node.

It remains to calculate $1/y'(x)$ and $y''(x)/y'(x)^2$. In [11] one can find the following recursive formulae for calculating these quantities:

$$\begin{aligned}
&\frac{1}{y'(x)} = \prod_{n=0}^{m-1} \frac{\pi}{2} \cos\left(\frac{\pi}{2} g_n(y)\right), \\
&h_0(x) = x, \quad h_m(x) = \frac{2}{\pi} \arcsin(h_{m-1}(x)), \\
&\frac{y''(x)}{y'(x)^2} = \frac{h_m''(x)}{h_m'(x)^2} = \frac{\pi}{2} \tan\left(\frac{\pi}{2} h_m(x)\right) + \frac{\pi}{2} \cos\left(\frac{\pi}{2} h_m(x)\right) \frac{h_{m-1}''(x)}{h_{m-1}'(x)^2}.
\end{aligned} \tag{3.2}$$

Note that $h_m(x)$ is the inverse transformation of $g_m(y)$, and so $y(x) = h_m(x)$. In addition, $h_{m-n}(x) = g_n(y)$ and $h_0''(x) = 0$.

**3.1. Combining transformation and resampling.** Consider that

$$v'(y) = \frac{d}{dy} u(g_m(y)) = u'(x) g_m'(y).$$

The derivative of $g_m(y)$ is (see [11])

$$g_0'(y) = 1, \quad g_m'(y) = \frac{\pi}{2} \cos\left(\frac{\pi}{2} g_{m-1}(y)\right) g_{m-1}'(y).$$

For $m = 1$, it is straightforward to see that $g_1'(\pm 1) = 0$. It follows that $v'(\pm 1) = 0$ for $m > 0$.

This imposes artificial Neumann boundary conditions on the problem. They are seen in the term $y''(x)/y'(x)^2$, where infinities are calculated at $x = \pm 1$. This also causes $1/y'(\pm 1) = 0$, and so the other terms vanish.

Recall that standard methodology recommends removing rows to make room for boundary conditions. By removing the first and last rows, the underlying Neumann conditions are not seen. By contrast, when incorporating resampling, these conditions become important, and the infinities calculated in $y''(x)/y'(x)^2$ cause the resampling to fail.

**3.2. Regularized sine-transformation.** This can be fixed by using a regularized form of this transformation so that $g_m'(\pm 1)$ is nonzero. The regularized sine-transformation is defined as

$$g_m(y) = \mu g_{m-1}(y) + (1 - \mu) \sin\left(\frac{\pi}{2} g_{m-1}(y)\right), \quad 0 < \mu < 1, \tag{3.3}$$

which leads to the following changes to the iterated sine-transformation:

$$\begin{aligned}
\frac{1}{y'(x)} &= g_m'(y) \\
&= \left[\mu + (1 - \mu) \frac{\pi}{2} \cos\left(\frac{\pi}{2} g_{m-1}(y)\right)\right] g_{m-1}'(y), \\
\frac{y''(x)}{(y'(x))^2} &= -\frac{g_m''(y)}{g_m'(y)} \\
&= -\frac{g_{m-1}''(y)}{g_{m-1}'(y)} + g_{m-1}'(y) \left(\frac{\pi}{2}\right)^2 \frac{(1 - \mu) \sin\left(\frac{\pi}{2} g_{m-1}(y)\right)}{\mu + (1 - \mu) \frac{\pi}{2} \cos\left(\frac{\pi}{2} g_{m-1}(y)\right)}.
\end{aligned} \tag{3.4}$$

These terms can be computed recursively.

LEMMA 3.1. *The transformation defined in* (3.3) *is bijective. The spacing between points near the boundary is*

$$g_m(y_0) - g_m(y_1) = \frac{\mu^m}{2}\frac{\pi^2}{N^2} + \mathcal{O}\left(1/N^4\right).$$

*In addition,* $1/y'(\pm 1) = \mu^m$ *and* $y''(\pm 1)/y'(\pm 1)^2 = \pm\frac{1-\mu^m}{\mu}$.

The proof of this lemma is presented in Appendix A.

The distance between points near the boundary still shrinks with each iteration under the regularized sine-transformation; the rate, however, is significantly reduced from $\mathcal{O}(1/N^{2^{m+1}})$ to $\mathcal{O}\left(\mu^m/N^2\right)$. As $\mu$ approaches zero, we retrieve the infinities calculated for the iterated sine-transformation. As $m$ approaches infinity, $y''(\pm 1)/y'(\pm 1)^2$ approaches $\pm 1/\mu$.

It should be mentioned that in the limit as $\mu$ approaches zero, the regularized sine-transformation becomes the original. Therefore, while the spacing appears to depend primarily on $\mu$, it cannot decrease below the spacing defined by the original transformation.

**4. RSC.** Rather than transforming the equation by stretching the coordinates, we can construct a collocation method for the new transformed points. One way to do so is through RSC [12]. This method is used in a number of applications. For example, Chen, Wang, and Wu use it with a sinh-transformation for a system of singularly perturbed convection-diffusion equations [6].

The RSC method changes the differentiation operator to act on the transformed set of points. Given a set of points $X = \{x_k\}$, the new differentiation operators are defined as

$$D_{X,jk} = \begin{cases} \frac{c_k}{c_j(x_j - x_k)}, & j \neq k, \\ -\sum_{i\neq j} D_{X,ji}, & j = k, \end{cases}$$
$$D^{(2)}_{X,jk} = \begin{cases} 2D_{X,jk}\left(D_{X,jj} - \frac{1}{x_j - x_k}\right), & j \neq k, \\ -\sum_{i\neq j} D^{(2)}_{X,ji}, & j = k, \end{cases} \tag{4.1}$$

where $D_X$ is the first-order operator, $D^{(2)}_X$ is the second-order operator, and $c_j$ are the weights previously defined in (2.2). Note that if $X$ is the set of Chebyshev points, this returns the original Chebyshev differentiation operators.

This method arises, like resampling, from the barycentric form of the interpolating polynomial. By changing the interpolation points without adjusting the weights, the interpolant is no longer a polynomial; instead, it is a rational function. This rational interpolant has been introduced in [4] and explored in depth in [2].

The new discrete system to solve is

$$(\epsilon D^{(2)}_X + P_X D_X + Q_X)U = F_X,$$

where $P_X$ and $Q_X$ are diagonal matrices with entries $p(x_k)$ and $q(x_k)$, respectively, and $F_X$ is a column vector with entries $f(x_k)$. The distance between points is defined by the transformation and can be found in the previous sections.

Using RSC removes the need to compute $1/y'(x)$ and $y''(x)/y'(x)^2$. However, points must be separated by at least machine epsilon to prevent singularities when calculating $D_X$ and $D^{(2)}_X$.

**5. Experiments.** We apply the various methods described to the ODE from [11],

$$\epsilon u''(x) - xu'(x) - u(x) = \left(\frac{x+1}{\epsilon} - 1\right) e^{-\frac{x+1}{\epsilon}} - 2\left(\frac{x-1}{\epsilon} + 1\right) e^{\frac{x-1}{\epsilon}}, \tag{5.1}$$

with exact solution $u(x) = e^{-(x+1)/\epsilon} + 2e^{(x-1)/\epsilon}$ and Dirichlet boundary conditions. For sufficiently small $\epsilon$ the boundary conditions are numerically equivalent to $u(1) = 2$ and $u(-1) = 1$ in IEEE double precision. We will focus on $\epsilon = 10^{-9}$.

Consider the exact solution evaluated at $x + \delta$ for some small $\delta$:

$$u(x+\delta) = e^{-(x+1)/\epsilon} e^{-\delta/\epsilon} + 2e^{(x-1)/\epsilon} e^{\delta/\epsilon}.$$

In particular, let $x$ be near 1 and $\delta = 10^{-16}$, on the order of machine epsilon. Then the difference between $u(x+\delta)$ and $u(x)$ is on the order of $\delta/\epsilon = 10^{-7}$. Therefore, we cannot measure the error below this level. More precisely, the error of the numerical solution near the boundary cannot drop below the level of $\delta/\epsilon$, simply because of the conditioning of evaluating the solution. Away from the boundary the error decreases below that threshold; see Figure 6.

The transformations described in the previous section are designed to place more points within the boundary layers of the ODE. These layers have a width related to the magnitude of $\epsilon$. Therefore, we wish for the distance between points near the boundary to be of the same magnitude as $\epsilon$. We use this distance as the independent variable in the following experiments and define it as

$$\eta = \frac{8}{\pi^2}\left(\frac{\pi^2}{4N}\right)^{2^{m+1}} \tag{5.2}$$

for the iterated sine-transformation (see (3.1)) and

$$\eta = \frac{\mu^m}{2}\frac{\pi^2}{N^2} \tag{5.3}$$

for the new regularized sine-transformation (see Lemma 3.1). We would then like $\eta = \mathcal{O}(\epsilon)$.

Consider $\epsilon = 10^{-9}$ and a variety of values of $N$. The following table gives the number of iterations of the iterated sine-transformation that gives the desired magnitude. Note this describes only the minimum number of iterations to guarantee that at least one point lies in the boundary layer.

| $N$ | $m$ | Magnitude |
|---|---|---|
| 128 | 2 | $10^{-14}$ |
| 256 | 2 | $10^{-17}$ |
| 512 | 1 | $10^{-10}$ |
| 1024 | 1 | $10^{-11}$ |

Let us now consider the regularized sine-transformation. The following table gives the value of $\mu$ that provides the desired magnitude for various $N$ and $m$. This indicates a maximum value that $\mu$ is allowed to take. Smaller $\mu$ will ensure the boundary layer is better resolved.

| $N$ | $m$ | $\mu$ | $m$ | $\mu$ | $m$ | $\mu$ |
|---|---|---|---|---|---|---|
| 128 | 1 | $3.32 \times 10^{-6}$ | 2 | $1.82 \times 10^{-3}$ | 3 | $1.49 \times 10^{-2}$ |
| 256 | 1 | $1.33 \times 10^{-5}$ | 2 | $3.64 \times 10^{-3}$ | 3 | $2.37 \times 10^{-2}$ |
| 512 | 1 | $5.31 \times 10^{-5}$ | 2 | $7.29 \times 10^{-3}$ | 3 | $3.76 \times 10^{-2}$ |
| 1024 | 1 | $2.12 \times 10^{-4}$ | 2 | $1.46 \times 10^{-2}$ | 3 | $5.97 \times 10^{-2}$ |

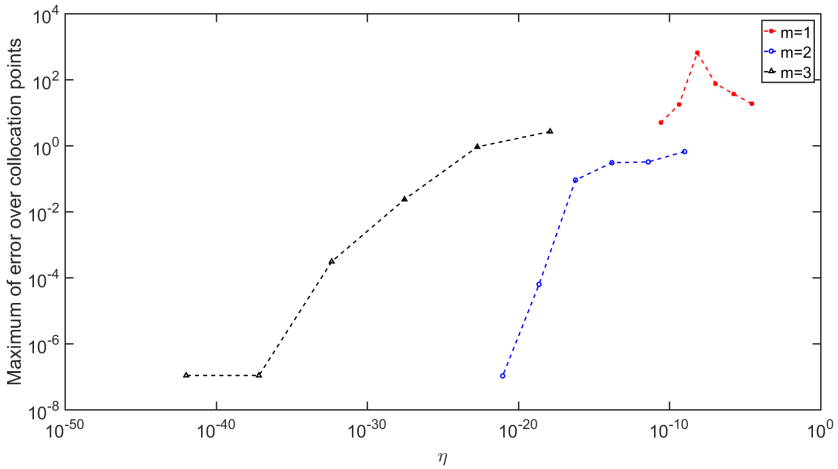


FIG. 1. *Approximation to the $L^\infty$ error of the solution as a function of the spacing between points near the boundary. $N$ is equal to $32, 64, 128, 256, 512$, and $1024$, increasing from right to left along each line.*

In sections 5.1–5.3 the methods are first tested using row replacement of the first and last rows to include the boundary conditions. Once ideal parameter choices are established, the regularized sine-transformation is tested with resampling in section 5.4.

**5.1. Iterated sine-transformation.** In the introduction to this section, we discussed at length the effect of the transformation on the spacing between points at the boundary. This is, however, not always the leading restriction on accuracy. Figure 1 shows the accuracy of solving (5.1) using the iterated sine-transformation for $m = 1$, 2, and 3 for various values of $N$. The error is plotted as a function of the approximate spacing at the boundary, $\eta$.

The spacing for this transformation, $\eta$, depends on the parameters $N$ and $m$. Excepting those values of the parameters that do not admit points within the boundary layer, increasing $N$ improves the accuracy. Neither increasing $m$ nor decreasing $\eta$ will generally give better accuracy. Thus, $N$ dictates the accuracy more than the spacing $\eta$ and the number of iterations $m$.

**5.2. Regularized sine-transformation.** Figure 2 shows the same results but for the regularized sine-transformation. This transformation is highly dependent on the regularization parameter $\mu$. We fix $N = 128$ and $m = 3$. Boundary conditions are included through the replacement of the first and last rows.

Accuracy does not improve until $\eta$ is of order $\mathcal{O}(\epsilon)$. Also note that error falls below $\mathcal{O}(1)$ only once there is at least one point within the boundary layer. The number of local minima corresponds to the number of points allowed in the boundary layer by the transform. There is, however, no pattern to when these minima occur in terms of spacing or distribution of points.

If we change $N$, we expect similar dependence of accuracy on the spacing. Figure 3 shows that the plots of error for $N = 128$, 256, and 512 have many similar features. There are about twice as many local minima for $N = 256$, likely the result of more points available to move into the boundary layer. The global minima of error occurs at roughly the same spacing, approximately $10^{-11}$. At this spacing there are 6, 9, and 12 points within $\epsilon$ of $x = 1$ for $N = 128$, 256, and 512 respectively.

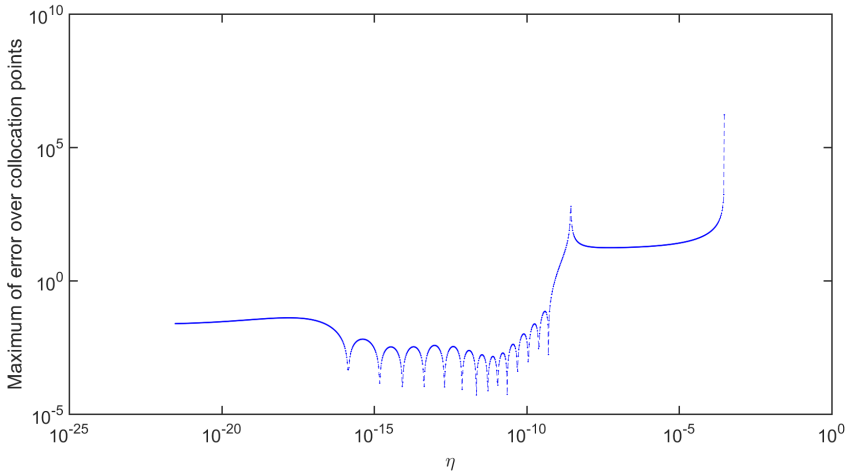


FIG. 2. *Error of the regularized sine-transformation as a function of the spacing between points near the boundary for* $N = 128$ *and* $m = 3$.

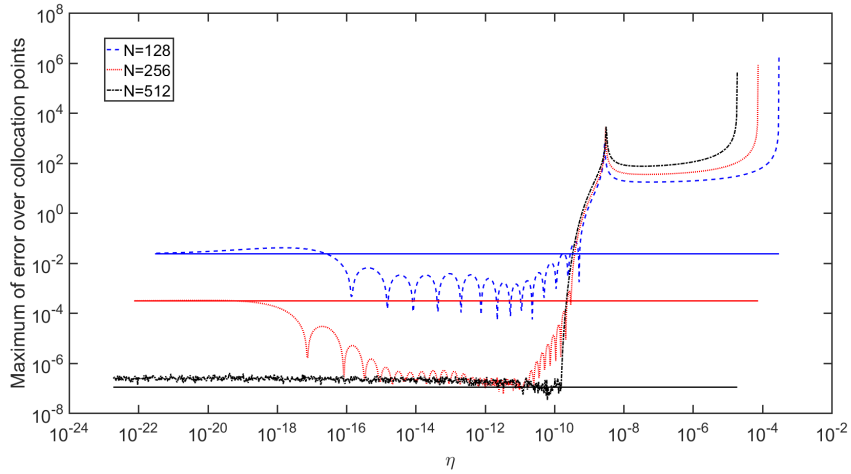


FIG. 3. *Error of the regularized sine-transformation as a function of the spacing between points near the boundary for* $N = 128, 256,$ *and* $512$ *and* $m = 3$. *The horizontal lines represent the error of the iterated sine-transformation for the same* $N$ *and* $m$.

The error of the iterated sine-transformation is provided for reference as horizontal lines for the same $N$ and $m$. The spacing they occur at does not appear on the $x$–axis. Since the regularized sine-transformation converges to the iterated sine-transformation as $\mu$ converges to zero, the errors likewise converge as the spacing decreases. For appropriate choice of $\mu$ the regularized sine-transformation outperforms the iterated sine-transformation with the same $m$ and $N$. Based on the shapes of the data for the previous two figures, we can conjecture that the spacing should be between $\mathcal{O}(\epsilon)$ and $\mathcal{O}(\epsilon^2)$.

Figure 3 is replicated for $m = 2$ in Figure 4. The iterated sine-transformation does not achieve the best accuracy, unlike for $m = 3$, where both transformations had error on the order of $10^{-7}$ for $N = 512$. It should be even more evident that the regularized

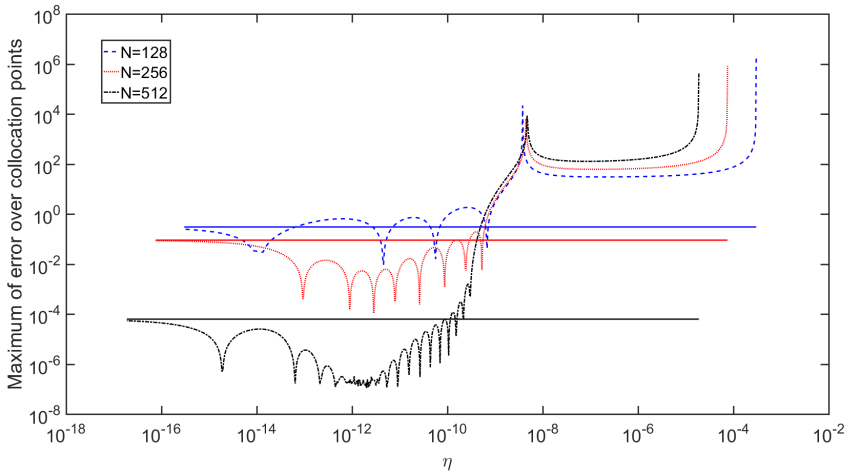


FIG. 4. *Error of the regularized sine-transformation as a function of the spacing between points near the boundary for* $N = 128, 256,$ *and* $512$ *and* $m = 2$. *The horizontal lines represent the error of the iterated sine-transformation for the same* $N$ *and* $m$.

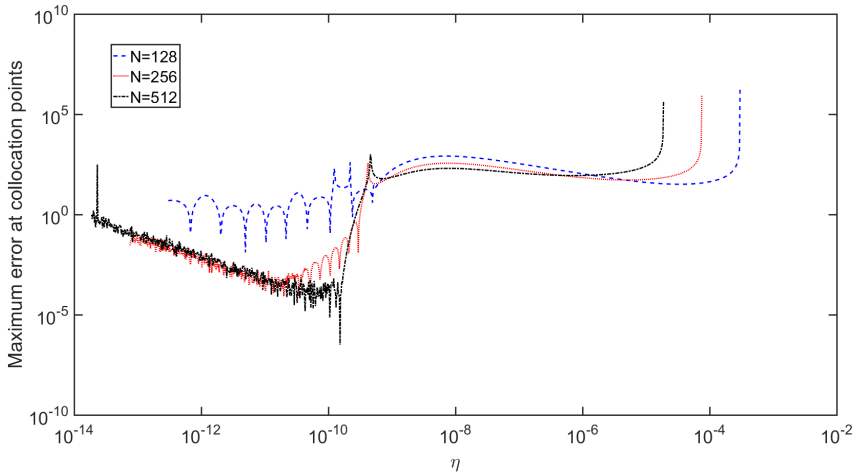


FIG. 5. *Error of the RSC method with the regularized sine-transformation as a function of the spacing* ($\eta$) *between points near the boundary for* $N = 128, 256,$ *and* $512$ *and* $m = 3$. *Spacing is changed by altering* $\mu$ *in the regularized sine-transformation for two boundary layers.*

sine-transformation offers significant benefits over the iterated sine-transformation, with the regularized sine-transformation possessing error several orders of magnitude lower than the iterated sine-transformation for all choices of $N$.

**5.3. RSC.** We now apply the rational spectral collocation method with the regularized sine-transformation. The iterated sine-transformation has points too close together to be used by RSC. A value of $\mu$ is needed, and Figure 5 examines which value is best. The relation between $\eta$ and $\mu$ is explained in the introduction to this section.

Figure 5 shows that, again, spacing needs to be below the value of $\epsilon$ for any accuracy to be achieved. Also shown is an increase in error as spacing grows smaller. This

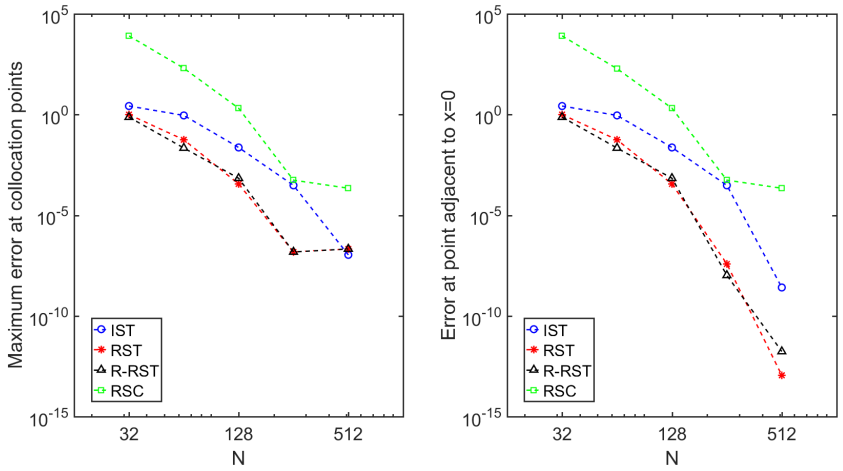


FIG. 6. *Comparison of error for the iterated sine-transform (IST), the regularized sine-transform (RST), the regularized sine-transform with resampling (R-RST), and the regularized sine-transform with RSC (RSC) as functions of $N$. The spacing between points near the boundary is $10^{-10}$ for RSC and $10^{-11}$ for the remaining methods, and $m = 3$. The left figure shows error over all points, while the right figure shows error for points near zero.*

is likely due to round-off error in calculating the differentiation matrices associated with RSC. Fits to these increases show that this source of error grows as $\mathcal{O}(1/\eta)$. Given this, we should optimize this method by choosing the largest $\eta$ that allows an accurate result, which appears to be roughly $\eta = 10^{-10}$.

**5.4. Resampling.** Having thoroughly tested the behavior of the regularized sine-transformation, we are now prepared to combine it with Driscoll's and Hale's resampling. The given problem appears to have an ideal spacing of $10^{-11}$. We choose $m = 3$ and an appropriate value of $\mu$. Additionally, we compare the iterated sine-transformation and RSC with the regularized sine-transformation, using spacing $10^{-11}$ and $10^{-10}$, respectively. The independent variable is now $N$, the number of collocation points: This parameter directly influences computation time.

Figure 6 provides a comparison of accuracy of the four methods. The left figure shows error over all points. The regularized sine-transformation with resampling performs best for smaller $N$. The regularized sine-transformation with and without resampling have the same accuracy for sufficiently large $N$. The iterated sine-transformation shows spectral convergence but of slower order than the regularized sine-transformation. RSC suffers round-off error on the order of $10^{-3}$ and cannot converge below this.

Recall that the minimum error we achieve ($10^{-7}$) is related to round-off error in calculating the values of the exact solution near the boundary. Away from $x = \pm 1$ this problem does not occur, and the error continues to decrease systematically. To this end, the right of Figure 6 shows the error at a single point adjacent to $x = 0$ for all four methods. In [11] it is pointed out that this is the location of the largest error for $N < 128$. We see that the error at this point continues to decrease spectrally, and the regularized sine-transformation provides the best accuracy in this measure.

**5.5. Other examples.** Changing the value of $\epsilon$ affects the width of the boundary layer. This results in new optimal spacing of points near the boundary and a new

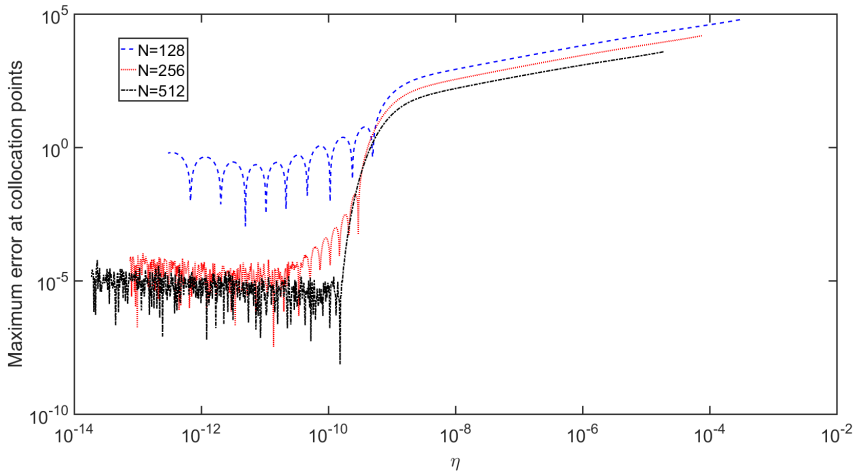


FIG. 7. *Error of the RSC method for the second example with the regularized sine-transformation as a function of the spacing ($\eta$) between points near the boundary for $N = 128, 256,$ and 512 and $m = 3$. Spacing is changed by altering $\mu$ in the regularized sine-transformation for two boundary layers.*

minimum error, but all other behavior mentioned above remains the same. The optimal spacing ranges are presented for $\epsilon = 10^{-9}$, $10^{-6}$, and $10^{-3}$ in the following table.

| $\epsilon$ | Spacing range | Minimum error |
|---|---|---|
| $10^{-9}$ | $10^{-10}$ – $10^{-12}$ | $10^{-7}$ |
| $10^{-6}$ | $10^{-7}$ – $10^{-9}$ | $10^{-10}$ |
| $10^{-3}$ | $10^{-4}$ – $10^{-6}$ | $10^{-13}$ |

We also consider another example, this one taken from [3],

$$\epsilon u''(x) - u'(x) = 1/2, \quad u(\pm 1) = 0,$$

which has exact solution

$$u(x) = -\frac{x+1}{2} + \frac{e^{(x-1)/\epsilon} - e^{-2/\epsilon}}{1 - e^{-2/\epsilon}}.$$

For a fair comparison we again use $\epsilon = 10^{-9}$.

In almost all respects the results for this example are identical to those from before, except for the RSC method, where the round-off error increases at a significantly lower rate, as can be seen in Figure 7. Fits to these increases give a growth rate of $\mathcal{O}\left(\eta^{-0.17}\right)$, approximately. The RSC method, however, is still unable to achieve the minimum error for this problem.

**6. Conclusion.** Resampling eliminates the ambiguity of row removal in spectral collocation problems. When coupled with the iterated sine-transformation, resampling reveals artificial Neumann boundary conditions imposed by the transformation. The transformation is regularized to prevent such artificial conditions.

The regularized sine-transformation has greater control over the spacing between points near the boundary. This allows it to achieve orders of magnitude improvement in accuracy over the iterated sine-transformation (see, for example, Figures 3 and 4).

Combined with resampling the regularized sine-transformation achieves the best possible accuracy. Resampling only offers improvement for low numbers of collocation points. In Figure 6 the regularized sine-transformation performs equally well with and without resampling for $N \geq 128$.

The accuracy of all methods used is predicated on knowing the ideal spacing between points near the boundary. This appears to be closely tied to the value of the perturbation parameter $\epsilon$. It can be estimated from the above data that the ideal spacing is within an order of magnitude of $\epsilon \times 10^{-2}$.

## Appendix A. Proof of Lemma (3.1).

*Proof.* We show that the transformation is bijective. It is clear that $g_m(\pm 1) = \pm 1$ for all $m$. Therefore, it suffices to show $g'_m(y) > 0$ for $y \in (-1, 1)$.

We proceed by induction. It is trivially true for $m = 0$. We suppose $g'_k(y) > 0$ for all $y \in (-1, 1)$. Since $g_k(\pm 1) = \pm 1$, we have that $|g_k(y)| \leq 1$ for all $y \in [-1, 1]$. Therefore, $\cos(\pi g_k(y)/2) \geq 0$ for all $y \in (-1, 1)$. The proof then follows from (3.4):

$$g'_{k+1}(y) = \left[\mu + (1-\mu)\frac{\pi}{2}\cos\left(\frac{\pi}{2}g_k(y)\right)\right] g'_k(y) \geq \mu g'_k(y) > 0.$$

Thus, the transformation is bijective.

Consider the Taylor expansion of $g_m(y)$ around $y = 1$:

$$\begin{aligned} g_m(y) &= 1 + g'_m(1)(y-1) + \mathcal{O}\left((y-1)^2\right) \\ &= 1 + \mu^m(y-1) + \mathcal{O}\left((y-1)^2\right). \end{aligned}$$

If we then consider the difference between $g_m(y_0)$ and $g_m(y_1)$, we have that

$$\begin{aligned} g_m(y_0) - g_m(y_1) &= -\mu^m(\cos(\pi/N) - 1) + \mathcal{O}\left((1 - \cos(\pi/N))^2\right) \\ &= \frac{\mu^m}{2}\frac{\pi^2}{N^2} + \mathcal{O}\left(1/N^4\right). \end{aligned}$$

It is trivial to show that $1/y'(\pm 1) = \mu^m$. Finally,

$$\begin{aligned} \frac{y''(\pm 1)}{(y'(\pm))^2} &= -\frac{g''_m(\pm 1)}{g'_m(\pm 1)} = -\frac{g''_{m-1}(\pm 1)}{g'_{m-1}(\pm 1)} \pm \mu^{m-1}\frac{1-\mu}{\mu} \\ &= \pm\frac{1-\mu}{\mu}\sum_{n=0}^{m-1}\mu^n = \pm\frac{1-\mu^m}{\mu}. \end{aligned}$$

∎

## REFERENCES


[1] R. Baltensperger and J.-P. Berrut, *The linear rational collocation method*, J. Comput. Appl. Math., 134 (2001), pp. 243–258, https://doi.org/10.1016/S0377-0427(00)00552-5.

[2] R. Baltensperger, J.-P. Berrut, and B. Noël, *Exponential convergence of a linear rational interpolant between transformed Chebyshev points*, Math. Comp., 68 (1999), pp. 1109–1120.

[3] R. Baltensperger and M. R. Trummer, *Spectral differencing with a twist*, SIAM J. Sci. Comput., 24 (2003), pp. 1465–1487, https://doi.org/10.1137/S1064827501388182.

[4] J.-P. Berrut, *Rational functions for guaranteed and experimentally well-conditioned global interpolation experimentally well-conditioned global interpolation*, Comput. Math. Appl., 15 (1988), pp. 1–16.

[5] J.-P. Berrut and L. N. Trefethen, *Barycentric Lagrange interpolation*, SIAM Rev., 46 (2004), pp. 501–517, https://doi.org/10.1137/S0036144502417715.

[6] S. Chen, Y. Wang, and X. Wu, *Rational spectral collocation method for a coupled system of singularly perturbed boundary value problems*, J. Comput. Math., 29 (2011), pp. 458–473.

[7] T. A. Driscoll and N. Hale, *Rectangular spectral collocation*, IMA J. Numer. Anal., 36 (2016), pp. 108–132, https://doi.org/10.1093/imanum/dru062.

[8] N. J. Higham, *The numerical stability of barycentric Lagrange interpolation*, IMA J. Numer. Anal., 24 (2004), pp. 547–556, https://doi.org/10.1093/imanum/24.4.547.

[9] J.-Y. Lee and L. Greengard, *A fast adaptive numerical method for stiff two-point boundary value problems*, SIAM J. Sci. Comput., 18 (1997), pp. 403–429, https://doi.org/10.1137/S1064827594272797.

[10] S. Olver and A. Townsend, *A fast and well-conditioned spectral method*, SIAM Rev., 55 (2013), pp. 462–489, https://doi.org/10.1137/120865458.

[11] T. Tang and M. R. Trummer, *Boundary layer resolving pseudospectral methods for singular perturbation problems*, SIAM J. Sci. Comput., 17 (1996), pp. 430–438, https://doi.org/10.1137/S1064827592234120.

[12] T. W. Tee and L. N. Trefethen, *A rational spectral collocation method with adaptively transformed Chebyshev grid points*, SIAM J. Sci. Comput., 28 (2006), pp. 1798–1811, https://doi.org/10.1137/050641296.