A minimal distance constrained adaptive mesh algorithm with application to the dual reciprocity method

Kamal Shanazari* and Ke Chen†
Department of Mathematical Sciences,
The University of Liverpool, Liverpool L69 7ZL, UK.

Abstract

We present a new technique for generating error equi-distributing meshes that satisfy both local quasi-uniformity and a preset minimal mesh spacing. This is firstly done in the one-dimensional case by extending the Kautsky and Nichols' method and then in the two-dimensional case by generalizing the tensor product methods to alternating curved line equi-distributions. With the new meshing approach, we have achieved better accuracy in approximation using interpolatory radial basis functions. Furthermore improved accuracy in numerical results has been obtained when the interpolatory strategy is applied to the dual reciprocity boundary element method for solving a class of linear and non-homogeneous partial differential equations.
Key words. Adaptive mesh, error equi-distribution, interpolation, RBF, DRBEM, numerical partial differential equations.
AMS subject classifications. 65N50, 65D05, 31B10, 65N38.

1 Introduction

Adaptive mesh algorithms have been widely used in the numerical solution of partial differential equations (PDEs) for boundary value problems by finite difference and finite element methods [1, 3, 5, 4, 12, 2, 15]. Adaptive meshes for the boundary element method (BEM) have also been considered for Poisson type equations; see [17, 19]. The purpose of adaptivity is to have the best accuracy with a minimum number of mesh points through error equi-distribution. The idea is to distribute the mesh points non-uniformly such that the solution error is uniform in the whole domain. This paper considers the use of adaptive meshes for the generalised boundary element method, the so-called dual reciprocity boundary element method (DRBEM), for numerical solution of linear and non-homogeneous PDEs.

There are three main issues associated with adaptive techniques: how to measure the unknown solution error (choice of monitor functions), how to implement the adaptive solution procedure (adaptive strategy) and how to control the mesh quality (mesh smoothness). This paper addresses the last issue. We shall present numerical results using the arc-length monitor and adopting the simple adaptive strategy of re-positioning the mesh points adaptively (the so-called r-method [18]). However our proposed idea will be applicable to other monitor functions and adaptive strategies [3, 5, 12, 7] whenever there is a need to control the minimal mesh spacing.

We note that there are other well-known adaptive techniques: the h, p and h-p methods [18]. In the h-method, refinement is carried out i.e. the number of mesh points is increased to improve

*E-mail: kamals@liverpool.ac.uk. Support of a studentship by the Ministry of Education (Iran) is gratefully acknowledged.
†E-mail: k.chen@liverpool.ac.uk, URL: http//www.liv.ac.uk/∼cmchenke (for correspondence).

the accuracy, in the p-method, a varying order of approximation is used i.e. the number of mesh points is kept fixed but the degree of the approximating function is increased to achieve the better accuracy and in the h-p method, both the number of points and the degree of the approximating functions vary [1].

The idea of DRBEM is to approximately reformulate a linear PDE into a boundary integral equation with algebraic constraints. The approximation comes from a radial basis function approximation in the interior of the domain and the algebraic constraints connect the interior and boundary values of the solution. Our motivation of introducing minimal distance control stems from a desire to improve the interpolation accuracy of radial basis functions for scattered data, where small mesh spacing contributes to ill-conditioning in the interpolation matrix. As the overall solution error by DRBEM depends on accuracy of both the radial basis function (RBF) approximation and the boundary integral approximation, our proposed idea of improving the RBF approximation is a simple way to improve DRBEM. Other ideas may include preconditioning techniques or the domain decomposition methods.

Recently an adaptive technique based on one dimensional (1D) ideas has been applied to improve the accuracy of DRBEM and some improved results have been observed [10]. This paper presents our further work pursuing this idea.

In Section 2, we briefly discuss the DRBEM and RBF interpolation. In Section 3, we propose a method to design a minimal distance constrained adaptive mesh in 1D and in Section 4 generalise it to two dimensions (2D). As mentioned, to a large extent, the methods of Sections 3 and 4 can be independently used for various adaptive mesh applications. In Section 5, we apply the methods of Section 4 to DRBEM for solving some PDEs in 2D and show that the use of minimal distance control improves the solution accuracy.

We remark that generalising a 1D adaptive mesh method to 2D is not trial as satisfying the equidistribution condition is a complicated matter. Using tensor products gives one solution [11, 10]. Our approach based on curved lines appears to be new and suitable for scattered data interpolation as only minimal mesh spacing rather than mesh skewness is a concern.

2 Dual reciprocity boundary element method

We describe the method for the following linear PDE,

$$\bigtriangledown^2(u) = b \qquad \text{in } \Omega, \tag{1}$$

with the non-homogeneous part

$$b = f(x, y, u, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}),$$

where u is the dependent variable in some 2D domain Ω. The boundary condition can be of the Dirichlet type ($u = \bar{u}$ on Γ_D) or Neumann type ($\frac{\partial u}{\partial n} = \bar{q}$ on Γ_N) or Rabin type (a combination of both), where $\Gamma_D + \Gamma_N = \Gamma$, is the boundary enclosing Ω. Applying the weighted residual technique and Green's theorem, the integral formulation for equation (1) yields

$$c_i u_i = \int_\Gamma (u\frac{\partial G}{\partial n} - G\frac{\partial u}{\partial n}) d\Gamma + \int_\Omega G b d\Omega, \tag{2}$$

where c_i are coefficients depending on the location of the point $p_i = (x_i, y_i)$ and weighting function G, called the fundamental solution, is given by

$$G = -\frac{1}{2\pi}\ln r,$$

where r is the distance between the source point and the field point. The above integral equation contains a domain integral corresponding to the non-homogeneous term, b. The main idea in DRBEM is converting this domain integral to an equivalent boundary integral by approximating the function b in terms of interpolating functions at a certain number of boundary N and internal L nodes, that is,

$$b = \sum_{k=1}^{N+L} \alpha_k \phi_k \quad \text{and} \quad \sum_{k=1}^{N+L} F_{jk} \alpha_k = b_j \tag{3}$$

where $\phi_k = \phi(r_k) = \phi(|p - p_k|)$ is the RBF (with $p = (x, y)$ and $p_k = (x_k, y_k)$) and α_k the corresponding coefficients with $F_{jk} = \phi(|p_j - p_k|)$. Note that the RBF interpolation matrix F is ill-conditioned as the matrix size becomes large and the minimal distance of nodes becomes smaller. Although there exist techniques to precondition the matrix, our idea here is to use adaptive meshes and control the minimal mesh size so that the overall matrix needs not be too large while the minimal nodal distance is not too small.

The essential step in DRBEM [21] is to express ϕ_k as the Laplacian of function $\hat{u}_k$ i.e.

$$\nabla^2 \hat{u}_k = \phi_k.$$

With ϕ_k chosen as a RBF, function $\hat{u}_k$ is another RBF and can be analytically found. Although there exist many choices for RBF [9], in this paper, we shall use $\phi_k = 1 + r_k^3$ and therefore $\hat{u}_k = \frac{r_k^2}{4} + \frac{r_k^5}{25}$.

With this approximation for b, the domain integral in equation (2) is converted to an equivalent boundary integral by applying the same idea as for the main equation, that is,

$$\int_\Omega G b d\Omega = \sum_{k=1}^{N+L} \alpha_k \int_\Omega (\nabla^2 \hat{u}_k) G d\Omega$$

$$= \sum_{k=1}^{N+L} \alpha_k \Big\{ \int_\Gamma (G \frac{\partial \hat{u}_k}{\partial n} - \hat{u}_k \frac{\partial G}{\partial n}) d\Gamma + c_i \hat{u}_{ik} \Big\}$$

where $\hat{u}_{ik}$ is the known value of $\hat{u}_k$ at the source point p_i. Thus the main equation (2) becomes a boundary integral equation

$$c_i u_i - \int_\Gamma (u \frac{\partial G}{\partial n} - G \frac{\partial u}{\partial n}) d\Gamma = \sum_{k=1}^{N+L} \alpha_k \Big\{ c_i \hat{u}_{ik} - \int_\Gamma (\hat{u}_k \frac{\partial G}{\partial n} - G \frac{\partial \hat{u}_k}{\partial n}) d\Gamma \Big\} \tag{4}$$

which is solved with RBF interpolation relation (3). Further details can be found in [21, 13, 14, 6, 22, 10]. In [13], we have shown that the overall solution error is consisted of the RBF approximation error from (3) and the boundary element error for (4); this paper will address how to reduce the RBF interpolation error through controlling the minimal mesh size of an adaptive mesh.

3 An adaptive mesh with minimal mesh size control

We present a method to produce an adaptive mesh with minimal mesh size control, suitable for RBF approximation. Given an existing mesh $t_0, t_1, \cdots, t_n$ with corresponding error values (at midpoints) $f_1, \cdots, f_n$, the padding values for a locally bounded mesh are generated by [7]

$$\hat{f}_j = g(t_j), \qquad g(t) = \max_{a \le \tau \le b} \frac{f(\tau)}{1 + \lambda \mid t - \tau \mid f(\tau)}, \tag{5}$$

where $\mathtt{k} = \exp(\lambda c)$ and $c = \int_a^b f(x)dx/n$. Note that when setting $\mathtt{k} = 1$, the locally bounded mesh becomes the uniform mesh so one usually takes $\mathtt{k} \approx 2$. Let $\hat{x}_0, \hat{x}_1, \cdots, \hat{x}_n$ be the equi-distribution mesh based on $\hat{f}_j$ i.e.

$$\int_a^{\hat{x}_j} \hat{f}(x)dx = jd = \frac{j}{n}\int_a^b \hat{f}(x)dx.$$

Here we assume that at least one interval $\hat{h}_j = \hat{x}_{j+1} - \hat{x}_j$ is smaller than some desirable size h_{min}. Note that resetting smaller intervals to h_{min} is an easy step but it is important to ensure that the new mesh is somehow as close to the old mesh as possible.

We now propose a method to yield new error values z_j such that

$$\int_a^b z(x)dx = \int_a^b \hat{f}(x)dx = nd, \tag{6}$$

and consequently to obtain a new mesh x_j in such a way that the resulting equi-distribution mesh based on z_j satisfies $h_j = x_{j+1} - x_j \geq h_{min}$ and equi-distributes the same error constant d

$$d = d(\hat{f}) = \frac{1}{n}\int_a^b \hat{f}dx = d(z) = \frac{1}{n}\int_a^b zdx = \frac{1}{n}\sum_{j=1}^{n} z_j\hat{h}_j. \tag{7}$$

Here we assume that $\hat{f}$ and z are piecewise constant functions generated by the middle point values. The requirement of (6) is to ensure that the new mesh is somehow as close to the old mesh as possible in the sense of conservation of the total error (or in other words, keeping the same equi-distribution constant d fixed as in (7)).

To obtain the new z_j, we first note that the following constant $\bar{f}$ determines which values of $\hat{f}_j$ require modifying

$$\int_{x_j}^{x_{j+1}} \hat{f}dx = \bar{f}\ h_{min} = d \qquad \text{and} \qquad \bar{f} = d/h_{min}. \tag{8}$$

Then to ensure $h_j \geq h_{min}$, we must set $z_j = \bar{f}$ whenever $\hat{f}_j \geq \bar{f}$. In doing so, denote by I_e the overall error integral reduction. The remaining task in constructing our method is how to ensure the same error constant d for a fixed scale of error values is maintained. We achieve this by setting (where $\hat{f}_j < \bar{f}$)

$$z_j = \hat{f}_j + w_j\frac{I_e}{\hat{h}_j} \tag{9}$$

such that $w_j \geq 0$ and $\sum_j w_j = 1$. To make sure that the new error values are closest to the old equi-distribution errors, we select the weight w_j according to the magnitude of $\hat{f}_j$.

In details, suppose that, among the current $\hat{f}_j$ values, there are m of $\hat{f}_j$'s larger than $\bar{f}$; denote these values by $\hat{f}_{k_j}$ for $j = 1, \cdots, m$. Denote the remaining values (less than $\bar{f}$) by $\hat{f}_{k_j}$ for $j = m+1, \cdots, n$. Here the sequence $k_1, \cdots, k_n$ represents a permutation of midpoints $1, 2, \cdots, n$. Then the overall error integral reduction is defined by

$$I_e = \int_a^b \left(\hat{f}(x) - \bar{f}\right)_+ dx = \sum_{j=1}^{m} \left(\hat{f}_{k_j} - \bar{f}\right)\hat{h}_{k_j} \tag{10}$$

where $(.)_+$ denotes the usual non-negative function and $\hat{h}_{k_j} = \hat{x}_{k_j} - \hat{x}_{k_j-1}$.

Therefore to use (9) to redistribute the integral reduction I_e, we propose the following iterative scheme to modify the error values. Starting by $z_{k_j}^{(1)} = \hat{f}_{k_j}$ and $m^{(1)} = m$, the iteration for $i = 1, 2, \cdots$ is performed until $z_{k_j}^{(i)} \le \bar{f}$ for any j.

$$z_{k_j}^{(i+1)} = \begin{cases} \bar{f}, & \text{for} \quad j = 1, \cdots, m^{(i)} \quad (\text{when} \quad z_{k_j}^{(i)} \ge \bar{f}), \\ z_{k_j}^{(i)} + w_j^{(i)} \dfrac{I_e^{(i)}}{\hat{h}_{k_j}}, & \text{for} \quad j = m^{(i)} + 1, \cdots, n \quad (\text{when} \quad z_{k_j}^{(i)} < \bar{f}), \end{cases} \tag{11}$$

where the weights are chosen as (note $\sum_{j=m^{(i)}+1}^{n} w_j^{(i)} = 1$)

$$w_j^{(i)} = \frac{z_{k_j}^{(i)}}{z_{k_{m^{(i)}+1}}^{(i)} + \cdots z_{k_n}^{(i)}} = \frac{z_{k_j}^{(i)}}{\sum_{j=m^{(i)}+1}^{n} z_{k_j}^{(i)}},$$

in order for the new error value to be close to the old value, and $I_e^{(i)}$ the overall error reduction in iteration i can be evaluated by replacing $\hat{f}$ by $z_{k_j}^{(i)}$ in (10) and $m^{(i)}$ is the number of error values which are less than or equal to $\bar{f}$ in iteration i. Although the procedure will be completed in at most n iterations we observed that only one or two iterations are usually sufficient. The ultimate values of the sequence $z^{(i)}$ will be the new error values z_j satisfying (6). This can be seen from

$$\int_a^b z^{(i+1)}(x)dx = \sum_{j=1}^{m^{(i)}} \bar{f}\hat{h}_{k_j} + \sum_{j=m^{(i)}+1}^{n} (z_{k_j}^{(i)} + w_j^{(i)} \frac{I_e^{(i)}}{\hat{h}_{k_j}})\hat{h}_{k_j}$$

$$= \sum_{j=1}^{m^{(i)}} z_{k_j}^{(i)}\hat{h}_{k_j} - I_e^{(i)} + \sum_{j=m^{(i)}+1}^{n} z_{k_j}^{(i)}\hat{h}_{k_j} + \sum_{j=m^{(i)}+1}^{n} w_j^{(i)} I_e^{(i)} = \sum_{j=1}^{n} z_{k_j}^{(i)}\hat{h}_{k_j} = \int_a^b z^{(i)}(x)dx.$$

Thus we proved our main result with regard to minimal mesh size control.

Theorem 1 *Given $h_{min} \le h = \frac{b-a}{n}$ and the new (non-negative) error function obtained from the iterative scheme (11), the new mesh $x_0, x_1, \ldots, x_n$ resulting from equi-distributing z_j will satisfy (i) the total error in error equi-distribution remains the same; (ii) $h_j = x_j - x_{j-1} \ge h_{min}$.*

We remark that practically we found $h_{min} \in [h^2, h/2]$ is adequate. Finally we summarise our adaptive method with minimal mesh size control as follows.

Algorithm 1 (Minimal mesh size control) *Given a non-uniform mesh $a = t_0, t_1, \cdots, t_n = b$ and the error values $f_1, \cdots, f_n$, a quasi-uniform parameter k and a pre-set minimal mesh size h_{min},*

1. *Generate $\hat{f}_1, \cdots, \hat{f}_n$ and then $\hat{x}_0, \hat{x}_1, \cdots, \hat{x}_n$, the locally bounded mesh using the Kautsky and Nichols' method [7] as in (5).*
2. *Obtain z_j of a new error function from iterating (11).*
3. *Generate the new equi-distribution mesh x_j based on z_j.*

Remark that the above equi-distribution in Step 3 can be optionally replaced by the Kautsky and Nichols' method [7] to give more smoothness in the resulting mesh. As a locally bounded mesh tries to adjust the small error values only, the minimal mesh size control is not affected.

4 Extension to two dimensions

As is known, generalisation of the 1D equi-distribution method to two dimensions is not straightforward, although there exist several techniques to satisfy the 2D equi-distribution condition as much as possible [4, 11, 20]. Our idea of minimal mesh size control can be applied to most of these techniques. However, in this work, we only compare two tensor products based methods. The first and simple dimension reduction method using error slabs is from [11] while the second and new technique to generate a 2D mesh is our preferred method for RBF approximation, which is introduced next.

In the context of RBF approximation, the orthogonality of numerical grids is not as crucial as in other traditional approximations using piecewise polynomials. Our new idea is, within the framework of tensor products, to allow error equi-distribution along curved lines instead of straight lines as with the first method [11]. Starting from a uniform mesh in some rectangular region Ω, $x_{jk} = x_j = jh_1, y_{jk} = y_k = kh_2$ for $k = 0, 1, \ldots, m$ and $j = 0, 1, \ldots, n$, we propose to implement this idea in three stages (with option of repeating stages $2-3$) to obtain the final mesh points (x_{jk}, y_{jk}). Here h_1 and h_2 are the respective steplengths. Denote by f_1 and f_2 the monitor functions for the x and y coordinate directions respectively.

In the first stage, the error equi-distribution is performed for each line in the horizontal direction, that is,

$$\int_{x_j}^{x_{j+1}} f_1(x, y_k)dx = constant \quad \text{for} \quad j = 0, 1, \cdots, n-1. \tag{12}$$

This yields the new mesh (x_{jk}, y_k) where x_{jk} is the x-coordinate of the jth points in the kth horizontal line. Note that $y_{jk} = y_k$ is not altered and only the x-coordinates of all mesh points change (see the top part of the right plot in Fig 1b).

In the second stage, the mesh is redistributed in the vertical direction along the new grid lines. As each vertical grid line is curved, we propose to implement the equi-distribution along the arc rather than the vertical coordinate. Denote by s the arc length variable for each vertical grid line j (for $j = 1, \cdots, n$). Then the distance s_k (for $k = 1, \ldots, m$) from (x_{j0}, y_{j0}) to (x_{jk}, y_{jk}) along the vertical grid line can be computed piecewise linearly by $(s_0 = 0)$

$$s_k = s_{k-1} + \left|(x_{jk}, y_{jk}) - (x_{j,k-1}, y_{j,k-1})\right|.$$

Given the values of the error monitor function at the points (x_{jk}, y_{jk}) (corresponding to s_k), we use the equi-distribution along the arc to obtain the new mesh $s'_0, s'_1, \cdots, s'_m$ i.e.

$$\int_{s'_i}^{s'_{i+1}} f_2 ds = constant.$$

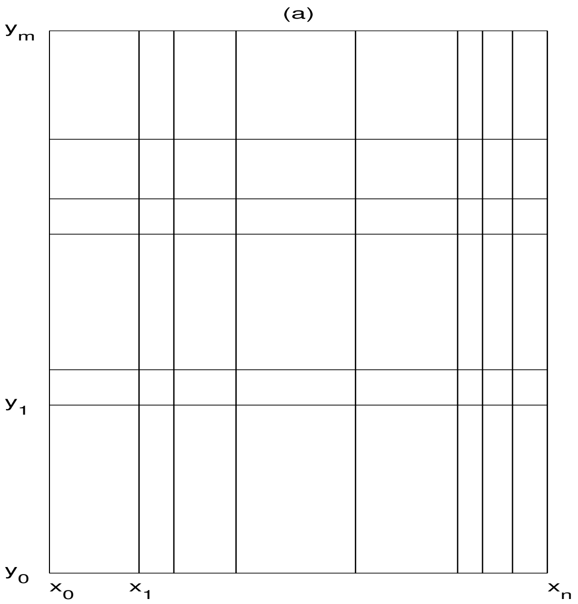
These values s'_i's can be used to generate the new coordinates (x_{jk}, y_{jk}) as the underlying arc is piecewise linearly represented. In details, the new mesh (x'_{ji}, y'_{ji}) can be obtained from

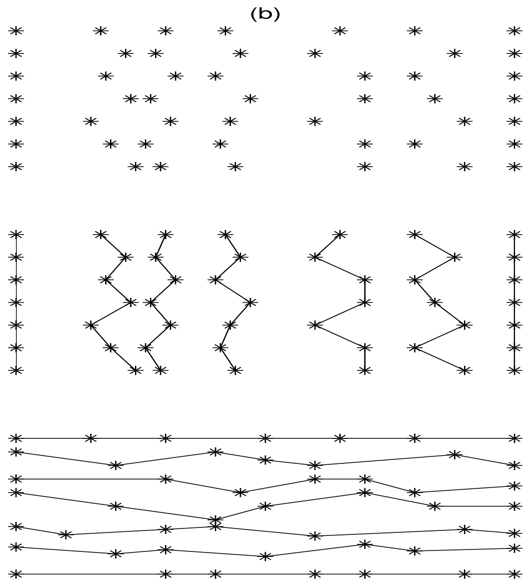
$$x'_{ji} = x_{jk} + \xi(x_{j,k+1} - x_{jk}), \qquad y'_{ji} = y_{jk} + \xi(y_{j,k+1} - y_{jk}),$$

where $\xi = \frac{s'_i - s_k}{s_{k+1} - s_k}$ and $s_k \leq s'_i \leq s_{k+1}$ for some k. See the middle part of the right plot in Fig 1b.

In the third and final stage, equi-distribution is repeated along the horizontal grid lines (refer to the bottom part of the right plot in Fig 1b). Numerical experiments have shown that repeating this procedure usually leads to a convergent mesh very quickly and the number of iterations to achieve convergence is at most five.

Figure 1: Comparison of a simple dimension reduction method using slabs [11] (left plot) and the new curved line equi-distribution method (bottom part of right plot). In the right plot, three stages of the new method are shown with the bottom part showing the final mesh.





For the above introduced curved line equi-distribution, we can add the minimal mesh size control idea (Algorithm 1) quite easily. This is because each curve line equi-distribution is a 1D process. We now give more examples using the combined approach. We consider generating a mesh for functions $u_1(x,y) = e^{\left(4-x^2-4y^2\right)^2}$ and $u_2(x,y) = 0.5 - 0.5\tanh(-4 + (4x-2)^2 + (4y-2)^2)$. For the first function, compare Figure 2a (left plot) using the new error equi-distribution method and Figure 2b (right plot) using the combined approach; clearly mesh points are less clustered on the right plot. For the second function, we display in Figure 3 the final mesh (right plot) produced by the combined approach.

5 Numerical Examples

We now present numerical results from RBF approximation using $1 + r^3$ and numerical solution of some linear PDEs by the DRBEM. We shall demonstrate that improvement in RBF approximation leads to improved accuracy in a DRBEM; this is consistent with the DRBEM convergence theory [13]. The results are given in three cases

(i) a uniform mesh,

(ii) an adaptive mesh with control and

(iii) an adaptive mesh without control.

We choose two sets of experiments: in set 1 (Examples 1 and 2) where the underlying function does not have sharp gradients in the solution domain, both the RBF approximation and the DRBEM are quite accurate with uniform meshes so adaptive mesh is not necessary and in set 2 (examples 3, 4, 5) where large variations of the solution present, we should observe that the proposed adaptive methods (especially the minimal control version (ii)) give better results. In Examples $1-4$, we solve the following linear PDE

$$\bigtriangledown^2 u + y\frac{\partial u}{\partial x} + x\frac{\partial u}{\partial y} + xyu = d, \tag{13}$$

with Dirichlet boundary conditions, d to be specified later and in the domain $\Omega = \{(x,y) \mid -2 \le x \le 2, -1 \le y \le 1\}$, and in Example 5, we solve [2]

$$-\epsilon\Delta u + \frac{\partial u}{\partial y} = x(1-x) + 2\epsilon\left(y - \frac{1-\exp(y/\epsilon)}{1-\exp(1/\epsilon)}\right) \tag{14}$$

where $\Omega = \{(x,y) \mid 0 \le x \le 1, 0 \le y \le 1\}$, ϵ to be specified later and $u|_{\partial\Omega} = 0$.

To test the numerical accuracy, we use these exact solutions (which are also used to test the accuracy of RBF interpolation and to define d in Examples $1-4$)

Example 1: $u = x^2y^2 + \sin(xy)$;

Example 2: $u = \exp(-x) + \exp(-y)$;

Example 3: $u = 0.5 - 0.5\tanh(-4 + (4x-2)^2 + (4y-2)^2)$;

Example 4: $u = (1 - \exp(3(x-1)))\sin(3\pi y/2)$;

Example 5: $u = x(1-x)\big(y - \dfrac{1-\exp(y/\epsilon)}{1-\exp(1/\epsilon)}\big)$.

The DRBEM solution is obtained using piecewise linear approximations and the collocation method with N boundary nodes and L internal nodes. The collocation points are selected as the Gaussian points. The solution error in the root mean-squares (RMS) norm is measured from uniformly distributed testing points different from the collocation points.

In Tables 1 and 2, we show respectively the experimental results for the 5 examples. There RBF refers to the task of RBF approximation for the u function using meshes (i)-(iii) and DRBEM refers to the DRBEM solution. Clearly one observes that the proposed mesh size control technique gives the overall best result. Further generalisation to non-rectangular domains is possible after mapping to a rectangular reference domain or using some domain embedding technique. Comparing this method with some preconditioning techniques for RBF interpolation is left for future work.

6 Conclusions

We presented a new technique for generating error equi-distributing meshes that satisfy both local quasi-uniformity and a preset minimal mesh size. The method is developed for 1D and then generated to the 2D case. Numerical experiments confirm that the proposed mesh size control combined with mesh adaption is more effective than mesh adaption alone in producing accurate solutions for our applications of RBF approximation and DRBEM solution.

Acknowledgement. The authors are grateful to the excellent anonymous referees who gave critical and helpful comments on the previous drafts of the paper.

Table 1: Comparison of solution accuracy (RMS error) for test set 1. Observe that the unform mesh already gives accurate solutions.

N=96 and L=105		Set 1 Example 1	Set 1 Example 2
Problem	Mesh	Error	Error
RBF	(i) Uniform	6.3E-3	1.6E-3
	(ii) Control	6.3E-3	9.4E-4
DRBEM	(i) Unform	6.3E-3	1.3E-3
	(ii) Control	6.5E-3	9.5E-4

Table 2: Comparison of solution accuracy (RMS error) for test set 2. Observe that the proposed mesh control gives more accurate solutions than the unform mesh and the adaptive mesh without control cases.

Test set 2		Example 3	Example 4	Example 5: N=400, L=2400	
	Results	N=240, L=740	N=300, L=1180	$\epsilon = 10^{-5}$	$\epsilon = 10^{-6}$
Problem	Mesh	Error	Error	Error	Error
RBF	(i) Uniform	8.2E-3	4.3E-3	1.2E0	1.2E-1
	(ii) Control	1.0E-3	4.6E-3	6.3E-2	6.5E-2
	(iii) No control	1.8E-3	1.2E-2	1.9E0	2.1E-1
DRBEM	(i) Uniform	5.5E-2	6.1E-3	6.2E-1	6.0E+1
	(ii) Control	2.1E-2	6.1E-3	2.0E-1	2.0E-1
	(iii) No control	5.7E-2	2.0E-1	3.3E-1	3.5E-1

References

[1] Ainsworth M. and Oden T. J., *A Posterior Error Estimation in Finite Element Analysis*, John Wiley, 2000.

[2] Beckett G., Mackenzie J. A., Ramage A. and Sloan D. M., *On the numerical solution of one-dimensional PDEs using adaptive methods based on equi-distribution*, `J. Comput. Phys.`, 167 (2), pp. 372-392, 2001.

[3] Carey G. F. and Dinh H. T., *Grading functions and mesh redistribution*, `SIAM J. Numer. Anal.`, 22 (5), pp. 1028-1040, 1985.

[4] Chen K., *Two-dimensional adaptive quadrilateral mesh generation*, `Comm. Numer. Meth. Engng.`, 10, pp. 815-825, 1994.

[5] Chen K., *Error equi-distribution and mesh adaptation*, `SIAM J. Sci. Comput.`, 15 (4), pp. 798-818, 1994.

[6] Chen K., *Improving the accuracy of DRBEM for convective PDEs*, `J. Engng. Anal. Bound. Elem.`, 23, pp. 639-644, 1999.

[7] Kautsky J. and Nichols N. K., *Equi-distributing meshes with constraints*, `SIAM J. Sci. Statist. Comput.`, 1 (4), pp. 449-511, 1980.

[8] Pereyra V. and Sewell E. G., *Mesh selection for discrete solution of boundary problems in ordinary differential equations*, `Numer. Math.`, 23, pp. 261-268, 1975.

[9] Powell M. J. D., *The theory of radial basis function approximation in 1990*, Advances in Numerical Analysis II, ed. W. Light, Oxford Sci. Pub., Oxford, UK, 1992.

[10] Profit A., Chen K. and Amini S., *Application of the DRBEM with adaptive internal points to nonlinear dopant diffusion*, `Proc. 2nd UK BIE conf.`, Brunel University Press, 1999.

[11] Sweby P. K., *Data-dependent grids*, `Numerical Analysis Report 7/87`, University of Reading, UK, 1987.

[12] Cao W., Huang W. and Russell R. D., *A study of monitor functions for two-dimensional adaptive mesh generation*, `SIAM J. Sci. Comput.`, 20, pp. 1978-1994, 1999.

[13] Jumarhon B., Amini S. and Chen K., *On the convergence of the boundary element dual reciprocity method*, `J. Engng. Anal. Bound. Elem.`, 20, pp. 205-211, 1997.

[14] Jumarhon B., Amini S. and Chen K., *The Hermite collocation method using radial basis functions*, `J. Engng. Anal. Bound. Elem.`, 24, pp. 607-611, 2000.

[15] White A. B., *On selection of equi-distribution meshes for two-point boundary value problems*, `SIAM J. Numer. Anal.`, 19, pp. 472-502, 1979.

[16] Alarcon E. and Reverter A., *P-adaptive boundary elements*, `Int. J. Numer. Meth. Engng.`, 23, pp. 801-829, 1986.

[17] Sun W. and Zamani N. G., *Adaptive mesh redistribution for the boundary element in elastostatics*, `Comput. and Struct.`, 36 (6), pp. 1081-1088, 1990.

[18] Sun W. and Zamani N. G., *Adaptive r and $h-r$ algorithms boundary elements*, `Advances in Engineering Software`, 15, pp. 241-247, 1992.

[19] Sun W. and Zamani N. G., *An adaptive $h-r$ boundary element algorithm for the Laplace equation*, `Int. J. Numer. Meth. Engng.`, 33, pp. 537-552, 1992.

[20] Thompson J. F., Warsi z. U. A. and Mastin C. W., *Numerical Grid Generation Foundations and Applications*, North-Holland, 1985.

[21] Partridge P. W. Brebbia C.A and Wrobel L. C., *The Dual Reciprocity Boundary Element Method*, Computational Mechanics Publications, 1992.

[22] Zhu S. P., Liu H. W. and Chen K., *A general DRBEM model for wave refraction and diffraction*, `J. Engng. Anal. Bound. Elem.`, 24, pp. 377-390, 2000.

Figure 2: Comparison of a new error equi-distribution method (left plot) and the combined approach of the new equi-distribution with minimal mesh size control (right plot with $h_{min} = h_2/2$) for function u_1.

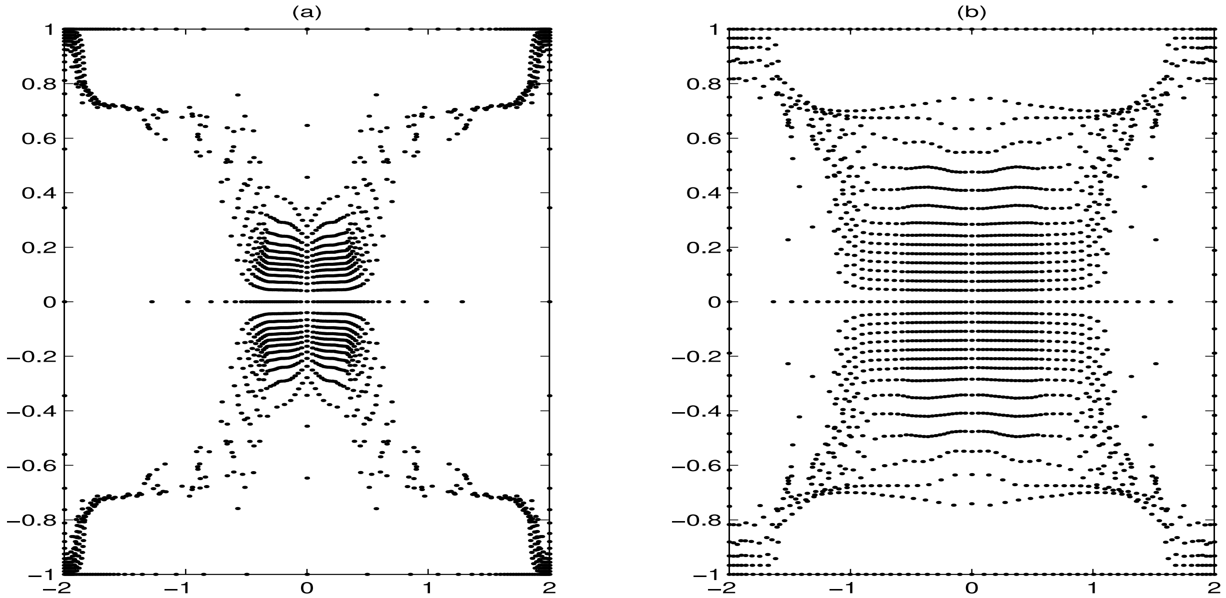


Figure 3: The adaptive mesh from a combined approach for function u_2 with the left plto showing the function and the right plot the final mesh.