Available online at www.sciencedirect.com

SCIENCE @ DIRECT®

Journal of Computational and Applied Mathematics 188 (2006) 265–282

JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS

www.elsevier.com/locate/cam

# Adaptive multiquadric collocation for boundary layer problems

Leevan Ling[a, b, *, 1], Manfred R. Trummer[a, b, 2]

[a] *The Pacific Institute for the Mathematical Sciences, Canada V5A 1S6*
[b] *Department of Mathematics, Simon Fraser University, 8888 University Drive, Burnaby, BC, Canada V5A 1S6*

Received 23 October 2003; received in revised form 21 May 2004

**Abstract**

An adaptive collocation method based upon radial basis functions is presented for the solution of singularly perturbed two-point boundary value problems. Using a multiquadric integral formulation, the second derivative of the solution is approximated by multiquadric radial basis functions. This approach is combined with a coordinate stretching technique. The required variable transformation is accomplished by a conformal mapping, an iterated sine-transformation. A new error indicator function accurately captures the regions of the interval with insufficient resolution. This indicator is used to adaptively add data centres and collocation points. The method resolves extremely thin layers accurately with fairly few basis functions. The proposed adaptive scheme is very robust, and reaches high accuracy even when parameters in our coordinate stretching technique are not chosen optimally. The effectiveness of our new method is demonstrated on two examples with boundary layers, and one example featuring an interior layer. It is shown in detail how the adaptive method refines the resolution.

*Keywords:* Multiquadric; Radial basis function; Integral formulation; Singular perturbations; Boundary layer problems; High-order discretizations; Spectral accuracy; Adaptive

[*] Corresponding author. Department of Mathematics, Simon Fraser University, 8888 University Drive, Burnaby, BC, Canada V5A 1S6. Tel.: +604 930 0932; fax: +604 648 8942.

*E-mail address:* lling@alumni.sfu.ca (L. Ling).

[1] The research of this author was supported by a Natural Science and Engineering Research Council of Canada (NSERC) postgraduate scholarship and by NSERC discovery Grant OGP003690.

[2] The research of this author was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) discovery Grant OGP003690.

doi:10.1016/j.cam.2005.04.018

## 1. Introduction

Recently, there has been a great deal of interest in radial basis functions (RBFs) for interpolation problems, and as a tool for numerically solving differential equations. The idea of RBFs is to use linear combinations of translates of a function $\varphi(r)$ of one real variable, centred at "data centres" or "knots" $x_k$, to approximate an unknown function:

$$s(x)=\sum_{k=1}^{n}\lambda_k\varphi(\|x-x_k\|)+\text{low-order polynomials.} \tag{1}$$

Common choices for such functions $\phi$ are

Multiquadric (MQ): $\varphi(r)=\sqrt{r^2+c^2}$,
Inverse multiquadric: $\varphi(r)=\left(r^2+c^2\right)^{-1/2}$,
Gaussian: $\varphi(r)=\mathrm{e}^{-r^2/c^2}$.

The parameter $c$ is the so-called shape parameter. As $c\to\infty$, the basis functions are becoming increasingly flat. The linear combination (1) can be used in an interpolation procedure, or when trying to find the solution of a differential equation.

The distinct scattered data centres $x_k$ can be chosen arbitrarily in the domain of interest. Since RBF methods only act upon the information at the data centres, the method requires no further domain or surface integration or discretization. Hence, RBFs lead to "meshless methods".

In [9], Franke's numerical experiments compared 29 interpolation methods with analytic two-dimensional test functions. According to his results, the most powerful methods are the radial basis function methods based on the multiquadric basis function suggested by Hardy [12] and the thin plate spline. Madych and Nelson [20,21] showed that interpolation with the multiquadric basis is exponentially convergent. Their proof is based on reproducing kernel Hilbert spaces. Wu and Schaback [29] use a different technique to prove the same results. Their technique is general enough to handle the case of interpolation with the power spline and the thin plate spline. Since the Hilbert space is small when the radial basis function is smooth, the function being interpolated has to be extremely smooth for the error estimates to apply. Yoon [30] showed that the multiquadric basis function method converges exponentially in a Sobolov space. This was verified numerically by Fedoseyev et al. [7].

Meshless methods have been under intense scrutiny in an effort to avoid some of the problems associated with more traditional schemes. The 1990s have seen a rise in the use of meshless methods for solving partial differential equations (PDEs), led by methods from the finite element community, including the Partition of Unity Method of Babuška and Melenk [3], the h-p Cloud Method of Duarte and Oden [6], and the Element Free Galerkin Method of Belytschko et al. [4]. Motivated by the success of surface approximation, Kansa [15,16] pioneered the use of RBFs for the numerical solution of the Navier–Stokes equations. Since then, RBFs have been used to solve a variety of ordinary and PDEs.

In this paper we employ RBFs to solve boundary value problems with very thin layers. Our aim is to demonstrate that RBF methods are capable of achieving high accuracy and robustness through the introduction of adaptivity. Our solution method below is based on an integral formulation of multiquadric collocation. Integration is a smoothing operation; the convergence rate may be expected to accelerate in line with the convergence rate estimates of Madych and Nelson. Further applications of the RBF integral formulation can be found in Mai-Duy and Tran-Cong [22,23], Kansa et al. [17].

We consider the singularly perturbed linear two-point boundary value problem (BVP)

$$\varepsilon u''(x) + p(x)u'(x) + q(x)u(x) = f(x) \quad \forall x \in [a,b],$$
$$u(a) = \alpha, \quad u(b) = \beta, \tag{2}$$

where $\varepsilon > 0$ denotes a fixed (small) constant. In many applications (2) possesses boundary layers, i.e., regions of rapid change in the solution near the endpoints with widths o(1) as $\varepsilon \to 0$.

It would not be difficult to extend the methods to nonlinear problems, although for small perturbation parameter $\varepsilon$ one may have to use continuation methods as well.

One can discretize (2) with finite difference methods, piecewise polynomials [1,2], spectral collocation methods (e.g., [27]), with radial basis functions [19] as suggested by expansion (1), or with any other discretization method having $n$ degrees of freedom. With a very small parameter $\varepsilon$ in (2) large $n$ is required to obtain accurate solutions. For good resolution of the numerical solution at least one of the collocation points should lie in the boundary layer. For example, if the problem possesses a boundary layer of width $\mathcal{O}(\varepsilon)$, then on a uniform grid with $\mathcal{O}(n^{-1})$ spacing between the points we need $n = \mathcal{O}(\varepsilon^{-1})$, which is not practical when $\varepsilon \ll 1$. Therefore, most numerical methods use specially designed grids that contain more points in and around the layer(s). For instance, Miller et al. [24] developed a successful upwind central difference scheme on a piecewise uniform mesh. Gartland [10] and Vulanović [28] suggested exponentially distributed grid points. These special meshes have a limitation: collocation points must not coincide numerically.

Polynomial spectral collocation (PSC) methods have underlying grids which are denser near the boundary, and hence are attractive in solving boundary layer problems, see [5]. Tang and Trummer [27] employed a Chebyshev spectral collocation (a. k. a. pseudospectral) method with coordinate stretching, and achieved a good resolution of boundary layers with relatively few collocation points. The coordinate stretching is accomplished via an analytic variable transformation; an advantage of the transformed BVP is that many more collocation points can be placed in the boundary layer without causing numerical difficulty. Recent developments of spectral collocation method can be found in [8,25,26].

In [19], the authors combined the transformation technique used in [27] with an MQ integral formulation to solve problems with thin boundary layers. In the computational domain we have a fixed, well separated grid, so the scheme does not suffer from numerical coincidence of collocation points. Furthermore, the "meshless" MQs allow a simple modification when there is only one layer. Numerical comparisons with the PSC method of [27], and Hon's adaptive MQ scheme [14] show that our MQ scheme can achieve superior accuracy for problems with very thin layers and relatively smooth solutions.

In this paper, we build upon the ideas of [19] to develop a robust adaptive scheme. The new adaptive scheme is much less sensitive to the number of transforms applied. It does not provide a huge improvement on the non-adaptive scheme using the optimal variable transform, but performs well for a range of transforms. Furthermore, our scheme shows impressive results for extremely thin boundary layers, i.e., very small $\varepsilon$ in (2).

The paper is organized as follows. In Section 2, we provide an overview of the multiquadric scheme with integral formulation [19]. The adaptive scheme is introduced in Section 3. Section 4 provides detailed numerical results for three problems: a singularly perturbed boundary value problem with an extremely small perturbation parameter $\varepsilon$, namely $\varepsilon = 10^{-12}$; an exponentially ill-conditioned boundary layer problem; and an interior layer problem with a moderately small perturbation parameter $\varepsilon$.

## 2. MQ integral formulation of the transformed BVP

This section summarizes the features of our non-adaptive scheme of [19] which forms the basis for the new adaptive scheme introduced in this paper. Without loss of generality we assume $[a, b]=[-1, 1]$. For $m \geqslant 1$ we consider the one-to-one mapping $x(\xi)=g_m(\xi)$ given by

$$g_0(\xi)=\xi, \quad g_m(\xi)=\sin\left(\frac{\pi}{2}g_{m-1}(\xi)\right), \quad m \geqslant 1. \tag{3}$$

As can be seen from Fig. 1 the graphs of the transforms are flat towards the end points of the interval. This means that the very thin boundary layer is mapped onto a much wider region. Effectively, collocation points are moved into the thin region. For example, for equidistant points $\xi_j=2j/N-1$, $j=0,\dots,N$, we have the estimate (see [19])

$$g_m(\xi_1)-g_m(\xi_0)=g_m(\xi_N)-g_m(\xi_{N-1})=\mathcal{O}(N^{-2^m}). \tag{4}$$

With the variable transformation $x \mapsto \xi(x)$, the singularly perturbed linear BVP (2) is transformed into

$$\begin{aligned}&\varepsilon v''(\xi)+P(\xi)v'(\xi)+Q(\xi)v(\xi)=F(\xi) \quad \forall \xi \in [a,b],\\ &v(a)=\alpha, \quad v(b)=\beta,\end{aligned} \tag{5}$$

where $v$ is the transplant of $u$, $v(\xi)=u(x(\xi))$. Throughout the paper we refer to $x$ as the physical variable, and to $\xi$ as the computational variable. The transformed coefficients are given by

$$\begin{aligned}&P(\xi)=\frac{p(x)}{\xi'(x)}+\varepsilon\frac{\xi''(x)}{\xi'(x)^2},\\ &Q(\xi)=\frac{q(x)}{\xi'(x)^2}, \quad F(\xi)=\frac{f(x)}{\xi'(x)^2},\end{aligned} \tag{6}$$

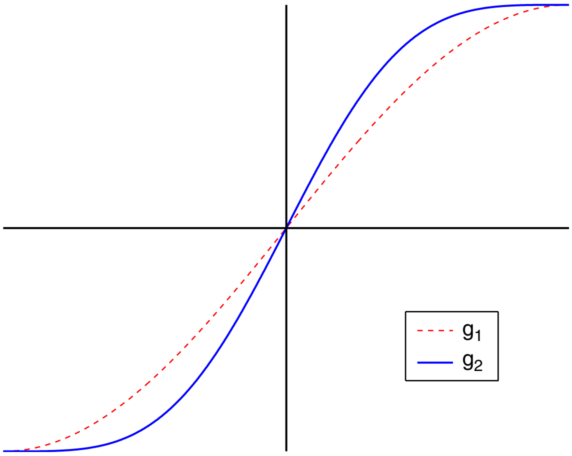

Fig. 1. Variable transformation $g_1(\xi)$ and $g_2(\xi)$.

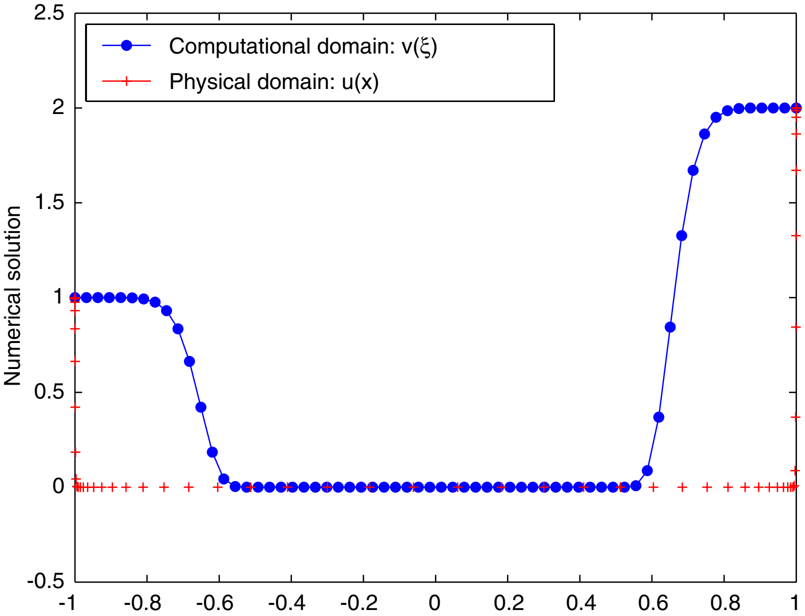

Fig. 2. Solution of a BVP plotted against the computational variable and the physical variable, respectively.

where $x = x(\xi)$. For the SINE-transformations $g_m$ (3), the quantity $1/\xi'(x)$ can be easily calculated as

$$\frac{1}{\xi'(x)} = \prod_{k=0}^{m-1} \left( \frac{\pi}{2} \cos\left( \frac{\pi}{2} g_k(\xi) \right) \right), \quad m \geqslant 1, \tag{7}$$

whereas the quantity $H_m = \xi''(x)/[\xi'(x)]^2$ can be computed using the recursion

$$\begin{aligned} H_0 &= 0, \\ H_k &= \frac{\pi}{2} \tan\left( \frac{\pi}{2} g_{m-k}^{(-1)}(x) \right) + \frac{\pi}{2} \cos\left( \frac{\pi}{2} g_{m-k}^{(-1)}(x) \right) H_{k-1}, \quad k = 1, \ldots, m. \end{aligned} \tag{8}$$

Here, $g_m^{(-1)}$ denotes the inverse function of $g_m$, i.e.,

$$g_0^{(-1)}(x) = x, \quad g_m^{(-1)}(x) = \frac{2}{\pi} \arcsin(g_{m-1}^{(-1)}(x)).$$

The interested reader is referred to [19] and [27] for details.

We note that the SINE-transforms (3) also map the intervals $[-1, 0]$ and $[0, 1]$ onto themselves. Hence, as suggested in [19], if (2) has only one layer on the left (or right), we translate the physical domain to $[-1, 0]$ (or $[0, 1]$, respectively). This avoids unnecessary collocation points in the smooth region. In Fig. 2, we show a solution of a BVP (our Example 1 in Section 4) in both the computational domain and the physical domain. Some of the data points in the physical domain coincide numerically near the boundaries. This does not cause difficulties, as there is no coincidence in the computational domain.

In our MQ integral formulation, the second derivative $v''(\xi)$ in (5) is approximated by the MQ basis. The expansion of the unknown function $v(\xi)$ is then of the form

$$v(\xi) = \sum_{j=1}^{N} \lambda_j \, \Phi_j(\xi) + \gamma_1 + \gamma_2 \, \xi, \tag{9}$$

where the last two terms are the polynomials resulting form integrations, and

$$\Phi_j(\xi) = \frac{1}{6}\phi_j(\xi)(\xi-\xi_j) + \frac{c_j^2}{2}[(\xi-\xi_j)\ln(\phi_j(\xi) - (\xi-\xi_j)) - \phi_j(\xi)],$$

$$\Phi'_j(\xi) = \frac{1}{2}(\xi-\xi_j)\phi_j(\xi) + \frac{c_j^2}{2}\ln(\phi_j(\xi) + (\xi-\xi_j)),$$

$$\Phi''_j(\xi) = \phi_j(\xi) := \sqrt{(\xi-\xi_j)^2 + c_j^2}.$$

The unknown coefficients in the numerical approximation (9) are determined by making (9) satisfy the transformed BVP (5) at all collocation points. Collocating the transformed BVP (5) at all $N$ data centres (knots) $\xi_j$, for $j = 1, \ldots, N$, provides $N$ linear equations:

$$\sum_{i=1}^{N} \lambda_i [\varepsilon \, \Phi''_i(\xi_j) + P(\xi_j)\, \Phi'_i(\xi_j) + Q(\xi_j)\, \Phi_i(\xi_j)] \\ + \gamma_1 Q(\xi_j) + \gamma_2 [P(\xi_j) + \xi_j \, Q(\xi_j)] = F(\xi_j), \tag{10}$$

for $1 \leqslant j \leqslant N$. The remaining 2 equations are obtained by substituting (9) into the boundary conditions in (5). This yields

$$\sum_{i=1}^{N} \lambda_i \Phi_i(a) + \gamma_1 + \gamma_2 \, a = \alpha,$$

$$\sum_{i=1}^{N} \lambda_i \Phi_i(b) + \gamma_1 + \gamma_2 \, b = \beta. \tag{11}$$

The $N + 2$ linear equations (10) and (11) determine all unknown coefficients in (9).

The numerical results in [19] show this method to be very effective. The PSC method of [27] has superior performance for “large” values of $\varepsilon$, but the MQ method (with constant $c = 0.815/(N-1)$ as suggested by Hardy [13]) produces better results for very small values of $\varepsilon$ (i.e., $\varepsilon = 10^{-8}$ and smaller). The MQ scheme provides more flexibility on centres placement, as it is based on a meshless method. Although the size of $\varepsilon$ provides a good basis for choosing a good value of $m$ (the number of sine transformations), there is no obvious way to choose the optimal $m$ a priori. With the optimal $m$, the MQ scheme is very accurate; it produces more accurate results than a recently proposed adaptive multiquadric scheme [14]. The new adaptive scheme proposed below has the advantage that high accuracy can be achieved even if one does not have the “best” $m$.

## 3. Adaptive scheme

Hon [14] proposes an adaptive MQ-RBF technique using an "a posteriori" indicator based on the weak formulation of the governing equation to add collocation points where deemed necessary. Hon's indicator $I_i$ is given by

$$I_i = \int_{x_{i-1}}^{x_{i+1}} [\varepsilon u'' + p(x)u' + q(x)u - f] w_i \, \mathrm{d}x, \tag{12}$$

for a BVP of the form (2), where $u$ is the current numerical approximation to the solution, and

$$w_i = w_i(x) = \begin{cases} \dfrac{x - x_{i-1}}{x_i - x_{i-1}} & \forall x \in (x_{i-1}, x_i), \\ \dfrac{x_{i+1} - x}{x_{i+1} - x_i} & \forall x \in (x_i, x_{i+1}). \end{cases}$$

An extra data point is added at both midpoints of the intervals $[x_{i-1}, x_i]$ and $[x_i, x_{i+1}]$ if $|I_i - I_\mu| \geqslant \theta I_\sigma$, where $I_\mu$ and $I_\sigma$ denote the mean and the standard deviation of the indicators $I_i$, and $\theta > 0$. We refer to the original article [14] for details. As shown in [19], Hon's indicator is not suitable for the transformed BVP (5). In this section, we propose a new indicator that appears to work well with our scheme.

We want to use our indicator to add points in the regions of the domain where the indicator is "large"; these regions should correspond to the regions where the (unknown) error in our solution is also large. To design such an indicator, we look at two main sources of error. When the solution of the transformed BVP (5) still changes rapidly in the region corresponding to the boundary layer of the original BVP (2), the error is usually large and more data points are needed to capture the solution, see Fig. 3 where Example 1 in Section 4 is solved with three SINE-transforms ($m = 3$) and 128 points. On the other hand, when the number of collocation points used in the approximation (9) is small on some region, it is common to observe oscillations, and therefore a large error, see Fig. 4 where the same BVP is solved with $N = 32$ points only. In all figures in this paper we plot dimply the function values at the data centres, not the RBF interpolant. We want to emphasize that although the curves in some figures may look piecewise linear (when the number of data points is small or when the functions are oscillatory), all RBF interpolants are smooth (i.e., infinitely differentiable).

In both cases, as shown in Figs. 3 and 4, we expect a close relation between the error and the second derivative values of the computed solution: rapid changes and oscillations result in a large value of the second derivative. Since all the derivatives of the MQ basis are globally defined, we define an indicator using the second derivative value evaluated at the midpoint of two nearby data points:

$$I_i^{(1)} = \frac{\mathrm{d}^2 v}{\mathrm{d}\xi^2}\left(\frac{\xi_i + \xi_{i-1}}{2}\right), \quad i = 1, \ldots, N. \tag{13}$$

One problem with this indicator is that the sum of $I_i^{(1)}$ will grow without bound as $N$ increases. A bounded version of (13) mimics the integral of the third derivative of the numerical solution, and retains some information on the grid spacing:

$$I_i^{(2)} = (\xi_i - \xi_{i-1})\frac{\mathrm{d}^3 v}{\mathrm{d}\xi^3}\left(\frac{\xi_i + \xi_{i-1}}{2}\right), \quad i = 1, \ldots, N. \tag{14}$$

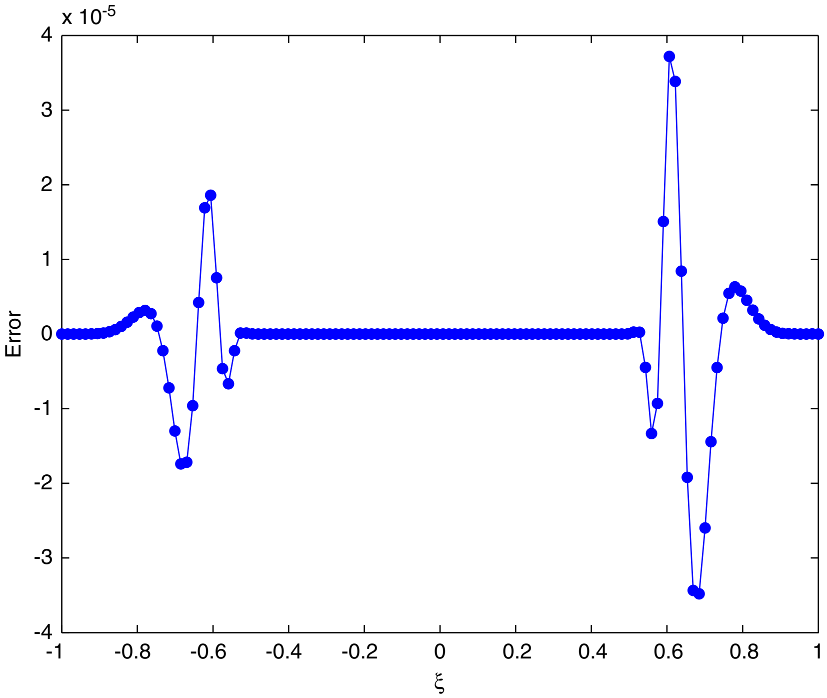

Fig. 3. The computed solution changes rapidly in the boundary layer: the error in the layer is large in comparison to the error on the smooth region ($m = 3$, $N = 128$).

We employ the indicator $I^{(2)}$ to solve Example 1 with $\varepsilon = 10^{-12}$. Starting with $N = 40$ points initially and after six iterations, we run into a situation as shown in Fig. 5. From the plot of the error, we see that past iterations added most of the points to the boundary layer around $\xi = \pm 0.7$, and the region in between. The new indicator $I^{(2)}$ is successful in adding points into the region with the largest error, namely the boundary layer. However, only very few data points are located near the boundary of the interval $[-1, 1]$. The undersampling near the boundary causes the numerical solution to oscillate, and therefore adversely affects the accuracy of the scheme. Proceeding with the adaptive iteration further with $I^{(2)}$, as shown in the plot of the indicator in Fig. 5 (bottom), all new points will eventually be added near the boundary layer. Due to its inability to refine data points in the smooth region, an adaptive iteration using $I^{(2)}$ works well when one starts with a sufficiently large number of data points, and only lets the adaptive scheme run for a small number of iterations so that the ratio of maximum and minimum centre spacing remains small.

A robust adaptive scheme, however, should also have some control over the ratio between the maximum and minimum centre spacing. Hence, we modify (14) to obtain our new indicator

$$J_i = (\xi_i - \xi_{i-1})^{\rho(k)} \frac{\mathrm{d}^3 v}{\mathrm{d}\xi^3}\left(\frac{\xi_i + \xi_{i-1}}{2}\right), \quad i = 1, \ldots, N, \tag{15}$$

where $k$ is the iteration counter for our adaptive scheme, and $\rho(\cdot)$ is any non-decreasing function.

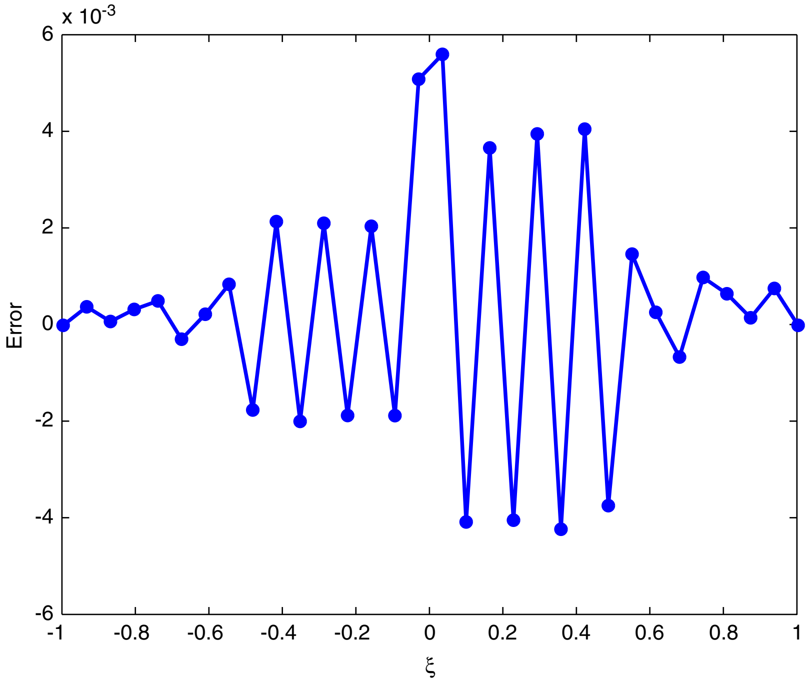

Fig. 4. When collocation points are not dense enough, the numerical solution oscillates in the smooth region ($m = 3$, $N = 32$).

The indicator $I^{(2)}$ is a special case of (15) with $\rho(k) \equiv 1$. For any constant $\rho(k) > 1$, a new balance or equilibrium between grid spacing and solution derivative will be achieved. Note that the value of each $J_i$ by itself is not important to the scheme. It is the relative magnitude of $J_i$ that controls the addition of new data points. The interesting case is when $\rho(k)$ increases with $k$. The indicator $J_i$ behaves very much like $I_i^{(2)}$ when $k$ is small. The problem shown in Fig. 5, however, will be corrected as $k$ gets large. Suppose $\xi_i - \xi_{i-1}$ is large in comparison to the other separation distances. The term $(\xi_i - \xi_{i-1})^{\rho(k)}$ assigns a relatively large value to $J_i$, so the adaptive scheme will add an extra point between $\xi_{i-1}$ and $\xi_i$. The adaptive scheme eventually reduces the large grid spacing. In general, a rapidly growing $\rho(k)$ will favour equally distributed grids, whereas a slowly increasing $\rho(k)$ will tend to add more points to the boundary layer. All our computations in the next section use the indicator $J_i$ with $\rho(k) = k$.

## 4. Numerical results

**Example 1.** Boundary layer problem with very thin layer.

We first demonstrate the robustness of our adaptive scheme with the example found in [27]. The BVP with variable coefficients is given by

$$\varepsilon u''(x) - xu'(x) - u(x) = f(x), \quad x \in [-1, 1], \tag{16}$$

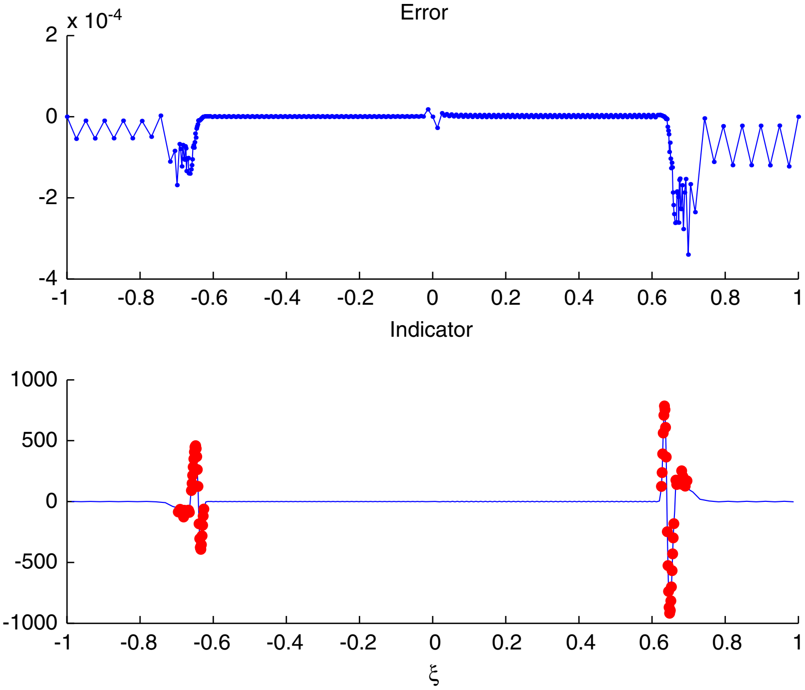

Fig. 5. The error function and the indicator function of an adaptive scheme using $I^{(2)}$ after a few iterations. The big bullets shown in the plots of the indicator are points that will be added in the next iteration.

where

$$f(x) = \left(\frac{x+1}{\varepsilon} - 1\right) \mathrm{e}^{-(x+1)/\varepsilon} - 2\left(\frac{x-1}{\varepsilon} + 1\right) \mathrm{e}^{(x-1)/\varepsilon}$$

is chosen such that the function

$$u(x) = \mathrm{e}^{-(x+1)/\varepsilon} + 2\,\mathrm{e}^{(x-1)/\varepsilon}$$

is an exact solution of the BVP. Boundary conditions are given by the values of the exact solution. This is not an easy test problem; besides exhibiting two boundary layers, it also features a turning point at $x=0$.

We use the indicator $J$ with $\rho(k)=k$ to solve the BVP with $\varepsilon=10^{-12}$. The number $m$ always refers to the number of SINE-transforms applied to obtain (5). The MQ shape parameter is set to $c_j = 0.815h_j$, where $h_j$ is the minimum distance between the centre $\xi_j$ and its nearby centres. An extra point $(\xi_j + \xi_{j+1})/2$ will be added to the set of data points if $|J_j - J_\mu| \geqslant \theta J_\sigma$, where $J_\mu$ and $J_\sigma$ denote the mean and the standard deviation of these indicators $J_j$. In our tests, we use $\theta = 0.5$. The adaptive iteration starts with $N = 40$ data points and stops when the $\ell_2$-norm difference of two consecutive solutions $\|v_k - v_{k-1}\|_2$, that is computed on the physical domain with trapezoidal rule evaluating at all data centres of $v_k$, is less than $\delta_a = 10^{-4}$. Note that in [19], the authors use $\ell_\infty$-norm, which is identical on both the computational and physical domain, to measure accuracy. To monitor convergence, however, we employ the $\ell_2$-norm for better stability. On the other hand, errors in both norms are reported in this section.

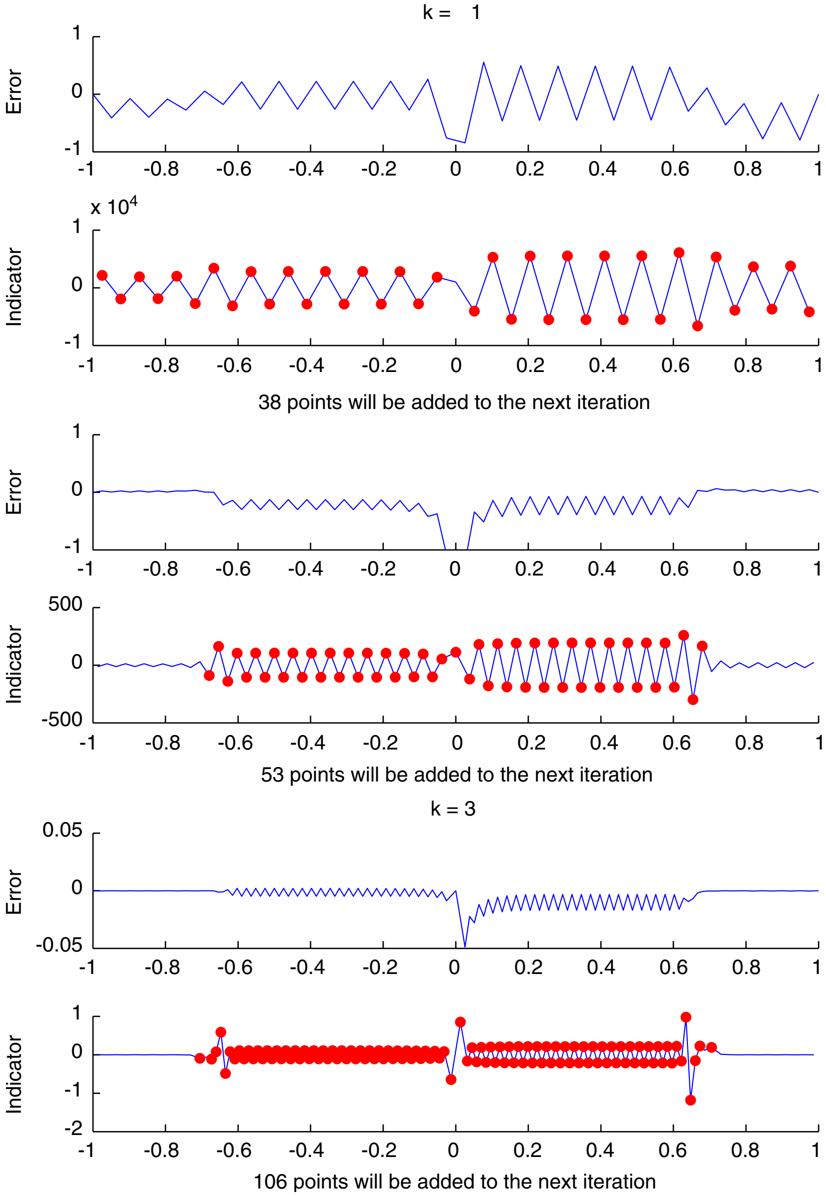

Fig. 6. Errors and indicators $J$ for boundary layer problem (16) with $\rho(k)=k$, $\varepsilon=10^{-12}$ and $m=5$ after 1–3 adaptive iterations.

In Figs. 6 and 7 the exact error function and the indicator for all adaptive iterations with $m=5$ are provided. The big bullets ( ● ) shown in the plots of the indicator are points that will be added in the next adaptive iteration. Initially, at $k=1$, the number of data points, $N=40$, is too small for the boundary layer problem with such a thin layer. Our scheme refines the set of data points throughout the whole computational domain. For $k=2$ and $k=3$, extra points are added only to the interior. At the fourth adaptive iteration, extra data points are added in the boundary layers, because of the large value of the third derivative, and near the origin, where relatively wide spacing of the data centres is causing an oscillation.

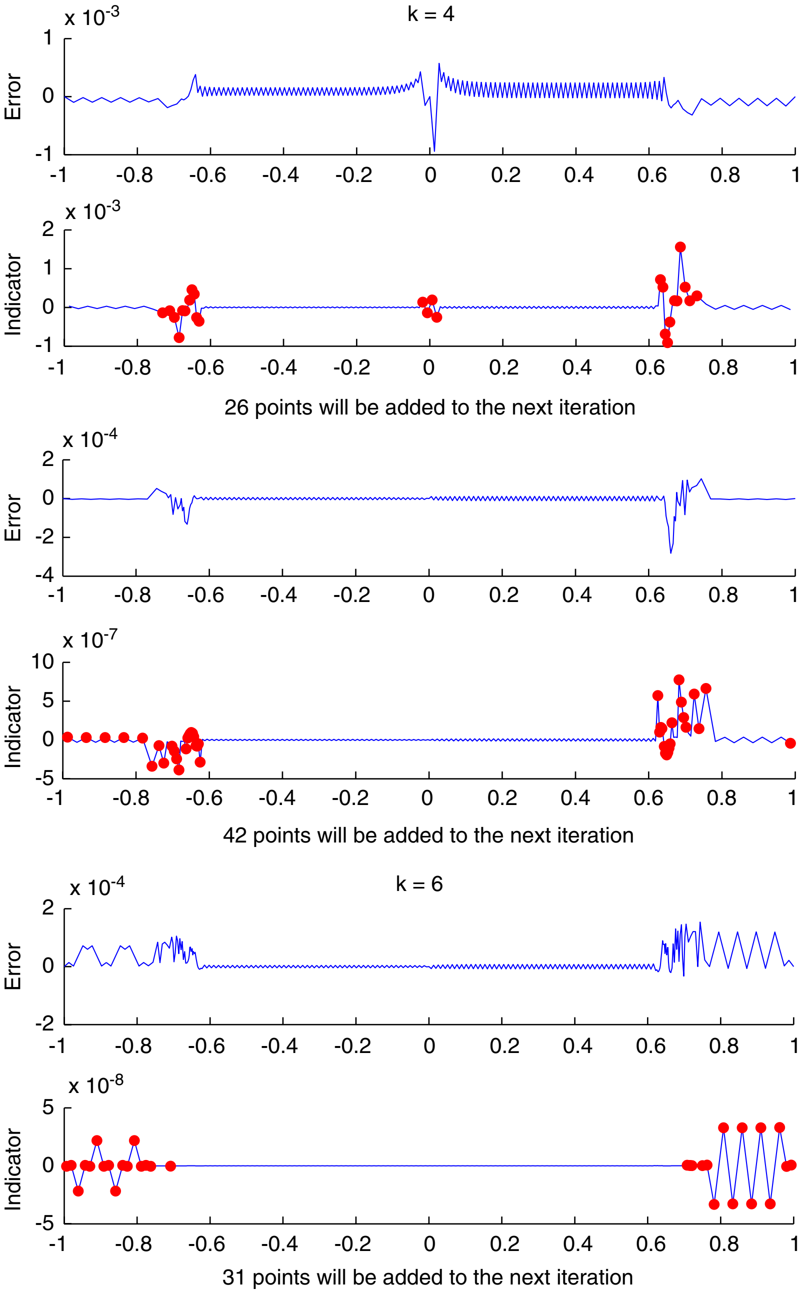

Fig. 7. Errors and indicators $J$ for boundary layer problem (16) with $\rho(k)=k$, $\varepsilon=10^{-12}$ and $m=5$ after 4–6 adaptive iterations.

The different behaviour of $I^{(2)}$ and $J$ is obvious at the last two adaptive steps. Comparing to Fig. 5, $J$ will add points near the boundaries where the error is large. In all cases, our indicator correctly indicates the regions of large error. A comparison of the non-adaptive method with $N=512$ and the adaptive method is given in Table 1. Details of the adaptive iterations are shown in Table 2.

Table 1
Errors for boundary layer problem (16) with $\varepsilon = 10^{-12}$

| $m$ | Error: non-adaptive | | | Error: adaptive | | |
|---|---|---|---|---|---|---|
| | $N$ | $\ell_\infty$-norm | $\ell_2$-norm | $N$ | $\ell_\infty$-norm | $\ell_2$-norm |
| 4 | 512 | $8.14e-04$ | $8.85e-07$ | 407 | $2.00e-04$ | $1.15e-05$ |
| 5 | 512 | $6.09e-04$ | $3.05e-05$ | 305 | $1.54e-04$ | $7.62e-06$ |
| 6 | 512 | $1.56e-03$ | $9.30e-05$ | 257 | $2.24e-04$ | $2.19e-06$ |
| 7 | 512 | $2.30e-03$ | $6.30e-04$ | 430 | $1.84e-04$ | $8.12e-05$ |

Comparison of the non-adaptive method with $N = 512$ and the adaptive method.

Table 2
Performance of the adaptive scheme on an exponentially ill-conditioned problem (17)

| $\varepsilon$<br>$m$ | $1/70$<br>0 | $1/70$<br>1 | $1/70$<br>2 | $10^{-4}$<br>2 | $10^{-4}$<br>3 | $10^{-8}$<br>3 | $10^{-8}$<br>4 |
|---|---|---|---|---|---|---|---|
| # iterations | 8 | 4 | 4 | 7 | 6 | 7 | 7 |
| $N$ | 71 | 33 | 39 | 179 | 95 | 313 | 155 |
| Final error | 3.9(−3) | 7.8(−3) | 4.6(−3) | 2.7(−4) | 7.5(−4) | 6.6(−3) | 6.1(−2) |

**Example 2.** Exponentially ill-conditioned problem.

For our next example, we consider the equation found in [11],

$$\varepsilon u''(x) - xu'(x) + u(x) = 0, \quad x \in [-1, 1], \tag{17}$$

with

$$u(-1) = 1, \quad u(1) = 2,$$

whose exact solution is

$$u(x) = \frac{1}{2}x + \frac{3}{2}\frac{\sqrt{-\frac{x^2\pi}{2\varepsilon}}\,\mathrm{erf}\left(-\frac{x^2}{2\varepsilon}\right) + \mathrm{e}^{(x^2/2\varepsilon)}}{\sqrt{-\frac{\pi}{2\varepsilon}}\,\mathrm{erf}\left(-\frac{1}{2\varepsilon}\right) + \mathrm{e}^{(1/2\varepsilon)}}.$$

This problem contains an eigenvalue of the order $\mathrm{e}^{-1/2\varepsilon}$, and is exponentially ill-conditioned. An analysis of this problem can be found in [10]. In [18], the problem is solved for $\varepsilon = 1/70$ with an adaptive method; the final error obtained is $2.2 \times 10^{-2}$ under the $\ell_2$-norm. It appears that the method of [10] cannot handle smaller values of $\varepsilon$.

A consequence of the ill-conditioning of this problem is that one may observe numerical divergence. We implement an extra stopping criterion which terminates the iteration and discards the newest numerical solution if divergence is detected.

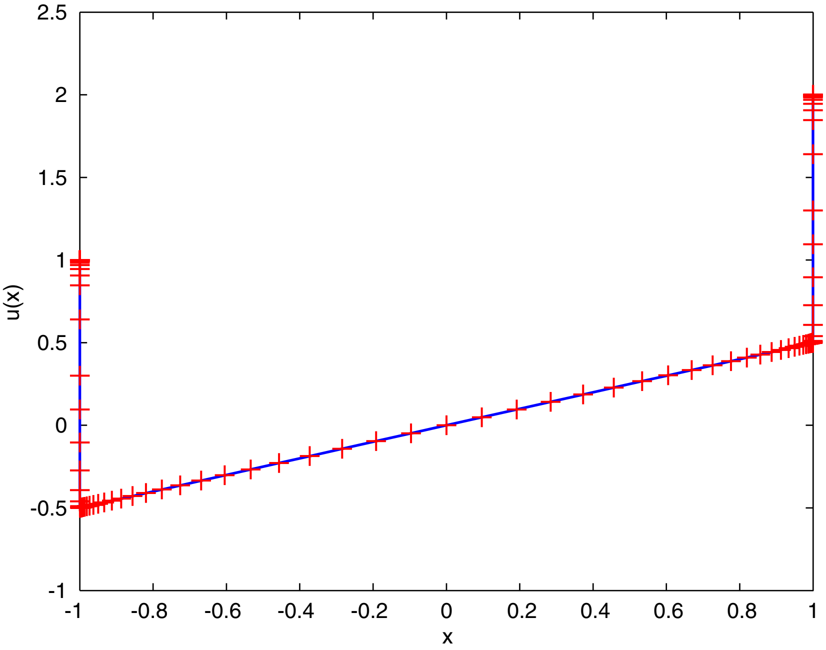

Fig. 8. Numerical solution (+) and exact solution (–) of (17) with $m = 3$ and $\varepsilon = 10^{-4}$ in the physical domain.

We solve the exponentially ill-conditioned problem (17) with our adaptive scheme for $\varepsilon = 1/70$, $\varepsilon = 10^{-4}$, and $\varepsilon = 10^{-8}$. The scheme begins with 11 equally spaced points, and $\theta = 0.5$. For comparison, we also list the final error in the $\ell_2$-norm. Plots of the numerical solution for $\varepsilon = 10^{-4}$ and $m = 3$ against the physical variable $x$, and the computational variable $\xi$ are shown in Figs. 8 and 9. Fig. 9 reveals the difficulty coming from the interior transition region. We summarize our results in Table 2. Our adaptive scheme shows clear advantage over the method in [10] on this ill-conditioned problem in terms of accuracy and the ability to solve problems with very small $\varepsilon$. On the other hand, we emphasize that the method found in [10] applies to a broader range of problems because of its domain decomposition feature. The adaptive method in [10] can even accurately capture the solution behaviour in regions where the exact solution is highly oscillatory.

**Example 3.** Interior layer problem.

The SINE-transforms are designed to work specifically on boundary layer problems, see Appendix. Hence, our method is not a natural candidate for solving problems with interior layers. To demonstrate the robustness of our new adaptive scheme, we apply it to an interior layer problem without using any coordinate transform. We solve the example in [14] where the coefficient functions of (2) are given by

$$p(x) = 4(x - 0.5)(1 + 0.3121(x - 1.5)),$$

and

$$q(x) = -1 - 0.2764(x - 0.5).$$

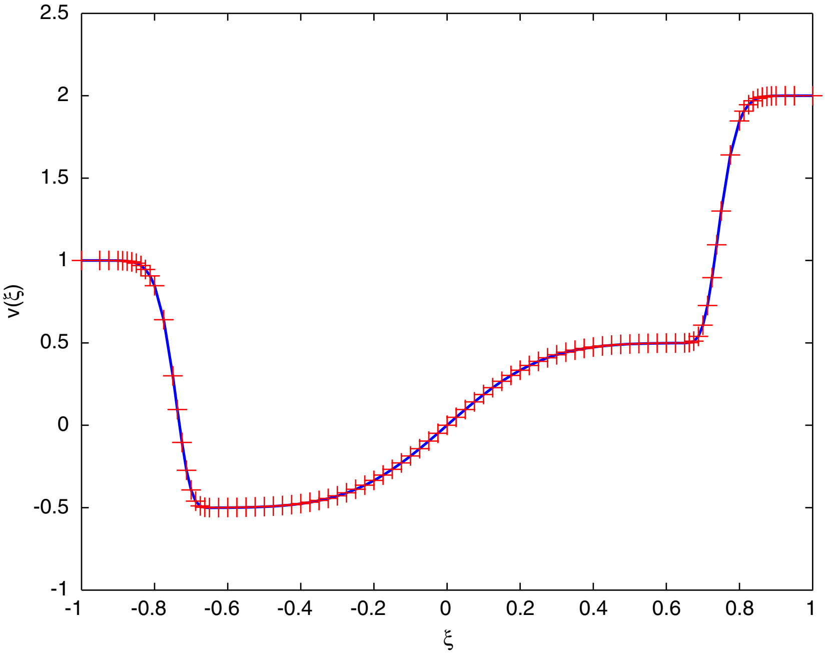

Fig. 9. Numerical solution (+) and exact solution (–) of (17) with $m = 3$ and $\varepsilon = 10^{-4}$ in the computational domain.

The right-hand side function $f(x)$ in (2) is chosen such that

$$u(x) = \frac{\left[\sqrt{0.291(x-0.5)^2+\varepsilon} + (x-0.5)\right]}{\left(0.291(x-0.5)^2+\varepsilon\right)^{0.375}} + \mathrm{e}^{-x^2/2}$$

is the exact solution. Without the help of coordinate stretching, our scheme is not capable of dealing with extremely small layers; here, we solve this internal layer problem with $\varepsilon = 10^{-4}$. No SINE-transform is applied, i.e., $m = 0$. We start our adaptive scheme with 21 data points and iteratively refine it to 103 data points after 6 iterations for the numerical solutions, see Fig. 10. The corresponding errors in the $\ell_\infty$-norm and $\ell_2$-norm are $5.84 \times 10^{-5}$ and $2.85 \times 10^{-5}$, respectively. Without the use of an appropriate transform it appears difficult to solve this problem with much smaller perturbation parameter $\varepsilon$.

## 5. Conclusion

We develop an adaptive scheme based on a multiquadric collocation method with a variable transform and integral formulation to solve boundary layer problems. Taking advantage of the meshless nature of RBF methods we easily add extra data points where necessary. The differentiability of the MQ basis allows us to design a suitable indicator using the derivative values of the numerical solution. The adaptive scheme makes the tuning of the parameter $m$ (which controls the variable transformation) unnecessary, and therefore increases the robustness of our solution scheme. Our adaptive scheme appears to work

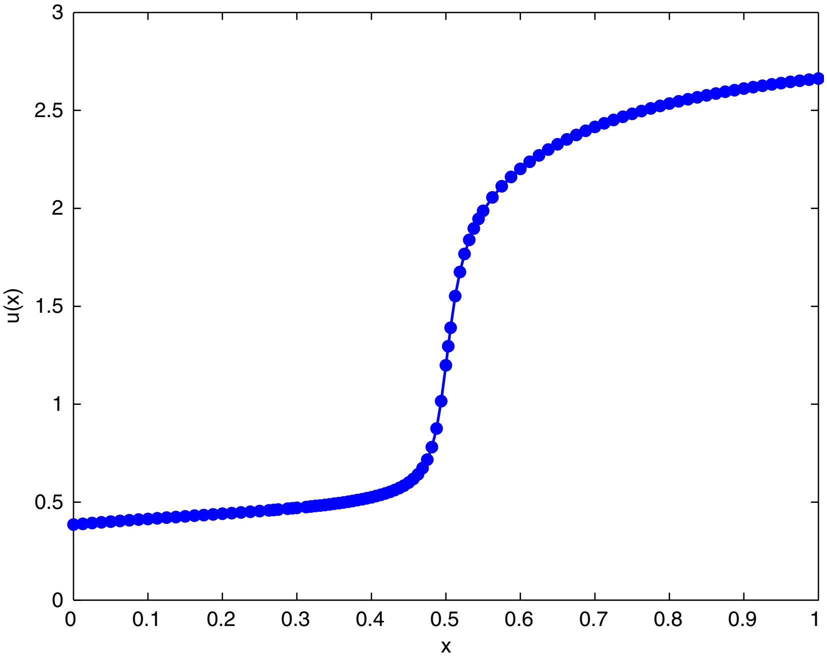

Fig. 10. The solution of an internal layer problem with 103 refined data points.

extremely well for small perturbation parameters $\varepsilon$, and in situations where the exact solution is not highly oscillatory.

We give an example of solving an interior layer problem with our RBF scheme; however, our method is not designed for such a problem. Like many of the existing schemes, we see that our scheme fails to capture very thin interior layers. One of our goals is to develop a new coordinate transform that allows us to handle interior layers as efficiently as boundary layers. On the other hand, finding the exact location of the interior layer may not be trivial. Quite a sophisticated adaptive scheme may be required to capture an interior layer.

In this paper we only presented linear examples, and demonstrate our results to those in [11,14,27]. Reports on the method in [27] indicate that the accuracy attained is comparable to the results with COLSYS. We are currently implementing a nonlinear variant of our method, and shall report on its performance relative to COLSYS in a future paper. The present work validates the RBF approach we have taken, and demonstrates the usefulness of adding adaptivity to the method.

The authors are working on extending the preliminary adaptive scheme presented in this paper to two dimensions. In particular, extending the indicator, which uses information about the third derivative of the numerical approximation, is not straightforward.

## Appendix

The boundary layer problems are specifically given in Table 3.

Table 3
Detailed data of the $k$ adaptive iteration for boundary layer problem (16) $\varepsilon = 10^{-12}$ with different $m$

| $m = 4/k$ | $N$ | Error in $\ell_\infty$-norm | Error in $\ell_2$-norm | $\|v_k - v_{k-1}\|_2$ |
|---|---|---|---|---|
| 1 | 40 | $1.03e+00$ | $6.07e-01$ | — |
| 2 | 78 | $1.68e+00$ | $1.18e+00$ | $1.01e+00$ |
| 3 | 154 | $1.40e-01$ | $7.81e-02$ | $1.37e+00$ |
| 4 | 287 | $3.47e-03$ | $1.57e-03$ | $8.20e-02$ |
| 5 | 366 | $2.50e-04$ | $1.44e-05$ | $1.34e-03$ |
| 6 | 407 | $2.00e-04$ | $1.15e-05$ | $8.09e-06$ |
| $m = 5/k$ | | | | |
| 1 | 40 | $8.43e-01$ | $8.70e-01$ | — |
| 2 | 78 | $1.22e+00$ | $1.04e+00$ | $6.98e-01$ |
| 3 | 131 | $4.88e-02$ | $2.62e-02$ | $1.20e+00$ |
| 4 | 237 | $9.42e-04$ | $4.66e-04$ | $2.66e-02$ |
| 5 | 263 | $2.82e-04$ | $1.12e-05$ | $4.11e-04$ |
| 6 | 305 | $1.54e-04$ | $7.62e-06$ | $3.63e-06$ |
| $m = 6/k$ | | | | |
| 1 | 40 | $1.22e+00$ | $1.38e+00$ | — |
| 2 | 78 | $6.60e+00$ | $6.88e+00$ | $6.88e+00$ |
| 3 | 115 | $2.51e-01$ | $1.56e-01$ | $8.06e+00$ |
| 4 | 185 | $2.78e-02$ | $8.32e-03$ | $1.58e-01$ |
| 5 | 206 | $4.38e-04$ | $3.04e-05$ | $7.44e-03$ |
| 6 | 257 | $2.24e-04$ | $2.19e-06$ | $1.32e-05$ |
| $m = 7/k$ | | | | |
| 1 | 40 | $2.86e+01$ | $3.54e+01$ | — |
| 2 | 78 | $6.78e+00$ | $7.55e+00$ | $3.00e+01$ |
| 3 | 105 | $6.84e-02$ | $1.23e-02$ | $8.97e+00$ |
| 4 | 161 | $5.40e-01$ | $3.82e-01$ | $3.67e-01$ |
| 5 | 202 | $9.75e-02$ | $6.68e-02$ | $4.03e-01$ |
| 6 | 252 | $7.38e-03$ | $6.22e-04$ | $6.61e-02$ |
| 7 | 302 | $1.69e-04$ | $9.11e-05$ | $5.81e-04$ |
| 8 | 430 | $1.84e-04$ | $8.12e-05$ | $7.69e-05$ |

## References

[1] U. Ascher, J. Christiansen, R.D. Russell, A collocation solver for mixed order systems of boundary value problems, Math. Comp. 33 (146) (1979) 659–679.

[2] U.M. Ascher, R.M.M. Mattheij, R.D. Russell, Numerical solution of boundary value problems for ordinary differential equations, Classics in Applied Mathematics, vol. 13, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1995.

[3] I. Babuška, J.M. Melenk, The partition of unity method, Internat. J. Numer. Methods Engrg. 40 (4) (1997) 727–758.

[4] T. Belytschko, Y. Lu, L. Gu, Element-free galerkin methods, Internat. J. Numer. Methods Engrg. 37 (1994) 229–256.

[5] C. Canuto, M.Y. Hussaini, A. Quarteroni, T.A. Zang, Spectral methods in fluid dynamics, Springer Series in Computational Physics, Springer, New York, 1988.

[6] C. Duarte, J. Oden, Hp clouds: a h–p meshless method, Numer. Methods PDEs 12 (1996) 673–705.

[7] A.I. Fedoseyev, M.J. Friedman, E.J. Kansa, Improved multiquadric method for elliptic partial differential equations via PDE collocation on the boundary, Comput. Math. Appl. 43 (3–5) (2002) 439–455 (Radial basis functions and partial differential equations).

[8] B. Fornberg, T.A. Driscoll, A fast spectral algorithm for nonlinear wave equations with linear dispersion, J. Comput. Phys. 155 (2) (1999) 456–467.

[9] R. Franke, Scattered data interpolation: tests of some methods, Math. Comp. 38 (157) (1982) 181–200.

[10] E.C. Gartland Jr., Graded-mesh difference schemes for singularly perturbed two-point boundary value problems, Math. Comp. 51 (184) (1988) 631–657.

[11] L. Greengard, V. Rokhlin, A fast algorithm for particle simulations, J. Comput. Phys. 73 (1987) 325–348.

[12] R. Hardy, Multiquadric equations of topography and other irregular surfaces, J. Geophys. Res. 176 (1971) 1905–1915.

[13] R.L. Hardy, Theory and applications of the multiquadric-biharmonic method 20 years of discovery 1968–1988, Comput. Math. Appl. 19 (8–9) (1990) 163–208.

[14] Y.C. Hon, Multiquadric collocation method with adaptive technique for problems with boundary layer, Internat. J. Appl. Sci. Comput. 6 (3) (1999) 173–184.

[15] E.J. Kansa, Multiquadrics—a scattered data approximation scheme with applications to computational fluid-dynamics. I. Surface approximations and partial derivative estimates, Comput. Math. Appl. 19 (8–9) (1990) 127–145.

[16] E.J. Kansa, Multiquadrics—a scattered data approximation scheme with applications to computational fluid-dynamics. II. Solutions to parabolic hyperbolic and elliptic partial differential equations, Comput. Math. Appl. 19 (8–9) (1990) 147–161.

[17] E.J. Kansa, H. Power, G.E. Fasshauer, L. Ling, A volumetric integral radial basis function method for time dependent partial differential equations, Eng. Anal. Bound. Elem. 28 (10) (2004) 1191–1206.

[18] J.-Y. Lee, L. Greengard, A fast adaptive numerical method for stiff two-point boundary value problems, SIAM J. Sci. Comput. 18 (2) (1997) 403–429.

[19] L. Ling, M.R. Trummer, Multiquadric collocation method with integral formulation for boundary layer problems, Comput. Math. Appl. 48 (5–6) (2004) 927–941.

[20] W.R. Madych, S.A. Nelson, Multivariate interpolation and conditionally positive definite functions, Approx. Theory Appl. 4 (4) (1988) 77–89.

[21] W.R. Madych, S.A. Nelson, Multivariate interpolation and conditionally positive definite functions II, Math. Comp. 54 (189) (1990) 211–230.

[22] N. Mai-Duy, T. Tran-Cong, Numerical solution of differential equations using multiquadric radial basis function networks, Neural Networks 14 (2001) 185–199.

[23] N. Mai-Duy, T. Tran-Cong, Numerical solution of navier–stokes equations using multiquadric radial basis function networks, Internat. J. Numer. Methods Fluid 37 (2001) 65–86.

[24] J.J.H. Miller, E. O'Riordan, G.I. Shishkin, On piecewise-uniform meshes for upwind- and central-difference operators for solving singularly perturbed problems, IMA J. Numer. Anal. 15 (1) (1995) 89–99.

[25] V.D. Natoli, M.H. Cohen, B. Fornberg, A new numerical algorithm for the analytic continuation of Green's functions, J. Comput. Phys. 126 (1) (1996) 99–108.

[26] S.A. Sarra, Chebyshev super spectral viscosity solution of a two-dimensional fluidized-bed model, Internat. J. Numer. Methods Fluids 42 (3) (2003) 249–263.

[27] T. Tang, M.R. Trummer, Boundary layer resolving pseudospectral methods for singular perturbation problems, SIAM J. Sci. Comput. 17 (2) (1996) 430–438.

[28] R. Vulanović, On a numerical solution of a type of singularly perturbed boundary value problem by using a special discretization mesh, Univ. u Novom Sadu Zb. Rad. Prirod. Mat. Fak. Ser. Mat. 13 (1983) 187–201.

[29] Z.M. Wu, R. Schaback, Local error estimates for radial basis function interpolation of scattered data, IMA J. Numer. Anal. 13 (1) (1993) 13–27.

[30] J. Yoon, Spectral approximation orders of radial basis function interpolation on the Sobolev space, SIAM J. Math. Anal. 33 (4) (2001) 946–958 (electronic).