CANADIAN APPLIED
MATHEMATICS QUARTERLY
Volume 20, Number 3, Fall 2012

ASYMPTOTICALLY CORRECT INTERPOLATION-BASED SPATIAL ERROR ESTIMATION FOR 1D PDE SOLVERS

TOM ARSENAULT, TRISTAN SMITH, PAUL MUIR
AND JACK PEW

ABSTRACT. BACOL and BACOLR are B-spline Gaussian collocation method-of-lines packages for the numerical solution of systems of one-dimensional parabolic partial differential equations (PDEs). In previous studies, they were shown to be superior to other similar packages, especially for problems exhibiting sharp spatial layer regions where a stringent tolerance is imposed. A significant feature of these solvers is that, in addition to the temporal error control provided by the underlying time-integrator, they adapt the spatial mesh to control a high order estimate of the spatial error. In addition to computing a primary collocation solution of a given spatial order, the BACOL/BACOLR codes also compute, at a substantial cost, a *secondary* collocation solution of one higher order, and then the difference between the two collocation solutions is used to give an estimate of the leading order term in the error for the lower order solution. In this paper we consider an approach in which the computation of lower order collocation solution is replaced by an inexpensive interpolant (based on evaluations of the higher order collocation solution) constructed so that the leading order term in the interpolation error agrees (asymptotically) with the leading order term in original BACOL/BACOLR error estimate. We provide numerical results to show that this interpolation-based error estimate can provide spatial error estimates of comparable accuracy to those currently computed by BACOL, but at a significantly lower cost.

This work was supported by the Mathematics of Information Technology and Complex Systems Network and by the Natural Sciences and Engineering Research Council of Canada.

AMS subject classification: 65M15, 65M20, 65M70.

Keywords: 1D Parabolic PDEs, collocation, spatial error estimation, interpolation, method-of-lines.

1 Introduction Systems of one-dimensional (1D) parabolic partial differential equations (PDEs) arise in a variety of applications - see, e.g., the recent book [**15**] and references within. Large systems of 1D parabolic PDEs also arise from the application of an approach we call the method-of-surfaces in which a 2D parabolic PDE is discretized in one of its two spatial dimensions [**17**].

In this paper, we will assume a system of PDEs with *NPDE* components having the form

$$\underline{u}_t(x,t) = \underline{f}\left(t, x, \underline{u}(x,t), \underline{u}_x(x,t), \underline{u}_{xx}(x,t)\right), \quad a \le x \le b, \ t \ge t_0, \tag{1}$$

with initial conditions

$$\underline{u}(x,t_0) = \underline{u}_0(x), \quad a \le x \le b, \tag{2}$$

and separated boundary conditions

$$\underline{b}_L\left(t, \underline{u}(a,t), \underline{u}_x(a,t)\right) = \underline{0}, \qquad \underline{b}_R\left(t, \underline{u}(b,t), \underline{u}_x(b,t)\right) = \underline{0}, \quad t \ge t_0. \tag{3}$$

The method-of-lines (MOL) is a popular framework upon which a number of software packages for the numerical solution of 1D parabolic PDEs have been developed. The approach typically involves partitioning the spatial domain with a set of mesh points and applying a standard spatial discretization scheme, e.g., finite differences, to discretize the PDEs, leading to a system of time-dependent ODEs. In earlier codes, the boundary conditions were differentiated and together with the ODEs from the discretization of the PDEs, these were solved by an ODE solver. In more recent codes, the boundary conditions are treated directly; the combined system of differential-algebraic equations (DAEs) is then solved using a high quality DAE solver such as DASSL [**6**] or RADAU5 [**13**]. Over the last few decades, a number of high quality MOL solvers for this problem class have been developed, including PDECOL [**18**]/EPDCOL [**16**], D03PPF [**8**], TOMS731 [**5**], MOVCOL [**14**], HPNEW [**19**], BACOL [**23, 25**], and BACOLR [**26**]. In all of the above codes control of an estimate of the temporal error is provided by the underlying initial value ODE/DAE solver. The most recently developed MOL codes also provide control of a high order estimate of the spatial error. That is, they compute a high order estimate of the spatial error and then based on this estimate adapt the spatial discretization, by changing the mesh and/or the order of accuracy of the spatial discretization scheme, in an attempt to compute a solution whose spatial error estimate is less than the user provided tolerance. HPNEW, BACOL, and BACOLR are examples of codes of this type.

In a recent study [**24**], BACOL was shown to be comparable to and in some cases superior to other similar packages, especially for problems exhibiting sharp spatial layer regions and for problems where a sharp tolerance is imposed. Furthermore, the paper [**26**] shows that BACOLR is generally comparable in performance to BACOL and that for certain problems, such as the cubic Schrödinger equation, BACOLR has substantially superior performance to BACOL.

Both BACOL and BACOLR use B-spline Gaussian collocation for the spatial discretization. The approximate solution is represented as a linear combination of known spatial basis functions - piecewise polynomials of a given degree p implemented using the B-spline package [**9**] - with unknown, time-dependent coefficients. Conditions on these coefficients are obtained by applying collocation conditions, i.e., by requiring the approximate solution to satisfy the PDEs at the images of the Gauss points on each subinterval of a mesh which partitions the problem domain. The resultant system of time dependent ODEs together with the boundary conditions gives a system of DAEs which is then solved, in BACOL, using a modified version of DASSL, or, in BACOLR, using a modified version of RADAU5, to obtain the time dependent coefficients.

In both BACOL and BACOLR, an estimate of the spatial error of the collocation solution is obtained by computing a second global collocation solution using B-splines of degree $p+1$ and the difference between the two collocation solutions gives an estimate of the leading order term in the spatial error for the lower order collocation solution. The spatial error estimate is used to determine the acceptability of the solution; that is, the estimate must be less than the user provided tolerance in an appropriately scaled norm. If the tolerance is not satisfied, the spatial error estimates on each mesh subinterval are used to guide a mesh refinement process based on the principle of equidistribution—the mesh points are redistributed (and possibly new mesh points are added to the mesh) in an attempt to (a) equidistribute the spatial error estimate over the subintervals of the new mesh, and to (b) obtain a spatial error estimate that satisfies the user tolerance.

While the spatial error estimate is central to the effective performance of BACOL and BACOLR, it is clear that the computation of two global collocation solutions represents a major computational cost in each of these codes, and thus it would appear to be worthwhile to attempt to replace the error estimation scheme currently employed within BACOL/BACOLR with a lower cost scheme of comparable quality, if possible. There is a substantial body of literature on error estimation for the numerical solution of PDEs, see, e.g., [**1**] and references within. However, the recent work most closely related to the current investigation is by Moore [**19-22**], in which interpolation error based error estimates for 1D parabolic PDEs are discussed.

The basic question we investigate in this paper is how to obtain a high order estimate of the spatial error while computing only one of the two collocation solutions currently computed by BACOL/BACOLR. In this paper, we will focus on BACOL; however, since BACOL and BACOLR use the same spatial discretization and spatial error estimation scheme, the results will also apply to BACOLR, and with appropriate adaptation, may be relevant to other solvers. The general approach is to replace one of the collocation solutions with a low cost interpolant that can then be used in place of that collocation solution in the estimate of the spatial error. Since BACOL computes a higher order and a lower order collocation solution, there are two obvious strategies: (i) compute only the lower order collocation solution and construct an interpolant to replace the higher order collocation solution, and (ii) compute only the higher order collocation solution and construct an interpolant to replace the lower order collocation solution.

In [**2**], the authors explore approach (i); the key observation is that because BACOL employs collocation at Gauss points there are certain points within each subinterval of the spatial mesh where the collocation solution is superconvergent, i.e., the order of accuracy is at least one order higher than at an arbitrary point in the spatial domain. The authors describe the development of a superconvergent interpolant (SCI) based on these superconvergent values and show that the approach can yield spatial error estimates of comparable quality to those currently obtained by BACOL but at a lower cost. The authors also observe, however, that this approach can significantly overestimate the spatial error on subintervals for which the ratio of adjacent subinterval sizes is large.

In the current paper, we consider approach (ii). This will involve computing a lower order interpolant (LOI) that can replace the lower order solution in the computation of spatial error. The LOI is based on evaluations of the higher order collocation at certain points within each subinterval such that the resultant interpolant has an interpolation error whose leading order term agrees (asymptotically) with the leading order term in the collocation error of the lower order collocation solution. The latter is the quantity that is currently employed as the spatial error estimate in BACOL. We provide numerical results to compare the LOI error estimates with the original BACOL error estimates and with the error estimates obtained from the SCI approach. We show that the LOI approach leads to error estimates that are of comparable quality to those computed by the original BACOL code but at a significantly lower cost. We also show that the LOI approach is superior to the SCI approach in certain respects.

This paper is organized as follows. In Section 2, we provide a brief review of the algorithms employed by BACOL. In Section 3 we briefly review the SCI-based spatial error estimation scheme since it is related to the LOI scheme. Section 4 introduces the LOI approach. Numerical results are provided in Section 5 to compare the three error estimation schemes. In Section 6 we briefly discuss the computational costs of the LOI approach and compare these costs with those for the error estimation scheme currently employed in BACOL and with the SCI scheme. We close, in Section 7, with our conclusions and a discussion of future work.

2 BACOL Given a spatial mesh $a = x_0 < x_1 < \cdots < x_{NINT} = b$, the approximate solution is represented in BACOL as a linear combination of known B-spline basis functions [**9**] (piecewise polynomials of degree p where $3 \leq p \leq 11$) with unknown time dependent coefficients. Letting $\{B_{p,i}(x)\}_{i=1}^{NC_p}$ be the B-spline basis functions, the approximate solution, $\underline{U}(x,t)$, then has the form

$$\underline{U}(x,t) = \sum_{i=1}^{NC_p} \underline{y}_{p,i}(t) B_{p,i}(x), \tag{4}$$

where $\underline{y}_{p,i}(t)$ represents the time dependent coefficient of the i-th B-spline basis function, $B_{p,i}(x)$, and where $NC_p = NINT(p-1)+2$ is the dimension of the piecewise polynomial subspace.

The PDE is discretized in space by imposing collocation conditions on the approximate solution at images of the $p-1$ Gauss points (see, e.g., [**4**]) on each subinterval and by requiring the approximate solution to satisfy the boundary conditions. The collocation conditions have the form

$$\frac{d}{dt}\underline{U}(\xi_l,t) = \underline{f}\left(t, \xi_l, \underline{U}(\xi_l,t), \underline{U}_x(\xi_l,t), \underline{U}_{xx}(\xi_l,t)\right), \tag{5}$$

where $l = 2, \ldots, NC_p - 1$, and where the collocation points are defined by

$$\xi_l = x_{i-1} + h_i\rho_j, \quad \text{where } \ l = 1 + (i-1)(p-1) + j, \quad \text{for } \ i = 1, \ldots, NINT, \quad j = 1, \ldots, p-1, \tag{6}$$

where $h_i = x_i - x_{i-1}$ and $\{\rho_i\}_{i=1}^{p-1}$ are the images of the $p-1$ Gauss points on $[0,1]$. The points, $\xi_1 = a$ and $\xi_{NC_p} = b$, are associated with requiring the approximate solution to satisfy the boundary conditions, and this gives the remaining two equations,

$$\underline{b}_L\left(t, \underline{U}(a,t), \underline{U}_x(a,t)\right) = \underline{0}, \qquad \underline{b}_R\left(t, \underline{U}(b,t), \underline{U}_x(b,t)\right) = \underline{0}.$$

The collocation conditions, (5), represent a system of ODEs (in time) for which the unknown solution components are the time dependent coefficients, $\underline{y}_{p,i}(t)$. These ODEs coupled with the boundary conditions give an index-1 system of DAEs, which, as mentioned earlier, is treated using DASSL. After DASSL has computed approximations for the $\underline{y}_{p,i}(t)$ values at a given time t, these can be employed together with the known B-spline basis functions, $B_{p,i}(x)$, within (4), to obtain values of the approximate solution at desired x values.

The collocation solution, $\underline{U}(x,t)$, for the current time is accepted by BACOL only if its spatial error estimate satisfies the user tolerance. This spatial error estimate is obtained by computing a second global collocation solution on the same spatial mesh at the same time t. This approximate solution, $\overline{\underline{U}}(x,t)$, has the form

$$\overline{\underline{U}}(x,t) = \sum_{i=1}^{NC_{p+1}} \underline{y}_{p+1,i}(t) B_{p+1,i}(x), \tag{7}$$

and is based on a set of B-spline basis functions, $B_{p+1,i}(x)$, polynomials of degree $p+1$ on each subinterval, with corresponding unknown time dependent coefficients, $\underline{y}_{p+1,i}(t)$. Here, $NC_{p+1} = NINT \cdot p + 2$. As above, these unknowns are determined by imposing p collocation conditions per subinterval as well as the boundary conditions on $\overline{\underline{U}}(x,t)$. The collocation points in this case are the images of p Gauss points on $[0,1]$ mapped onto each subinterval. This leads to a system of DAEs whose solution gives the functions, $\underline{y}_{p+1,i}(t)$. In order to ensure that the two approximate solutions, $\underline{U}(x,t)$ and $\overline{\underline{U}}(x,t)$, are available at the same time t, the two DAE systems are provided to DASSL as one larger

DAE system so that DASSL treats both systems of DAEs with the same time-stepping strategy. See [**25**] for further details.

It is shown in [**7**] and [**10**] that a collocation solution of degree p has an error that is $O(h^{p+1})$, where h is the maximum mesh spacing. The difference between the two collocation solutions gives, asymptotically, an estimate of the error in the lower order collocation solution, $\underline{U}(x,t)$.

In BACOL, the following *a posteriori* spatial error estimates are computed. Denote the sth component of $\underline{U}(x,t)$ by $U_s(x,t)$ and the sth component of $\overline{\underline{U}}(x,t)$ by $\overline{U}_s(x,t)$. Let $ATOL_s$ and $RTOL_s$ be the absolute and relative tolerances for the s-th component of the approximate solution. BACOL computes a set of $NPDE$ normalized error estimates over the whole spatial domain of the form

$$E_s(t) = \sqrt{\int_a^b \left(\frac{U_s(x,t) - \overline{U}_s(x,t)}{ATOL_s + RTOL_s|U_s(x,t)|} \right)^2 dx}, \quad s = 1, \ldots, NPDE. \tag{8}$$

BACOL also computes a second set of $NINT$ normalized error estimates of the form

$$\widehat{E}_i(t) = \sqrt{\sum_{s=1}^{NPDE} \int_{x_{i-1}}^{x_i} \left(\frac{U_s(x,t) - \overline{U}_s(x,t)}{ATOL_s + RTOL_s|U_s(x,t)|} \right)^2 dx}, \quad i = 1, \ldots, NINT. \tag{9}$$

Note that $E_s(t), s = 1, \ldots, NPDE$, and $\widehat{E}_i(t), i = 1, \ldots, NINT$, are estimates of the error associated with the *lower* order solution, $\underline{U}(x,t)$.

The approximate solution, $\underline{U}(x,t)$, is accepted at the current time, t, if

$$\max_{1 \le s \le NPDE} E_s(t) \le 1. \tag{10}$$

Otherwise, based on the error estimates, $\widehat{E}_i(t), i = 1, \ldots, NINT$, BACOL attempts to construct a new mesh, i.e., perform a remeshing, that (i) has as many mesh points as necessary to yield an approximate solution whose estimated error will satisfy the user tolerances, and (ii) approximately equidistributes the estimated error over the subintervals of the new mesh. See [**25**] for further details.

3 Spatial error estimation based on superconvergent interpolants As mentioned above, [**7**] and [**10**] provide the standard convergence results for Gaussian collocation applied to a 1D parabolic PDE. In summary, these results say that, (i) over the entire spatial domain, the collocation solution has a spatial error that is $O(h^{p+1}) \equiv O(h^{k+2})$, where k is the number of collocation points per subinterval and $p = k+1$ is the degree of the piecewise polynomials representing the collocation solution on each subinterval, and that, (ii) at mesh points, both the collocation solution and its derivative have errors that are $O(h^{2(p-1)}) \equiv O(h^{2k})$ (and are thus superconvergent when $2k > k+2 \Rightarrow k > 2$).

An additional related result comes from collocation theory for boundary value ODEs. With appropriate assumptions, Theorem 5.140/Corollary 5.142 of [**4**] states that (letting $u(x)$ be the exact solution and $U(x)$ be the collocation solution)

$$u(x) - U(x) = u^{(k+2)}(x_i) P_k\left(\frac{x - x_i}{h_i}\right) h_i^{k+2} + O(h_i^{k+3}) + O(h^{2k}), \tag{11}$$

where $x_i < x < x_{i+1}$, $i = 0, \ldots, NINT - 1$, and where

$$P_k(\xi) = \frac{1}{k!} \int_0^{\xi} (t - \xi) \prod_{r=1}^{k} (t - \rho_r)\, dt. \tag{12}$$

Thus, for boundary value ODEs, one can expect to see higher accuracy in the collocation solution at points within each subinterval that correspond to roots of $P_k(\xi)$. To our knowledge, the corresponding result for the PDE case has not been proved. However, it appears that this result does hold for the 1D parabolic PDE case: [**3**] provides experimental evidence demonstrating that, for the spatial discretization of a 1D parabolic PDE by Gaussian collocation, the order of convergence described by the above BVODE result (11) also holds for the PDE case. To be specific, in BACOL, since the collocation points are chosen to be the images of the Gauss points on each subinterval, there are $p-3$ points internal to each subinterval where the lower order collocation solution, $\underline{U}(x,t)$, generally of order $p+1$, is *superconvergent*, i.e., of order $p+2$.

In [**2**], the authors construct a superconvergent interpolant (SCI) that interpolates, on a given subinterval, the superconvergent endpoint solution and derivative values, all $p-3$ of the superconvergent *solution* values internal to the subinterval, and *the closest available superconvergent solution values from each adjacent subinterval.* (For the leftmost and rightmost subintervals, the scheme interpolates the two closest superconvergent solution values available in the lone adjacent subinterval.)

The SCI interpolates a sufficient number of superconvergent solution and derivative values so that *the interpolation error is dominated by the data error.* The SCI is of order $p+2$; this is the same as the order of the higher order collocation solution. However, the authors also present an expression for the interpolation error that shows its dependence on the ratios of the size of the current subinterval to the sizes of the immediately adjacent subintervals. When one of these ratios is large, this can lead to a large interpolation error which can impact negatively on the accuracy of the error estimate produced by the SCI-based scheme. See [**2**] for further details.

4 Spatial error estimation based on a lower order interpolant having an asymptotically correct interpolation error While the cost of computing the higher order collocation solution is slightly more expensive than that of the lower order collocation solution, the costs are in fact comparable. It follows that the overall costs for BACOL will be approximately halved if we compute only one of the two global collocation solutions. In the LOI approach, we will compute only the *higher* order global collocation solution, $\overline{\underline{U}}(x,t)$, and, instead of computing the lower order global collocation solution, $\underline{U}(x,t)$, we will construct an interpolant to replace it. The interpolant will be based on evaluations of the higher order collocation solution and will be constructed so that the leading order term in the interpolation error expansion for this interpolant agrees (asymptotically) with the leading order term in the error expansion for the lower order collocation solution. Thus the spatial error estimate we obtain from this approach will be asymptotically equivalent to the spatial error estimate computed by BACOL. In this case, *the interpolant will be designed so that the interpolation error dominates the data error*; this will imply that we will interpolate fewer points than in the SCI case.)

This idea is similar to the well known “local extrapolation” approach sometimes employed in software for the numerical solution of initial value ODEs; see, e.g, [**12**]. In that approach, a Runge-Kutta formula pair is used to compute two solutions of orders q and $q+1$. In standard mode, the lower order solution is propagated forward in time and the higher order solution is used only to obtain an estimate of the error in the lower order solution; error control is based on this estimate. In local extrapolation mode, only the *higher* order solution is propagated but error control is again based on the error estimate for the *lower* order solution.

While the details are different here, the general approach has been considered by Moore in his work on interpolation based error estimation [**19, 20, 21, 22**]. In Moore's work, the computation of the primary finite element solution is based on a spatial discretization that uses a finite element Galerkin technique with a piecewise polynomial hierarchical spatial basis. The error estimate is based on a Lobatto interpolant for which the leading term in the interpolation error agrees asymptotically with the leading term in the error for the finite element solution. Moore refers to this as the asymptotic equivalence property. One can then obtain an approximation for the error in the numerical solution of the PDE by estimating the error in the interpolant. The explicit form for the interpolation error estimate comes from an extension of the error formula for standard Lagrange interpolation.

As mentioned in the previous section, it appears that the error expansion associated with the lower order collocation solution, $\underline{U}(x,t)$, has the form given in (12). Our goal is then to construct an interpolant, $\widetilde{\underline{U}}(x,t)$, the LOI, such that, asymptotically, *the leading order term of the interpolation error equals the leading order term of the collocation error*, given in (11).

The LOI is a piecewise polynomial; we focus on how the polynomial interpolant for a given subinterval is constructed. From standard interpolation theory, it is clear that the leading term in the error expansion for the interpolant will include the factors $h_i^{k+2}u^{(k+2)}(x_i,t)$ as long as we construct the interpolant so that it is based on $k+2$ data values and as long as the error in each of the data values is at least one order higher, i.e., at least $O(h_i^{k+3})$, so that the *interpolation* error dominates the *data* error. Since the data values we will employ for the interpolant are obtained from the higher order collocation solution, those values will have an error that is $O(h_i^{k+3})$ *for any choice of interpolation points.* The remaining factor in the leading term of the error expansion for the interpolant will depend on the location of the interpolation points within the interval and these points must be chosen so that this factor equals $P_k\big(\frac{x-x_i}{h_i}\big)$, so that the leading term in the interpolation error with agree with the leading term in the collocation error.

We consider a Hermite-Birkhoff interpolant framework for the LOI on each subinterval. The paper [**11**] provides explicit expressions for the basis functions that implement Hermite-Birkhoff interpolant in our case. Let $s_1 = x_i$, $s_2 = x_{i+1}$, and let $w_j,\ j = 1,\dots,k-2$, be the points internal to the ith interval. (This gives a total of $4+(k-2) = k+2$ interpolation points.) The LOI is based on evaluations of (i) the collocation solution, $\overline{\underline{U}}(x,t)$, at the end points of the subinterval and at $k-2$ appropriately chosen points internal to the subinterval, and (ii) evaluations of the first derivative of the collocation solution at the endpoints of the subinterval. The LOI has the form

$$\widetilde{\underline{U}}(x,t) = \sum_{j=1}^{2} H_j(x)\underline{U}(s_j,t) + h\sum_{j=1}^{2} \overline{H}_j(x)\underline{U}'(s_j,t) + \sum_{j=1}^{k-2} G_j(x)\underline{U}(w_j,t),$$

where $x \in [x_i, x_{i+1}]$, $h = x_{i+1} - x_i$,

$$H_j(x) = (1-(x-s_j)\gamma_j)\frac{\eta_j^2(x)\phi(x)}{\eta_j^2(s_j)\phi(s_j)},$$

$$\overline{H}_j(x) = (x-s_j)\frac{\eta_j^2(x)\phi(x)}{\eta_j^2(s_j)\phi(s_j)},$$

$$G_j(x) = \frac{\phi_j(x)\eta^2(x)}{\phi_j(w_j)\eta^2(w_j)},$$

$$\phi(x) = \prod_{r=1}^{k}(x-w_r), \quad \phi_j(x) = \prod_{\substack{r=1\\ r\neq j}}^{k}(x-w_r),$$

$$\eta(x) = \prod_{r=1}^{2}(x-s_r), \quad \eta_j(x) = \prod_{\substack{r=1\\ r\neq j}}^{2}(x-s_r),$$

and

$$\gamma_j = \sum_{i=1}^{k}\frac{1}{s_j-w_i} + 2\sum_{\substack{i=1\\ i\neq j}}^{2}\frac{1}{s_j-s_i}.$$

The basic form for the leading term in the interpolation error expression for a given k (and a given t) has the general form

$$\frac{1}{(k+2)!}h_i^{k+2}u^{(k+2)}(x_i,t)\widetilde{P}_k(\xi), \tag{13}$$

where $\widetilde{P}_k(\xi)$ is a polynomial that depends on the points where interpolation of solution and derivative data is performed. (The detailed error expansion for general k, is given in [**11**] but since the expression is somewhat complicated, we do not reproduce it here.) Rather, we consider the details of the development of the LOI for the case when $k = 5$; the general approach will be clear from this discussion.

For $k = 5$, the leading term in the *collocation* error expansion is

$$\frac{1}{5!} h_i^7 u^{(7)}(x_i, t) P_5\left(\frac{x - x_i}{h_i}\right)$$

where

$$\begin{aligned} P_5(\xi) &= \frac{1}{540}\,\xi^2\,(1 - 2\,\xi)\left(6\xi^2 - 6\xi + 1\right)(\xi - 1)^2 \\ &= \frac{1}{42}\,\xi^2\left(\frac{1}{2} - \xi\right)\left(\xi - \frac{1}{2} + \frac{1}{6}\sqrt{3}\right)\left(\xi - \frac{1}{2} - \frac{1}{6}\sqrt{3}\right)(\xi - 1)^2 . \end{aligned}$$

The roots of this polynomial are $0,\ 0,\ \frac{1}{2},\ \frac{1}{2} \pm \frac{1}{6}\sqrt{3},\ 1$ and 1. Substituting this expression for $P_5(\xi)$ into the general form for the leading term in the collocation error above gives

$$\begin{aligned} &\left(\frac{1}{5!}\right) h_i^7 u^{(7)}(x_i, t) \left[\left(\frac{1}{42}\right)\xi^2\left(\frac{1}{2} - \xi\right)\right. \\ &\qquad\qquad \left.\times \left(\xi - \frac{1}{2} + \frac{1}{6}\sqrt{3}\right)\left(\xi - \frac{1}{2} - \frac{1}{6}\sqrt{3}\right)(\xi - 1)^2\right], \end{aligned}$$

or

$$\begin{aligned} &\left(\frac{1}{7!}\right) h_i^7 u^{(7)}(x_i, t)\,\xi^2\left(\frac{1}{2} - \xi\right) \\ &\qquad\qquad \times \left(\xi - \frac{1}{2} + \frac{1}{6}\sqrt{3}\right)\left(\xi - \frac{1}{2} - \frac{1}{6}\sqrt{3}\right)(\xi - 1)^2 . \end{aligned} \tag{14}$$

In order to construct the LOI such that the leading order term in its error expansion matches the above collocation error term (14), we need to choose the interpolation points to be the roots of $P_5(\xi)$, with the understanding that the repeated roots (at 0 and 1) mean that the interpolant must interpolate the higher order collocation solution at 0 and 1 and the derivative of the interpolant must interpolate the derivative of the higher order collocation solution at 0 and 1. Furthermore, the interpolant must interpolate the higher order collocation solution at $\frac{1}{2}$ and $\frac{1}{2} \pm \frac{1}{6}\sqrt{3}$. Thus, the total number of interpolation points is 7: interpolation of solution and derivative values at the endpoints and interpolation of solution values at the three interior points. (For general k, this is 4 endpoint solution and derivative values and $k-2$ internal solution values for a total of $k+2$ data values to be interpolated.)

The general form for the leading term in the interpolation error is given in (13). For the specific case of $k=5$, based on the interpolation points specified above, the leading term in the *interpolation* error of the LOI is

$$\frac{1}{7!}h_i^{k+2}u^{(k+2)}(x_i,t)\xi^2\left(\frac{1}{2}-\xi\right) \times \left(\xi-\frac{1}{2}-\frac{1}{6}\sqrt{3}\right)\left(\xi-\frac{1}{2}+\frac{1}{6}\sqrt{3}\right)(\xi-1)^2,$$

and we see that this matches the leading order term, (14), in the error expression for the collocation solution, (11).

5 A numerical comparison of the spatial error estimation schemes In this section, we will present numerical results in which the LOI error estimation scheme is compared with the original BACOL error estimation scheme as well as the SCI error estimation scheme. (A much larger set of test results is available in [**3**]).

The first test problem, taken from [**18**], is

$$u_t = u_{xx} + \pi^2 \sin(\pi x), \quad 0 < x < 1, \quad t > 0, \tag{15}$$

with initial condition

$$u(x,0) = 1, \quad 0 \le x \le 1,$$

and boundary conditions

$$u(0,t) = u(1,t) = 1, \quad t > 0,$$

for which the exact solution is

$$u(x,t) = 1 + \sin(\pi x)\left(1 - e^{-\pi^2 t}\right).$$

We integrate from $t=0$ to $t=1$ with $ATOL = RTOL = 10^{-8}$.

The second test problem (see, e.g., [**24**]) is

$$u_t = \epsilon u_{xx} - u u_x, \qquad 0 < x < 1, \quad t > 0, \quad \epsilon > 0, \tag{16}$$

with the initial condition and boundary conditions chosen so that exact solution is given by

$$u(x,t) = \frac{1}{2} - \frac{1}{2}\tanh\left(\frac{1}{4\varepsilon}\left(x - \frac{t}{2} - \frac{1}{4}\right)\right),$$

where ε is a problem dependent parameter. We choose $\epsilon = 10^{-3}$ and integrate from $t = 0$ to $t = 1$, with $ATOL = RTOL = 10^{-8}$. For $\varepsilon = 10^{-3}$, the solution is plotted in Figure 1.

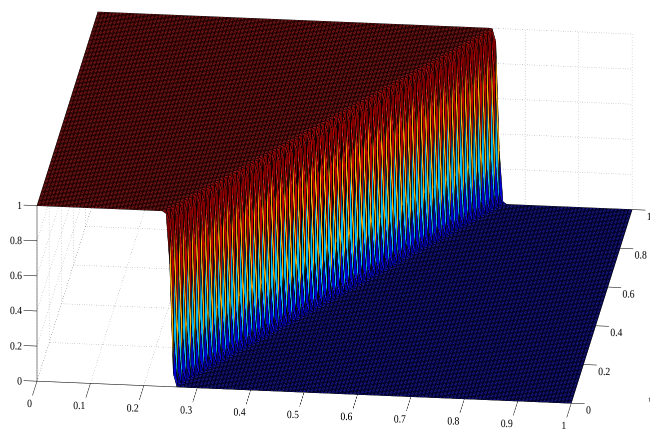

FIGURE 1: Solution of Burgers' equation with $\varepsilon = 10^{-3}$. Initially ($t = 0$), there is a sharp layer region near 0.25. As t goes from 0 to 1, the layer moves to the right and is near 0.75 when $t = 1$.

When BACOL is applied to the first test problem, (15), with $k = 3$, the error estimates obtained by each of the schemes at the final time compare well with each other and with the true error; see Figure 2. These results were obtained by letting the original error estimation scheme of BACOL provide the error estimates for mesh adaptation and for the spatial error acceptance tests. However, as is evident from the figure, the SCI and LOI error estimates are quite similar to those of the original BACOL error estimation scheme and thus using the SCI or LOI error estimates to control the mesh would give similar results. The locations of the mesh points are indicated by tick marks along the horizontal axis.

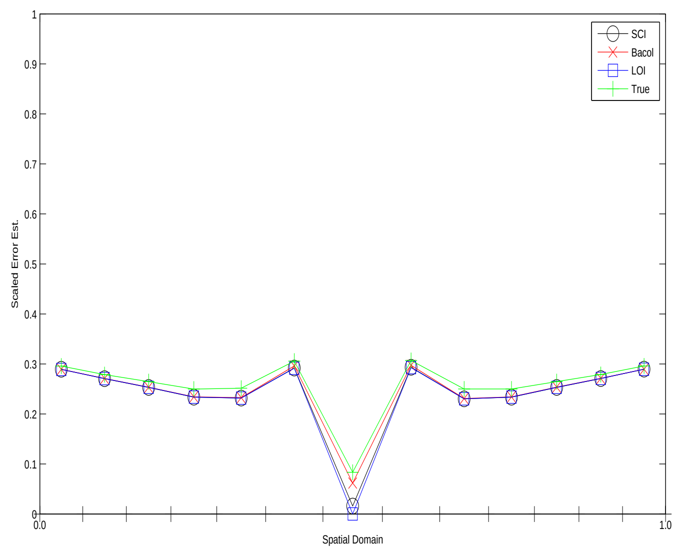

FIGURE 2: Plot of the BACOL, SCI, and LOI scaled error estimates and the scaled exact error (as in (8)) for the collocation solution of (15), with $tol = 10^{-8}$. The problem is solved using $k = 3$. The spatial mesh at the final time has $NINT = 13$.

For the second test problem, (16), we consider the case where $k = 6$. We again plot the error estimates computed by each of the schemes at the final time. Since this problem is much more difficult very nonuniform spatial meshes are required. The locations of the mesh points are indicated by tick marks along the horizontal axis. It is appropriate to consider a set of results in which we allow each of the three error estimation schemes the opportunity to control the mesh.

In Figure 3, the BACOL error estimate controls the mesh adaptivity. We see that there is generally good agreement among the schemes except that the SCI estimates substantially overestimate the true error on the leftmost and rightmost subintervals. (From Figure 3, it is clear that the leftmost and rightmost subintervals are substantially larger than the adjacent subintervals and we recall that the SCI scheme can yield overestimates of the error is such cases.)

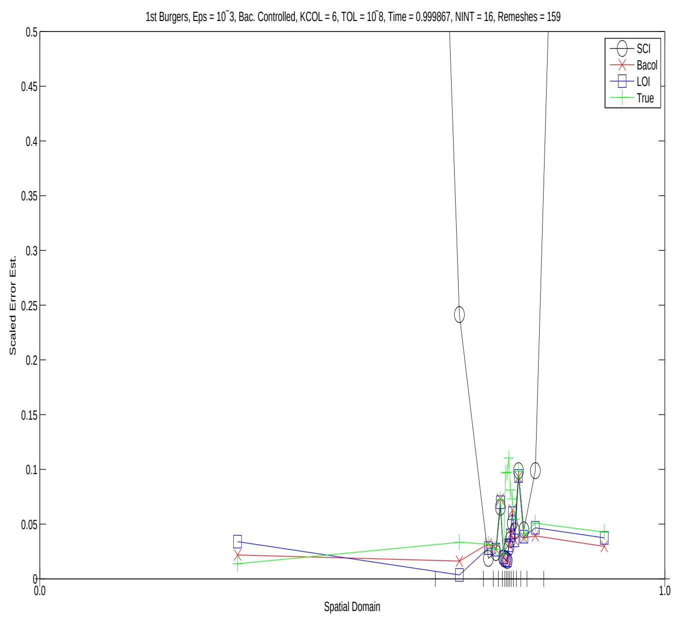

FIGURE 3: Plot of the BACOL, SCI, and LOI scaled error estimates and the scaled exact error (as in (8)) for the collocation solution of (16) with $\epsilon = 10^{-3}$, $tol = 10^{-8}$. The problem is solved with $k = 6$ and **the BACOL error estimates are used to control the mesh.** The spatial mesh at the final time has $NINT = 16$ and a total of 159 remeshings were performed.

In Figure 4, we show the results of again solving the second test problem (16) but this time *we allow the SCI error estimates to control* *mesh refinement.* Comparing the results in Figures 3 and 4, we see that when the SCI estimates are used to control the mesh, the mesh ratios are not as extreme as in the previous case since the large overestimates of the errors lead to a reduction in the sizes of the subintervals in those parts of the spatial domain where the error estimates are large. Thus, to some extent, the use of the SCI error estimates to control the mesh "self-corrects" the overestimates of the error that are characteristic of this approach, at the cost of adding extra points to the mesh.

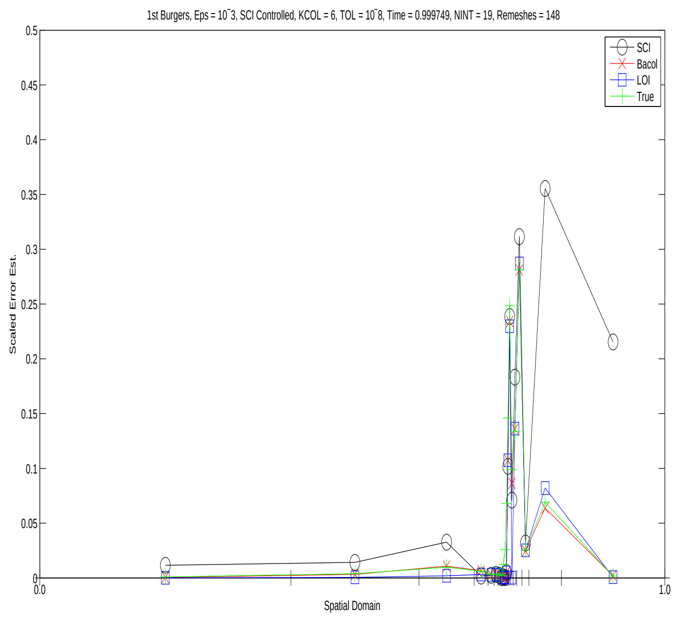

FIGURE 4: Plot of the BACOL, SCI, and LOI scaled error estimates and the scaled exact error (as in (8)) for the collocation solution of (16) with $\epsilon = 10^{-3}$, $tol = 10^{-8}$. The problem is solved with $k = 6$ and **the SCI error estimates are used to control the mesh.** The spatial mesh at the final time has $NINT = 19$ and a total of 148 remeshings were performed.

Figure 5 compares the error estimation schemes when *the LOI error estimates are used to control the mesh.* We see generally good agreement among the schemes except that the SCI scheme gives overestimates of the error when the adjacent subinterval ratios are large. *In particular, there is good agreement between the LOI error estimates and those from the original BACOL error estimation scheme.*

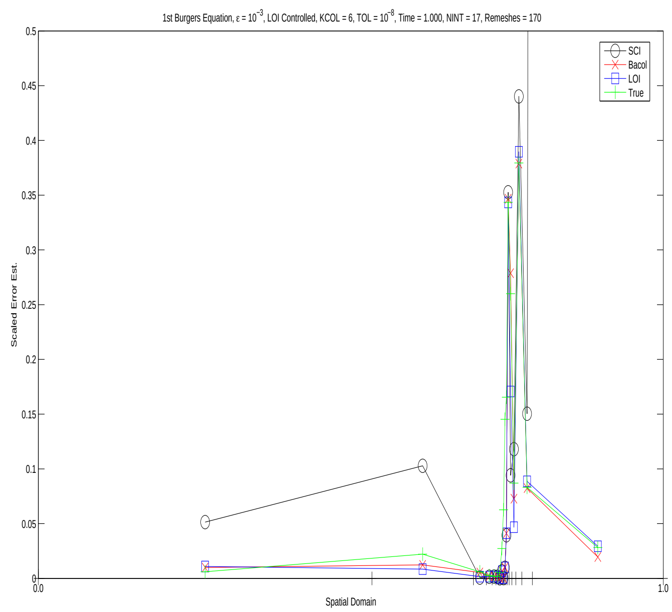

FIGURE 5: Plot of the BACOL, SCI, and LOI scaled error estimates and the scaled exact error (as in (8)) for the collocation solution of (16) with $\epsilon = 10^{-3},\ tol = 10^{-8}$. The problem is solved with $k = 6$ and **the LOI error estimates are used to control the mesh.** The spatial mesh at the final time has $NINT = 17$ and a total of 170 remeshings were performed.

The above results are generally representative of the larger set of results reported in [**3**]. The three schemes generally use about the same number of subintervals and perform about the same number of remeshings. There is generally good agreement between the BACOL and LOI error estimates. However, SCI estimates overestimate the error when adjacent subinterval ratios are large; this can be corrected to some extent when the SCI error estimates are used to control the mesh but this sometimes leads to the spatial mesh having extra points, thus increasing the cost. As well, for smaller k values, we have observed that computations in which the SCI estimate controls the mesh sometimes fail [**3**].

6 Computational costs for the BACOL, SCI, and LOI error estimation schemes As mentioned earlier, the BACOL error estimate requires the computation of two global collocation solutions, one using piecewise polynomials of degree p and the other using piecewise polynomials of degree $p+1$. For each of these solutions the most significant cost is the setup and solution of the Newton matrices that arise and, assuming a mesh of $NINT$ subintervals, k collocation points per subinterval ($k = p-1$), and $NPDE$ equations, and the use of a linear system solver specifically designed to handle the almost block diagonal structure of the Newton matrices, these costs are $O(NINT(NPDE \times k)^3)$. These are obviously more significant for large $NPDE$ and k values. The computation associated with the BACOL error estimates requires the evaluation of the two global collocation solutions at many points within the problem domain; these costs are $O(NINT \times NPDE \times k)$.

When the SCI or LOI approaches are used for error estimation, one of the two global collocation solutions need not be computed and this represents a significant saving in the overall cost. The corresponding error estimates are obtained through evaluations of the remaining collocation solution and the SCI or LOI. These costs are $O(NINT \times NPDE \times k)$.

It is clear then that for $NPDE > 1$ and larger k values, the cost of the error estimation scheme employed by BACOL will be significantly greater than the corresponding cost for either of the interpolation based error estimation schemes. For a sufficiently large combination of $NPDE$ and k values, we expect that the new version of BACOL based on the LOI scheme will be about twice as fast as the current version.

7 Conclusions and future work The SCI and LOI schemes appear to provide reasonable, low cost spatial error estimates of comparable quality to those currently computed by BACOL. The SCI scheme can overestimate the error when the adjacent subinterval ratios are large but it can compensate for this to some extent by adding extra points to the mesh. Also, for low k values, the SCI approach can experience difficulties (the computation can fail to start successfully). The LOI approach appears to have the advantages with respect to efficiency of the SCI approach but also appears to generally show better agreement with the original BACOL error estimates. It is not sensitive to adjacent subinterval ratios and does not exhibit difficulties at the beginning of any of the computations [**3**].

The modified form of BACOL that was used to obtain the numerical results reported here computes the BACOL, SCI, and LOI error estimates. This requires that the primary and secondary collocation solutions both be computed. Ongoing work involves further modification of BACOL so that only one collocation solution and either the SCI error estimate or the LOI error estimate are computed. This new version of BACOL should have comparable performance to that of the current version but with about twice the efficiency.

While the approach discussed in this paper is employed together with a collocation solution to yield an error estimate, it could be applied to any continuous numerical solution of a PDE to yield an error estimate. We chose the interpolation points to force the interpolation error to asymptotically agree with the collocation error but a different choice of interpolation points would allow the interpolation error to model a different type of error term consistent with the type of continuous solution approximation computed.

Since both the interpolation-based error estimates considered here are low cost, it is also possible that both could be computed for a given collocation solution and then together these error estimates could be used to improve the reliability of the overall error estimation algorithm.

It may be possible to generalize the interpolation-based error estimates discussed in this paper to provide error estimates for two dimensional PDEs, provided that a tensor product collocation framework on a rectangular mesh is employed.

Acknowledgments The authors wish to acknowledge the collaboration of Patrick Keast, of Dalhousie University, in the earlier stages of this work, and a number of helpful discussions with Zhi Li, of Saint Mary's University.

REFERENCES

1. M. Ainsworth and J. T. Oden, *A posteriori error estimation in finite element analysis*, Pure and Applied Mathematics (New York), Wiley-Interscience, John Wiley & Sons, New York, 2000.
2. T. Arsenault, T. Smith and P. H. Muir, *Superconvergent interpolants for efficient spatial error estimation in 1D PDE collocation solvers*, Canad. Appl. Math. Q. **17**(3) (2009) 409–431.
3. T. Arsenault, T. Smith, P. H. Muir and P. Keast, *Interpolation Based Spatial Error Estimation Schemes for 1D Parabolic Collocation Solvers*, Saint Mary's University, Dept. of Mathematics and Computing Science Technical Report Series, Technical Report 2011_001, cs.smu.ca/tech_reports, 2011.
4. U. M. Ascher, R. M. M. Mattheij and R. D. Russell, *Numerical solution of boundary value problems for ordinary differential equations*, in: Classics in Applied Mathematics, **13**, SIAM, Philadelphia, PA, 1995.
5. J. G. Blom and P. A. Zegeling, *Algorithm 731: A moving-grid interface for systems of one-dimensional time-dependent partial differential equations*, ACM Trans. Math. Softw. **20**(2) (1994), 194–214.
6. K. E. Brenan, S. L. Campbell and L. R. Petzold, *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*, SIAM, Philadelphia, PA, 1989.
7. J. H. Cerutti and S. V. Parter, *Collocation methods for parabolic partial differential equations in one space dimension*, Numer. Math. **26**(3) (1976), 227–254.
8. D03PPF, *NAG Fortran library*, The Numerical Algorithms Group, Ltd., Wilkinson House, Oxford, UK.
9. C. de Boor, *A Practical Guide to Splines*, in: Applied Mathematical Sciences, **27**, Springer-Verlag, New York, 1978.
10. J. Douglas, Jr. and T. Dupont, *Collocation Methods for Parabolic Equations in a Single Space Variable*, in: Lecture Notes in Mathematics, Vol. 385, Springer-Verlag, Berlin, 1974.
11. W. F. Finden, *An error term and uniqueness for Hermite-Birkhoff interpolation involving only function values and/or first derivative values*, J. Comput. Appl. Math. **212**(1) (2008), 1–15.
12. E. Hairer, S. P. Nørsett and G. Wanner, *Solving Ordinary Differential Equations. I*, second ed., in Springer Series in Computational Mathematics, Vol. 8, Springer-Verlag, Berlin, 1993.
13. E. Hairer and G. Wanner, *Solving Ordinary Differential Equations. II*, second ed., in Springer Series in Computational Mathematics, Vol. 14, Springer-Verlag, Berlin, 1996.
14. W. Huang and R. D. Russell, *A moving collocation method for solving time dependent partial differential equations*, Appl. Numer. Math. **20**(1-2) (1996), 101–116.
15. W. Huang and R. D. Russell, *Adaptive Moving Mesh Methods*, in: Applied Mathematical Sciences, Vol 174, Springer, New York, 2011.
16. P. Keast and P. H. Muir, *Algorithm 688: EPDCOL: a more efficient PDECOL code*, ACM Trans. Math. Softw. **17**(2) (1991), 153–166.
17. Z. Li, *B-spline collocation software for 2D parabolic PDEs*, M.Sc. Thesis (in preparation), Saint Mary's University, 2012.
18. N. K. Madsen and R. F. Sincovec, *Algorithm 540: PDECOL, general collocation software for partial differential equations [D3]*, ACM Trans. Math. Softw. **5**(3) (1979), 326–351.

19. P. K. Moore, *Interpolation error-based a posteriori error estimation for two-point boundary value problems and parabolic equations in one space dimension*, Numer. Math. **90**(1) (2001), 149–177.
20. P. K. Moore, *Applications of Lobatto polynomials to an adaptive finite element method: a posteriori error estimates for hp-adaptivity and grid-to-grid interpolation*, Numer. Math. **94**(2) (2003), 367–401.
21. P. K. Moore, *An implicit interpolation error-based error estimation strategy for hp-adaptivity in one space dimension*, J. Numer. Math. **12**(2) (2004), 143–167.
22. P. K. Moore, *Interpolation error-based a posteriori error estimation for hp-refinement using first and second derivative jumps*, Appl. Numer. Math. **48**(1) (2004), 63–82.
23. R. Wang, P. Keast and P. H. Muir, *BACOL: B-spline Adaptive COLlocation software for 1-D parabolic PDEs*, ACM Trans. Math. Software **30**(4) (2004), 454–470.
24. R. Wang, P. Keast and P. H. Muir, *A comparison of adaptive software for 1D parabolic PDEs*, J. Comput. Appl. Math. **169**(1) (2004), 127–150.
25. R. Wang, P. Keast and P. H. Muir, *A high-order global spatially adaptive collocation method for 1-D parabolic PDEs*, Appl. Numer. Math. **50**(2) (2004), 239–260.
26. R. Wang, P. Keast and P. H. Muir, *Algorithm 874: BACOLR—spatial and temporal error control software for PDEs based on high-order adaptive collocation*, ACM Trans. Math. Software **34**(3) (2008), Art. 15, 28.

University of Western Ontario, London, ON, Canada N6A 5B8

Scotiabank, Halifax, NS, Canada B3J 1N9

Saint Mary's University, Halifax, NS, Canada B3H 3C3,
E-mail address: muir@smu.ca

Saint Mary's University, Halifax, NS, Canada B3H 3C3