Abstract

This paper describes a multiresolution approach to field modeling that can be used with any meshfree or mesh-based method for adaptive solution refinement. The refined solution is represented as a superposition of a coarse (unrefined) solution and a sequence of refinements that provide additional degrees of freedom with higher spatial or functional resolution. Each refinement is treated as a solution to a boundary value problem within a specified refinement window. The proposed approach is based on the meshfree method with distance fields [19, 29] and guarantees $C^m$ continuity of the refined solutions with matching or non-matching grids. The method does not restrict the shape of the refinement window and does not place any constraints on the type of basis functions, or relative position and resolution of the refinement grids. Combining the proposed approach with hierarchical space decompositions and a posteriori error estimators results in an effective tool for automatic solution refinement. Carefully chosen numerical examples illustrate the power and advantages of the proposed approach.

1 Introduction

1.1 The Field Refinement Problem

The accuracy of the physical fields being simulated on a computer largely depends on the properties of the basis functions used to approximate the solution. It is well known [5] that the degree of the basis functions and the size of their supports are the most important factors affecting the accuracy of a solution. This provides several recipes to reduce approximation error in field simulations. More accurate results can be obtained either by raising the degree of the basis functions, or decreasing the size of their supports or by both. These approaches are known as $p$- and $h$-refinements respectively [32]. Although increasing the degree $p$ of the basis functions results in better convergence of the solution [27], $h$ refinement remains quite popular [15, 10]. It is usually achieved by either global or local remeshing of the underlying geometric model or by introducing additional grids of finer resolution. Remeshing of the domain requires a global solution of the physical problem even if the spatial mesh was only locally refined. Furthermore, this refinement procedure restricts the shape of the refinement windows, and it cannot be used with non-matching grids such as those utilized by meshfree methods.

In contrast, methods that use multiple grids (matching or non-matching) enable local refinements utilizing previously computed solutions, without resolving the global problem. Additional degrees of freedom provided by grids with higher spatial resolution refine the coarse solution reducing the residual of the differential equation of the problem. The use of multiple grids allows to perform both global and local refinement.
Figure 1: (a) Global multiresolution refinement: grids with increased resolution are placed over the whole geometric domain. (b) Local multiresolution refinement: grids with increased resolution are placed over only those regions where the solution needs to be refined.

Figure 2: (a) Solution of a Poisson equation with homogeneous Dirichlet boundary conditions on a coarse $30 \times 30$ grid of bicubic B-splines. (b) Refined solution by "telescope" method (circular refinement window). (c) Magnitude of a gradient of the refined solution. (d) Discontinuity of the magnitude of a gradient on the boundary of the refinement window.
as shown in Figure 1. For a global solution, refinement grids with higher resolution are placed over the whole geometric domain. The coarse solution is propagated to the finer grid and then refinement procedure determines the refined field. Multiple grids are also used for local solution refinement. In this case, locations where the accuracy of the solution needs to be improved are determined by a posteriori error estimators [1, 8, 32]. Local refinement saves computational resources that might be wasted to refine the solution in the areas where the error is small or negligible.

Figure 3: (a) Refined solution by “telescope” method (square refinement window). (b) Magnitude of a gradient of the refined solution. (c) Discontinuity of the magnitude of a gradient on the boundary of the refinement window. (d) Discontinuity of the Laplacian at corners of the refinement window.

The key challenge in using multiple grids lies in merging solutions computed at different levels of spatial resolution with required degree of continuity [14]. A common approach to this problem involves a concept of the window function, i.e. a function $\omega_{w_j}$ that vanishes on the boundary of the refinement window and is strictly positive inside it. Such functions are also used with moving least square approximations [2]. In particular, in [21] Rvachev and Shevchenko proposed to represent the refined solution as:

$$u_j = u_{j-1} + u_{r_j}$$

where the refinement function $u_{r_j}$ is represented as a product of a window function $\omega_{w_j}$ corresponding to the refinement window $\Omega_{w_j}$ and undetermined function $\Phi$ that is represented by a linear combination of basis functions $\{\chi_i\}_{i=1}^N$:

$$u_{r_j} = \omega_{w_j} \Phi = \omega_{w_j} \sum_{i=1}^N C_i \chi_i.$$  \hspace{1cm} (2)

This refinement approach, called by the authors a “method of telescope”, is a direct application of the Kantorovich method for solving Dirichlet boundary value problems. It guarantees $C^0$ continuity of the solution, but it may fail when a higher order of continuity is required. In fact, the gradient of the refined solution $u_j$ represented by expression (1) is usually discontinuous at the boundary of the refinement window.
Differentiating expression (1) we see that the difference in gradient values on opposite sides of the refinement window \( \Omega_{w_j} \) is measured by the term \( \nabla \omega_{w_j} \sum_{i=1}^{N} C_i \chi_i \). Since \( \nabla \omega_{w_j} |_{\partial \Omega_{w_j}} \neq 0 \), the difference in gradient values is completely determined by the value of the linear combination \( \sum_{i=1}^{N} C_i \chi_i \) on the boundary \( \partial \Omega_{w_j} \). Figure 2(d) illustrates discontinuity in the gradient of the refined solution of a boundary value problem

\[
\nabla^2 u = -f; \\
u|_{\partial \Omega} = 0,
\]

where \( f = 10^4 \exp(-10^4(x^2+y^2)) \). \( C^0 \) continuity also prevents using overlapping refinement windows (Figure 4(b)) because this leads to divergent solutions as shown in Figure 5. In turn, side by side placement of the refinement windows (Figure 4(a)) precludes solution refinement on the boundaries of the windows. This results in insufficient accuracy of the solution and gradient distortions (Figure 6) along these boundaries.

**Figure 4:** (a) Side by side placement of refinement windows. (b) Overlapping refinement windows.

**Figure 5:** Divergence of the refinement using \( C^0 \) continuous overlapping window functions.

Furthermore, in order to refine the solution in the vicinity and on the boundary of a geometric domain, refinement functions must satisfy the boundary conditions prescribed on the domain’s boundary. However, representation of the refinement function by expression (2) allows treatment of Dirichlet boundary conditions only. In addition, improperly constructed window functions may introduce non-physical singularities into the refined solution \( u_j \). For example, window functions constructed using R-functions [23] are non-differentiable
Figure 6: Distortion of the gradient of the refined solution when the refinement windows do not overlap: (a) $C^0$ continuous; (b) $C^2$ continuous.

at the corners of the refinement window. Figure 3(d) illustrates discontinuities of the Laplacian of the solution refined using a square window used in [21]. Usually, singularities in physical fields are expected at corners of a geometric domain, but their introduction at internal points leads to non-physical field distributions.

1.2 Refinement as a Boundary Value Problem

In this paper we propose an approach to multiresolution modeling by reducing the problem of global refinement to a sequence of smaller boundary value problems solved locally within each refinement window. This implies that the constructed refinements must preserve the prescribed (arbitrary) order of continuity, satisfy given boundary conditions, and perform refinement near and on the domain’s boundary. To achieve these goals the refinement functions $u_{r_j}$ have to satisfy the following conditions:

- $C^m$ continuity condition on the window’s boundary. This condition implies that $u_{r_j}$ together with its normal derivatives up to order $m$ vanish on the boundary of the refinement window:

$$u_{r_j} \bigg|_{\partial \Omega_{w_j}} = 0, \quad \frac{\partial^k u_{r_j}}{\partial n^k} \bigg|_{\partial \Omega_{w_j}} = 0, \quad k = 1, \ldots, m.$$  (4)

The necessary order of continuity is determined by the order of the differential equation of the problem. For a second order differential equation we need to construct refinement functions preserving $C^2$ continuity. In this case $C^m$ continuity condition transforms into:

$$u_{r_j} \big|_{\partial \Omega_{w_j}} = 0, \quad \frac{\partial u_{r_j}}{\partial n} \bigg|_{\partial \Omega_{w_j}} = 0, \quad \frac{\partial^2 u_{r_j}}{\partial n^2} \bigg|_{\partial \Omega_{w_j}} = 0,$$  (5)

- Homogeneous boundary conditions on the domain’s boundary enclosed in the refinement window. These boundary conditions should be of the same type as the prescribed boundary conditions.
- Refinement function $u_{r_j}$ should be identically zero outside refinement window $\Omega_{w_j}$.
- Refinement function $u_{r_j}$ should not introduce non-physical singularities at internal points of a geometric domain.

Viewing each refinement as a boundary value problem suggests that the same numerical solution method can be used for computing a coarse solution and its refinements, provided that the above conditions may
be enforced. With a meshfree approach, the refinement functions may be constructed for arbitrarily shaped refinement windows, as well as using non-matching and/or overlapping grids. In the context of the meshfree method with distance fields the refinement functions \( u_{r_j} \) can be constructed using solution structures\(^1\) satisfying continuity and homogeneous boundary conditions prescribed on the boundaries of the refinement window and geometric domain. The key feature of this meshfree method is exact treatment of the prescribed boundary conditions using approximate distance fields to the boundaries where the boundary conditions are specified. Approximate distance fields, being smooth approximations to the Euclidean distance, behave as Euclidean distance only in the vicinity of the boundary. They may, however, substantially deviate from the Euclidean distance away of the boundary. In \([23, 29, 24, 7]\) we demonstrated automatic construction of the approximate distance fields with guaranteed differential properties from most geometric representations. The Appendix contains a brief technical summary of the meshfree method with distance fields; for additional details, the reader is referred to \([18, 19, 30]\).

The rest of the paper is organized as follows. Section 2 describes the construction of the refinement functions for all types of boundary conditions for second order partial differential equations. In Section 3, we discuss the adaptive solution procedure based on the proposed approach to multiresolution modeling. The method was fully implemented, and Section 4 demonstrates several engineering applications where the adaptive refinement solutions proved to be effective.

2 Construction of refinement functions using distance fields

In this Section we will show how to construct multiresolution solution refinements \( u_{r_j} \) for the Poisson equation

\[
\nabla^2 u = -f
\]

using distances to boundaries of the geometric domain and refinement windows.

If the refinement window intersects the boundary of a geometric domain, the refinement function must satisfy both the continuity on the window’s boundary and the boundary conditions specified on the domain’s boundaries. Two approaches are possible for constructing such refinement functions. We could construct individual solutions structures to satisfy the continuity conditions and the boundary conditions separately, and then transfinite interpolate them as described in \([20]\). This would result in a refinement function that contains several sets of basis functions corresponding to the individual solution structures. This approach works well if these basis functions are defined globally, as is the case with polynomials or radial basis functions. But if the basis functions have finite support, transfinite interpolation of the solution structures leads to highly conditioned matrices which reduce stability and accuracy of the solution \([28]\).

Alternatively, the refinement function can be represented by a single solution structure that incorporates basis functions, continuity and boundary conditions without using transfinite interpolation. In the rest of this Section we will demonstrate construction of such refinement functions \( u_{r_j} \) for Dirichlet, convective and mixed boundary conditions using a general meshfree approach with distance fields.

2.1 Refinement functions for windows that do not intersect boundary

We start our discussion with the simplest case — when the refinement window does not intersect the boundary of a geometric domain. This requires the refinement function \( u_{r_j} \) to satisfy the continuity condition and introduce no singularities to the refined solution \( u_i \) \((1)\). Let us represent the refinement function as a product of a window function \( W_j \) and linear combination of basis functions \( \{\chi_k\}_{k=1}^N \):

\[
 u_{r_j} = W_j \sum_{k=1}^N C_k \chi_k. 
\]

\(^1\)Term solution structure was introduced by Rvachev \([18]\), and it defines a function that satisfies prescribed boundary conditions exactly and contains sufficient number of degrees of freedom in order to approximate the differential equation of the problem. Please see Appendix for more information.
Figure 7: Window functions that preserve (a) $C^0$ continuity, (b) $C^1$ continuity, (c) $C^2$ continuity. (d-f) Absolute values of gradients of the window functions in Figures 7(a-c). (g-h) Laplacians of the window functions in Figures 7(a-c).
The window function $W_j$ should be constructed so that the addition of $u_{rj}$ to the coarse solution $u_{j-1}$ will guarantee $C^m$ continuity of the refined solution $u_j$ on the boundary of the refinement window. In order to satisfy this condition the window function $W_j$ must vanish together with its normal derivatives up to $m$th order on the window’s boundary. Such a function can be obtained by raising the approximate distance field $\omega_{wj}$ to the $(m + 1)$ power $W_j = \omega_{wj}^{m+1}$. In order to make $W_j$ identically zero outside of the refinement window $\Omega_{wj}$ we multiply the last expression by a characteristic function $\delta_{wj}$:

$$W_j = \omega_{wj}^{m+1} \delta_{wj}. \quad (8)$$

The function $\delta_{wj}$ returns 1 for all internal and boundary points of the refinement window $\Omega_{wj}$ and 0 for all other points:

$$\delta_{wj}(x) = \begin{cases} 
1, & x \in \Omega_{wj}; \\
0, & x \not\in \Omega_{wj},
\end{cases}$$

If the approximate distance field $\omega_{wj}$ is positive inside of $\Omega_{wj}$ and negative outside $\delta_{wj}$ can be defined using the Heaviside function:

$$\delta_{wj} = H(\omega_{wj}).$$

Figures 7(a), (b) and (c) show plots of window functions that guarantee $C^0$, $C^1$ and $C^2$ continuity of the refined solution. These plots illustrate substantial vanishing of the window function with increasing order of continuity. This could lead to numerically singular matrices the during solution process. To avoid excessive growth or vanishing of the window function $W_j$ it needs to be scaled so that its maximum value coincides with the maximum value of $\omega_{wj}$:

$$W_j = a \omega_{wj}^{m+1} \delta_{wj}, \quad (9)$$

where

$$a = \max(\omega_{wj}(x))^{-m}, \quad \forall x \in \Omega_{wj}. \quad (10)$$

Figure 8 presents window functions shown in Figures 7(a), (b) and (c) scaled using expression (9). Comparing these figures we can see dramatic improvement of the window functions: scaling guarantees that the maximum value of the function does not depend on the order of continuity.

Figure 8: Scaled window functions that guarantee (a) $C^0$ continuity, (b) $C^1$ continuity, (c) $C^2$ continuity.

Representation of the window functions $W_j$ in the form (9) also suppresses singularities that could be introduced by the approximate distance field $\omega_{wj}$. If $\omega_{wj}$ is not differentiable at certain points on the boundary of the refinement window, raising $\omega_{wj}$ to the $m + 1$ power assures it is $m$ times differentiable at these points.

The continuity order $m$ on the boundary of a refinement window is mainly dictated by the order of a boundary value problem being solved. For second order partial differential equation with continuous coefficients and continuous the right hand side function $f$ (6), the refinement functions $u_{rj}$ should preserve second order continuity. Discontinuities in the coefficients of the differential equation lead to solutions with
discontinuous gradients and therefore require special treatment as described in [30]. Numerical values of the coefficients $C_k$ in expression (7) are determined from the refinement equation:

$$\nabla^2 u_{r_j} = -f - \nabla^2 u_0 - \sum_{i=1}^{j-1} \nabla^2 u_{r_i},$$

which can be solved by the same method that was used to find a coarse solution $u_0$.

Figure 9: (a) Coarse solution of a Poisson equation with homogeneous Dirichlet boundary conditions computed using a uniform 30 × 30 Cartesian grid of bicubic B-splines. (b) Accurate solution obtained using uniform 300 × 300 Cartesian grid of bicubic B-splines. (c) Refined solution computed using a circular refinement window and bicubic B-splines over a 150 × 150 grid. (d) Refined solution computed using a square refinement window and bicubic B-splines over a 150 × 150 grid.

**Example** Let us illustrate solution refinement by solving the Poisson equation (3) with right hand side function $f$ given by the expression $f = 10^4 \exp(-10^4(x^2 + y^2))$. Homogeneous Dirichlet boundary condition are specified on the boundary of a circle with unit radius. For clarity, we put detailed derivation of the solution into Appendix A.3. Here we just show and compare solution results. The coarse solution shown in
Figure 9(a) was obtained by using bicubic B-splines defined on 30 × 30 uniform Cartesian grid. It takes on its maximum value of 1.8713 at the origin. It is 23.46% less than maximum value (2.4448) of more accurate solutions computed on 300 × 300 grid (Figure 9(b)).

Now, let us refine the coarse solution using the circular and square windows as illustrated in Figures 10(a) and 10(b). The center of the circular window is at the origin, and the radius of the window is 0.5. The square window is defined by its extreme points (−0.5, −0.5) and (0.5, 0.5). Expression (7), with a window function given by (9), represents the refinement function $u_r$. The latter expression uses an approximate distance field $\omega_w$ whose zero set describes the boundary of the refinement window. The approximate distance field for the circular window can be written as follows: $\omega_w = 0.25 - x^2 - y^2$. For a square window $\omega_w$ can be constructed using $R$-functions [17, 18]. The square shown in Figure 11 is an intersection of two bands, $\Omega_1$ and $\Omega_2$ that are parallel to coordinate axes: $\Omega_w = \Omega_1 \cap \Omega_2$. Each band can be described by the inequalities: $\Omega_1 = 0.25 - x^2 \geq 0$ and $\Omega_2 = 0.25 - y^2 \geq 0$. Syntactic substitution of $R_0$-conjunction for Boolean intersection gives us a distance-like function $\omega_w = (0.25 - x^2) \land (0.25 - y^2)$. This function is positive inside the refinement window $\Omega_w$, negative outside and takes on zero on the window’s boundary. Figure 7(a) presents a plot of $\omega_w H(\omega_w)$. After raising $\omega_w$ to the third power and scaling according to expression (9) we obtain a window function $W$ that guarantees $C^2$ continuous refinement (Figure 8(c)).

We choose the basis functions $\chi_k$ in the expression (7) to be bicubic B-splines defined on a 150 × 150 uniform Cartesian grid. This grid has the same spatial resolution as a 300 × 300 grid put over the whole geometric domain. To determine numerical values of the coefficients $C_k$ in the refinement function $u_r$ (7) we solve the refinement equation (11) with the function $u_r$, subject to $C^2$ continuity conditions on the window’s boundary. Application of the Ritz method results in a system of linear algebraic equations $\mathbf{AC} = \mathbf{B}$ whose elements are given by the following integrals:

$$a_{lm} = \iint_{\Omega_w} \nabla (W \chi_l) \cdot \nabla (W \chi_m) d\Omega_w;$$

$$b_l = \iint_{\Omega_w} (W \chi_l) 10^4 e^{-10^4(x^2+y^2)} d\Omega_w - \iint_{\Omega_w} \nabla (W \chi_l) \cdot \nabla (u_0) d\Omega_w.$$

Plots of the refined solutions are shown in Figures 9(c) and (d) for circular and square refinement windows respectively. Comparing these plots with the accurate solution shown in Figure 9(b), we notice that the maximum values of the refined solutions are just 0.81% and 0.72% higher than the maximum value of the solution obtained on a 300 × 300 grid.

---

2 $R$-functions have been introduced by Rvachev in [17] as a mathematical tool to construct implicit equations $\omega = 0$ for the boundaries of the geometric domains defined by Boolean functions. For more information on $R$-functions please refer to [22, 23].

3 $R_0$-conjunction “$\land$” is given by expression $x \land y = x + y - \sqrt{x^2 + y^2}$ [23].
Figure 11: A square window is the intersection of two bands parallel to coordinate axes.

Figure 12: (a) The magnitude of the gradient of the refined solution (circular refinement window) shown in Figure 9(c). The white rectangle encloses the area around an edge of the refinement window. (b) Continuous distribution of the magnitude of gradient of the refined solution near an edge of the refinement window. (c) Continuous distribution of Laplacian of the refined solution near an edge of the refinement window. (d) A magnitude of the gradient of the refined solution (square refinement window) shown in Figure 9(d). White rectangle encloses the area around the corner of the refinement window. (e) Distribution of a magnitude of gradient of the refined solution near the corner of the refinement window. (f) Distribution of Laplacian of the refined solution near a corner of the refinement window.
Figures 12(a) and (d) present plots of the absolute values of the gradient of the refined solutions. The areas within the white rectangles (intersecting the refinement windows) are magnified in Figures 12(b) and (e). The corresponding distributions of the Laplacian of the refined solutions in the vicinity of the refinement window’s boundaries are shown in Figures 12(c) and (f). The continuity of the Laplacian in these plots is consistent with the enforcement of $C^2$ continuity condition imposed on the refinement function (7) and the absence of singularities at the corners of the refinement window. It is instructive to compare the gradient and the Laplacian obtained by our method to those computed by the “telescope” method (Figures 2(d), 3(c), and 3(d) respectively.

![Figure 13](image)

Figure 13: (a) Placement of the refinement window $\Omega_w$. (b) Load function $f = 10^4 \exp(-10^4((x-0.9)^2+y^2))$.

![Figure 14](image)

Figure 14: (a) A window function that satisfies $C^2$ continuity conditions and homogeneous Dirichlet boundary condition. (b) A magnitude of the gradient of the window function shown in Figure 14(a). (c) Laplacian of the window function shown in Figure 14(a).

2.2 Refinement functions for windows that intersect boundary with Dirichlet boundary conditions

When the refinement window intersects a portion of the domain’s boundary with Dirichlet boundary conditions, the refinement function $u_{r_j}$ can still be represented by expression (7). In this case, however, it has to satisfy not only $C^2$ continuity condition (5) imposed on the window’s boundary $\partial\Omega_{w_j}$, but also the homogeneous Dirichlet boundary condition on the portion of the domain’s boundary:

$$u_{r_j}|_{\partial\Omega_{w_j}} = 0.$$ (12)
In order to satisfy these conditions the window function \( W_j \) needs to be constructed in such a way that it behaves as \( \omega^3_{w_j} \) in the neighborhood of the window’s boundary \( \partial \Omega_{w_j} \) and as \( \omega \) in the vicinity of the domain’s boundary \( \Omega \). Simple multiplication\footnote{Functions \( \omega^3_{w_j} \) and \( \omega \) can be also joined by R-functions \cite{18, 22, 23}. They result in functions that better preserve distance properties \cite{23}.} of \( \omega^3_{w_j} \) and \( \omega \) gives us a window function with the desired properties:

\[
W_j = \omega^3_{w_j} \omega,
\]

(13)

where

\[
a = \max(\omega_{w_j}(x))^{-2}, \quad \forall x \in \Omega_{w_j}.
\]

(14)

Finally, we multiply expression (13) by a characteristic function \( \delta_{w_j} \):

\[
W_j = a \omega^3_{w_j} \omega \delta_{w_j}
\]

(15)

that is defined as follows:

\[
\delta_{w_j}(x) = \begin{cases} 
1, & x \in \Omega_{w_j} \cap \Omega; \\
0, & x \not\in \Omega_{w_j} \cap \Omega.
\end{cases}
\]

Figure 15: (a) Coarse solution of a Poisson equation with homogeneous Dirichlet boundary conditions obtained using bicubic B-splines on a 30 x 30 uniform Cartesian grid. (b) Accurate solution obtained using bicubic B-splines on a 300 x 300 uniform Cartesian grid. (c) Refined solution computed using bicubic B-splines on a 150 x 150 grid.

Example Let us solve the Poisson equation (6) with homogeneous Dirichlet boundary condition \( u|_{\partial \Omega} = 0 \) prescribed on the boundary of a circular domain of unit radius (Figure 13). The right hand side of a Poisson equation is given by a function \( f = 10^4 \exp(-10^4((x-0.9)^2 + y^2)) \) whose plot is shown in Figure 13(b). In contrast to the previous example, the peak is moved from the center towards the domain’s boundary.

Figure 15(a) presents a plot of a coarse solution computed using bicubic B-splines on 30 x 30 uniform Cartesian grid. Its maximum value is 1.1336 which is 29.4\% less than the maximum value (1.6057) of the more accurate solution obtained on a 300 x 300 grid (Figure 15(b)). We will refine the coarse solution shown in Figure 15(a) by placing a square window \( \Omega_{w} \) (Figure 13(a)) around the point (0.9,0) where the right hand side function \( f \) takes on its maximum value. The refinement function \( u_r \) is represented by expression (7). The window function \( W \), shown in Figure 14(a), is constructed using expression (15) where \( \omega = \frac{1}{2}(1-x^2-y^2) \) and \( \omega_{w} = (0.25 - (x-0.9)^2) \wedge (0.25 - y^2) \). As previously, \( \wedge \) denotes \( R_0 \) conjunction \cite{18, 23}. Plots of
the magnitude of the gradient and Laplacian, presented in Figures 14(b) and (c), illustrate that \( W \) has zero gradient and Laplacian on the window’s boundary \( \partial \Omega_w \). This guarantees \( C^2 \) continuity of the refined solution. The window function \( W \) takes on zero value on the domain’s boundary \( \partial \Omega \) where it has non-zero gradient. This allows to satisfy the homogeneous Dirichlet boundary condition and assure completeness of the refinement function \( u_r \) [13]. To approximate the refinement equation (11) \( u_r \) uses bicubic B-splines defined over a 150 \( \times \) 150 uniform Cartesian grid. Numerical values of the coefficients \( C_k \) in expression (7) are computed using the Ritz method. Figure 15(c) presents a plot of the refined solution. Its analysis reveals that the maximum value (1.6097) of refined solution is just 0.24\% higher than accurate solution.

Figure 16: (a) Coarse solution of a Poisson equation with homogeneous convective boundary conditions obtained using biquintic B-splines on 30 \( \times \) 30 uniform cartesian grid. (b) Non-refined solution obtained using biquintic B-splines on 300 \( \times \) 300 uniform cartesian grid. (c) Solution refined by the function given by expression (18). (d-f) Laplacians of the solutions presented in Figures 16(a-c).

2.3 Refinement functions for windows that intersect boundary with convective boundary conditions

If the refinement window contains a portion of the domain’s boundary with a convective boundary condition, the refinement function \( u_r \) has to satisfy \( C^2 \) continuity conditions (5) on the window boundary \( \partial \Omega_w \) and a homogeneous convective boundary condition on the domain’s boundary \( \partial \Omega \):

\[
\left. \left( \frac{\partial u_r}{\partial n} + hu_r \right) \right|_{\partial \Omega} = 0.
\]

Such a refinement function can be constructed using a solution structure that satisfies the homogeneous mixed boundary condition

\[
u_r \big|_{\partial \Omega_w} = 0, \quad \left. \left( \frac{\partial u_r}{\partial n} + hu_r \right) \right|_{\partial \Omega} = 0.
\]
Figure 17: (a) Normalized distance field to circle’s boundary. (b) Normalized distance field to refinement window’s boundary. (c) Normalized distance field $\omega^*$. 

Following Rvachev [18], this solution structure appears as follows:

$$u_{rj} = \omega_{wj} \Phi - \omega^* D^1_1[\omega_{wj} \Phi] - h\omega^* \omega_{wj} \Phi,$$

(17)

where $\omega^*$ is a normalized function that takes on zero value on both boundaries $\partial \Omega_{wj}$ and $\partial \Omega$; $D^1_1[\cdot]$ is a first order differentiation operator in the normal direction to the domain’s boundary $\partial \Omega$. If $\omega$ is a normalized function, $D^1_1$ can be represented as a dot product of gradients: $D^1_1[\cdot] = \nabla \omega \cdot \nabla [\cdot]$. The term $\omega_{wj} \Phi$ in the solution structure (17) ensures exact treatment of the homogeneous Dirichlet boundary condition on the boundary $\partial \Omega_{wj}$. To satisfy the $C^2$ continuity condition (5) imposed on the refinement function this term should be replaced by a term that satisfies this condition: $a\omega_{wj}^3 \delta_{wj} \sum_{k=1}^{N} C_k \chi_k$, where $a = \max(\omega_{wj}(x))^{-2}$, $\forall x \in \Omega_{wj}$. As a result we obtain:

$$u_{rj} = a \omega_{wj} \sum_{k=1}^{N} C_k \left( \omega_{wj}^3 \chi_k - \omega^* D^2_1[\omega_{wj}^3 \chi_k] - h\omega^* \omega_{wj}^3 \chi_k \right).$$

(18)

The normalized function $\omega^*$ can be easily constructed from the normalized distance fields $\omega$ and $\omega_{wj}$ to the domain’s and refinement window’s boundaries. If $\omega$ and $\omega_{wj}$ are signed distance fields application of $R$-conjunction gives the desired function: $\omega^* = \omega \wedge \omega_{wj}$. Otherwise a normalized distance field to the trimmed portion of the domain’s boundary needs to be constructed first, then it can be combined by $R$-conjunction with $\omega_{wj}$.

**Example** Let us illustrate refinement of the solution of the Poisson equation (6) with right hand side function $f = 10^4 \exp(-10^4((x-0.9)^2 + y^2))$ and homogeneous convective boundary conditions

$$\left( \frac{\partial u}{\partial n} + 0.1u \right) \bigg|_{\partial \Omega} = 0,$$

where $\partial \Omega$ is the boundary of a circle of unit radius with center at the origin (Figure 13(a)). We represent $u$ by a solution structure that satisfies the homogeneous convective boundary condition (16) exactly:

$$u = \Phi + h\omega \Phi - \omega D^1_1[\Phi],$$

(19)

$^5$A function $\omega$ is called normalized up to $m$th order on the boundary $\partial \Omega$ if it satisfies the following conditions: $\omega|_{\partial \Omega} = 0$, $\frac{\partial \omega}{\partial n}|_{\partial \Omega} = 1$, $\frac{\partial^i \omega}{\partial n^i}|_{\partial \Omega} = 0$, $i = 2, \ldots, m$. Construction of normalized functions using R-functions is described in references [18, 23].

$^6$Sign of a distance field can be used to distinguish internal points of a geometric domain from external ones. In our notation positive values of distance fields correspond to internal points.
where $\Phi$ is represented by a linear combination of biquintic B-splines $\Phi = \sum_{l=1}^{N} C_l \chi_l$ defined on a $30 \times 30$ uniform Cartesian grid. Application of the Ritz method gives a coarse solution whose plot is shown in Figure 16(a). The maximum value of this solution (7.6558) is 6.25\% smaller than the maximum value of a more accurate solution computed using a $300 \times 300$ grid. The solution structure (19) assures exact treatment of boundary conditions, while degrees of freedom in function $\Phi$ are used to approximate the Poisson equation (6). Since boundary conditions are satisfied exactly, the solution error is determined by the approximation error of the differential equation. Figure 16(d) presents distribution of the remainder of Poisson equation. A peak is clearly visible in the vicinity of the point $(0, 0.9)$ which means that the $30 \times 30$ grid has insufficient resolution to accommodate the Poisson equation. A solution computed on a $300 \times 300$ grid (Figure 16(b)) better approximates the differential equation. The plot of the remainder term in Figure 16(e) illustrates that the peak at the point $(0, 0.9)$ is absent but there are spikes on the boundary. The appearance of these spikes is explained by a high conditioning of the matrix caused by those B-splines that have only a small portions of their supports inside the geometric domain. This problem has been addressed by Höllig in [11, 9].

Now let us refine the coarse solution that is shown in Figure 16(a). The refinement window is defined by its lower left $(0.65, -0.25)$ and upper right $(1.15, 0.25)$ points. We represent the refinement function $u_r$ by expression (18). It contains normalized distance fields for the domain’s and the window’s boundaries as well as normalized distance for the combined domain and window boundaries. The normalized distance field for the boundary of a circle (Figure 17(a)) is given by the expression $\omega = \frac{1 - x^2 - y^2}{\sqrt{1 - x^2 - y^2}}$. The normalized distance field in Figure 17(b) is constructed by application of $R_0$-conjunction to normalized distance fields to the boundaries of two bands parallel to the coordinate axes: $\omega_w = 2 \left(0.25^2 - (x - 0.9)^2\right) \wedge 2 \left(0.25^2 - y^2\right)$. Since $\omega$ and $\omega_w$ are signed normalized functions, they can be combined into the normalized function $\omega^*$ (Figure 17(c)) by $R_0$-conjunction: $\omega^* = \omega \wedge \omega_w$. Numerical values of the coefficients $C_k$ in (18) were obtained by the Ritz method. Figure 16(c) presents a plot of the refined solution computed using biquintic B-splines defined on a $75 \times 75$ grid. The maximum value (8.2345) of the refined solution is only 0.84\% higher than the maximum value of the solution obtained using a $300 \times 300$ grid of B-splines. Analysis of the remainder term whose plot is given in Figure 16(f) shows that the peak at the point $(0, 0.9)$ is absent and spikes on the boundary are much smaller than for the non-refined solution computed using the high resolution $300 \times 300$ grid.

![Figure 18: (a) Circular domain on whose boundaries Dirichlet and convective boundary conditions are prescribed. (b) Normalized distance field to the boundary $\partial\Omega_D$ shown in Figure 18(a). (c) Normalized distance field to the boundary $\partial\Omega_N$ shown in Figure 18(a).](image-url)
2.4 Refinement functions for windows that intersect boundary with mixed boundary conditions

When a refinement window intersects portions of the domain’s boundary with Dirichlet and convective boundary conditions, the refinement function $u_{r_j}$ is subject to the $C^2$ continuity condition (5) on the window boundary $\partial \Omega_{w_j}$, homogeneous Dirichlet boundary condition on $\partial \Omega_D$ and homogeneous convective boundary condition on $\partial \Omega_N$:

$$u_{r_j}|_{\partial \Omega_D} = 0, \quad \left( \frac{\partial u_{r_j}}{\partial n} + h u_{r_j} \right)|_{\partial \Omega_N} = 0. \quad (20)$$

First, we combine the mixed boundary conditions (20) with the continuity conditions (5):

$$u_{r_j}|_{\partial \Omega_{w_j} \cup \partial \Omega_D} = 0; \quad (21)$$

$$\left( \frac{\partial u_{r_j}}{\partial n} + h u_{r_j} \right)|_{\partial \Omega_N} = 0; \quad (22)$$

$$\frac{\partial u_{r_j}}{\partial n}|_{\partial \Omega_{w_j}} = \frac{\partial^2 u_{r_j}}{\partial n^2}|_{\partial \Omega_{w_j}} = 0. \quad (23)$$

Then we construct a refinement function $u_{r_j}$ that satisfies mixed boundary conditions (21) and (22). Such a function can be represented by the following solution structure [18]:

$$u_{r_j} = \omega_{w_j} \omega_D \Phi - \omega^* D_1^{\Omega_N} [\omega_{w_j} \omega_D \Phi] - h \omega^* \omega_{w_j} \omega_D \Phi. \quad (24)$$

Now we modify this solution structure in order to satisfy the $C^2$ continuity condition (23). The term $\omega_{w_j} \omega_D \Phi$ in the solution structure (24) is responsible for treatment of the Dirichlet boundary condition (21). Multiplication of this term by $a \omega_{w_j} ^2 \delta_{w_j}$, where $a = \max (\omega_{w_j}(x))^{-2}$, $\forall x \in \Omega_{w_j}$, gives the refinement function $u_{r_j}$ that satisfies mixed and $C^2$ continuity boundary conditions:

$$u_{r_j} = a \omega_{w_j} ^3 \omega_D \delta_{w_j} \Phi - \omega^* D_1^{\Omega_N} [a \omega_{w_j}^3 \omega_D \delta_{w_j} \Phi] - h \omega^* a \omega_{w_j} ^3 \omega_D \delta_{w_j} \Phi. \quad (25)$$

Finally, substituting $\Phi = \sum_{k=1}^{N} C_k \chi_k$ into (25) and regrouping the terms we obtain:

$$u_{r_j} = a \delta_{w_j} \sum_{k=1}^{N} C_k \left( \omega_D \omega_{w_j}^3 \chi_k - \omega^* D_1^{\Omega_N} [\omega_D \omega_{w_j}^3 \chi_k] - h \omega^* \omega_D \omega_{w_j}^3 \chi_k \right). \quad (26)$$
Example This example will demonstrate solution refinement of the Poisson equation (6) with right hand side function \( f = 10^4 \exp(-10^4((x-0.9)^2 + y^2)) \) and mixed homogeneous boundary conditions

\[
\begin{align*}
    u|_{\partial \Omega_D} &= 0; \\
    \left( \frac{\partial u}{\partial n} + 0.1u \right) |\partial \Omega_N &= 0,
\end{align*}
\]

where \( \partial \Omega_D \) and \( \partial \Omega_N \) are upper and lower arcs of a circle of unit radius with center at the origin (Figure 18). We represent the non-refined solution \( u \) by a solution structure that satisfies mixed boundary conditions:

\[
u = \omega_D \Phi - \omega_D \omega_N \Phi - h \omega_D \Phi.
\]

Plots of normalized distance fields \( \omega_D \) and \( \omega_N \) to Dirichlet and convective boundaries are shown in Figures 18(b) and (c). These distance fields were constructed from the normalized distance field shown in Figure 17(a) by normalized trimming [23]. Figure 19(a) shows a coarse solution computed using a 30 \times 30 uniform Cartesian grid of biquintic B-splines. The maximum value of this solution is 1.2879 which is 33.18\% less than the maximum value (1.9273) of more accurate solution (Figure 19(b)) computed on a 300 \times 300 grid.

To refine the coarse solution, we place a refinement window as shown in Figure 18(a). The position of the window is defined by the coordinates of its lower left (0.4, -0.5) and upper right (1.4, 0.5) points. We represent the refined solution by expression (26), where the basis functions \( \chi_k \) are biquintic B-splines defined over a 150 \times 150 uniform Cartesian grid. After numerical values of the coefficients \( C_k \) in expression (26) are computed using the Ritz method, the refined solution (1) is sampled on a 500 \times 500 grid. The plot of the refined solution is shown in Figure 19(c). The maximum value of the refined solution is 1.9197 which is just 0.39\% less than the maximum value of the solution computed on a 300 \times 300 grid.

3 Adaptive refinement of meshfree solution using window functions

In the previous Section we showed how to construct refinement functions that satisfy continuity and homogeneous boundary conditions imposed on the window and domain’s boundaries. Additional degrees of freedom contained in these functions allow to improve solution’s accuracy locally within the refinement window. In this Section we will show how the refinement functions can be combined for adaptive meshfree refinement of the solution. Usually this procedure consists of the following steps: (1) solution of a boundary value problem using initial set of basis functions; (2) a posteriori error estimation; and (3) refinement if the error is larger than specified. Steps 2 and 3 can be repeated until the solution error becomes equal or smaller.
than prescribed. The design of adaptive refinement procedure requires careful choice of a posteriori error estimator, locations of the refinement windows, and resolution of the basis functions for each window.

Figure 21: Solutions of the Poisson equation with right hand side function $f$ (Figure 20(b)) computed using (a) $50 \times 50$ and (b) $500 \times 500$ grids of bicubic B-splines.

An error estimator is used to determine which areas of a geometric domain require refinement and to assess an overall solution error in order to terminate the refinement process. It can be chosen from two classes of error estimators. Residual based error estimators \[1, 8\] assess the residuals of the differential equation and boundary conditions. They are easy to construct and use, but they often require higher smoothness of the solution than the chosen solution method. For example, application of the Ritz method to solve a second order PDE makes linear basis functions admissible \[26\]. But, in order to estimate the residual in a differential equation, the basis functions have to be at least twice differentiable. Alternatively, recovery based error estimators determine the solution error by comparing the approximate solution to its local polynomial approximation (called “recovered solution”) \[32\]. These error estimators are considered to be more accurate than residual based ones, but the construction of a reliable recovery based error estimator still is a non-trivial task.

Since construction of an error estimator falls outside the scope of this paper, we demonstrate adaptive refinement using a residual based estimator because of its simplicity. Let us solve the Poisson equation with right hand side given by a highly oscillating function whose plot is shown in Figure 20(b):

$$f = 10^6 \omega_c H(\omega_c) \sin(x_1) \sin(y_1),$$

where $\omega_c = 2 \left( 0.25^2 - (x - 0.5)^2 - (y - 0.5)^2 \right)$, $H(\omega_c)$ is a Heaviside function, $x_1 = 50 \pi (x \cos(\pi/3) + y \sin(\pi/3))$, $y_1 = 50 \pi (-x \sin(\pi/3) + y \cos(\pi/3))$. Dirichlet boundary conditions are specified on the boundaries of a geometric domain shown in Figure 20(a):

$$u|_{\partial \Omega_1} = 0, \quad u|_{\partial \Omega_2} = 1.$$

Boundaries $\Omega_1$ and $\Omega_2$ are concentric circles with centers at the origin and radii of 1 and 0.25 respectively.

A coarse (non-refined) solution $u_0$ is represented by a solution structure that satisfies Dirichlet boundary conditions (28) exactly:

$$u_0 = \omega \sum_{i=1}^{N} C_i \chi_i + \varphi,$$

where $\varphi$ interpolates values 0 and 1 between the boundaries $\partial \Omega_1$ and $\partial \Omega_2$. If $\omega_1$ and $\omega_2$ are approximate distances to $\partial \Omega_1$ and $\partial \Omega_2$ boundaries, $\varphi$ can be represented by transfinite interpolation \[20\]:

$$\varphi = \frac{\omega_2 u|_{\partial \Omega_1} + \omega_1 u|_{\partial \Omega_2}}{\omega_1 + \omega_2} = \frac{0 \cdot \omega_2 + 1 \cdot \omega_1}{\omega_1 + \omega_2} = \frac{\omega_1}{\omega_1 + \omega_2}.$$
Application of the Ritz method results in an algebraic system $\mathbf{AC} = \mathbf{B}$ with matrix and vector elements given by the following expressions:

$$a_{lm} = \int_{\Omega} \nabla (\omega \chi_l) \cdot \nabla (\omega \chi_m) \, d\Omega;$$

$$b_l = -\int_{\Omega} \nabla (\omega \chi_l) \cdot \nabla \phi \, d\Omega + \int_{\Omega} f \omega \chi_l \, d\Omega.$$

Figures 21(a) and (b) present isolines of solutions computed using $50 \times 50$ and $500 \times 500$ uniform Cartesian grids of bicubic B-splines. Comparison of these plots shows that the coarse solution in Figure 21(a) definitely needs to be refined.

Given a candidate refinement window $\Omega_{wi}$ with area $A_i$, we employ an estimator that computes the averaged residual of the Poisson equation:

$$\sigma_i = A_i^{-1} \sqrt{\int_{\Omega_{wi}} (\nabla^2 u - f)^2 \, d\Omega}. \tag{29}$$

If the error is below the prescribed value, the candidate window is rejected; otherwise, the candidate window is employed with an additional set of basis functions at the desired resolution.

There are many strategies for generating candidate windows. Ideally, a suitable collection of candidate refinement windows should cover the original domain $\Omega$, or the portion of the domain where refinement is desired. For example, it is easy to generate candidate windows using hierarchical space decomposition. Starting with a bounding box that encloses the entire geometric domain, each consecutive refinement level is obtained by a subdivision of the refinement cells in half in each coordinate direction. Solution error, computed within each refinement cell (Figure 22(a)), drives placement of the refinement windows: they are allocated over only those cells in which the error exceeds the prescribed value. The red color in Figure 22(b)
Figure 23: (a) Solution of a Poisson equation after five refinement steps; (b) gradient of the function shown in Figure 23(a).

Figure 24: (a) Star-shaped refinement window. (b) Normalized distance field to boundary of star-shaped refinement window shown in Figure 24(a). (c) Window function that guarantees $C^2$ continuity of refinement function (only non-zero portion is shown).
depicts locations of the refinement windows at the different refinement levels. To avoid gradient distortions, such as shown in Figure 6(b), refinement windows are slightly dilated to overlap each other as illustrated in Figure 4(b). Refined solution and its gradient after five refinement steps are shown in Figure 23.

Alternatively, the neighboring refinement windows can be united into a single refinement window; a variety of construction methods can be used to construct the window function $W_i$ for the combined window, for example using R-functions. In fact, the proposed approach to multiresolution modeling places no restrictions on the shape of the refinement windows. As we showed in [23, 29, 24, 7], such functions with guaranteed differential properties can be constructed from most geometric representations. Figure 24(a) shows a star-shaped refinement window, and a plot of the normalized distance field is presented in Figure 24(b).

4 Conclusions

The proposed multiresolution refinement approach allows seamless merging of non-matching grids of basis functions, while preserving high order continuity of the solution and satisfying prescribed boundary conditions exactly. The described approach can be used with meshfree or mesh based methods, and independent types of basis functions may be associated with different refinement windows. Combined with a suitable strategy for sizing and placement of refinement windows, for example, using a hierarchical space decomposition described in Section 3, recursive application of refinement (1) constitutes an effective multiresolution tool.

Figure 25: (a) Unrefined (coarse) solution of the Poisson equation $\nabla^2 u = -10^4 \exp(-10^4((x-0.9)^2 + y^2 + z^2))$ with homogeneous Dirichlet boundary condition obtained using tricubic B-splines on a $31 \times 31 \times 31$ uniform Cartesian grid; (b) solution refined by using a $31 \times 31 \times 31$ grid of tricubic B-splines placed within a cubical refinement window defined by its extreme points $(0.65, -0.25, -0.25) - (1.15, 0.25, 0.25)$.

For simplicity, we demonstrated our multiresolution approach using two-dimensional examples. However, the proposed approach can be applied to refinement of three-dimensional solutions in a straightforward fashion. Figure 25 illustrates refinement of the solution to a Poisson equation $\nabla^2 u = -10^4 \exp(-10^4((x-0.9)^2 + y^2 + z^2))$ with homogeneous Dirichlet boundary condition. The coarse solution in Figure 25(a) was obtained using tricubic B-splines on a $31 \times 31 \times 31$ uniform Cartesian grid. To increase accuracy of the field, the refinement function was constructed as described in Section 2.2 using a $31 \times 31 \times 31$ grid of tricubic B-splines placed within a cubical refinement window defined by its extreme points $(0.65, -0.25, -0.25) - (1.15, 0.25, 0.25)$. Figure 25(b) shows a plot of the refined solution. The described method of refinement is broadly applicable to field simulation problems. In additional to the classical boundary value problems, it can be used effectively to deal with a variety of non-traditional field modeling tasks. Below we illustrate three examples of such problems.

Multiresolution distance field Figure 26(c) shows a plot of an approximate distance field obtained by a least square fitting of bicubic B-splines defined on $15 \times 15$ grid to samples of Euclidean distance shown
Figure 26: (a) Boundary of a geometric domain; (b) Euclidean distance to the boundary shown in Figure 26(a); (c) approximate distance field computed by a least square fitting of a $15 \times 15$ bicubic grid of B-splines to Euclidean distance shown in Figure 26(b); (d, e) refined approximate distance fields. Positions of the refinement windows are shown by dashed rectangles.

in Figure 26(b). Such fields are often constructed from scanned or sampled data and used for a variety of applications, including meshfree engineering analysis[7]. In this case, insufficient spatial resolution of the $15 \times 15$ B-spline grid caused noticeable deviation of the zero set of approximate distance field from the zero set of Euclidean distance. Local placement of the refinement windows as illustrated in Figures 26(d) and (e) increases geometric accuracy. The higher resolution of B-splines used for the refinement allows better approximation of sharp corners of the geometric boundary (Figure 26(a)). Application of the proposed multiresolution approach makes it possible to preserve the continuity of an approximate distance field up to the desired order, which is often critical.

Figure 27: (a) Computed Tomography (CT) scan of a human femur; (b) reconstructed bone density using bicubic B-splines on a $52 \times 10$ grid; (c) refined bone density using a $260 \times 50$ grid of bicubic B-splines.

Material properties of biological tissues The proposed adaptive refinement technique is useful for modeling of physical fields in biological tissues [6]. For stress/strain analysis in bones, patient specific
physical properties of the bone tissue have to be reconstructed from biomedical images such as, for example, the Computed Tomography (CT) scan of a human femur shown in Figure 27(a). Gray values in the CT scan correspond to different values of bone density. The bone density is initially approximated by a linear combination of bicubic B-splines defined over a $52 \times 10$ grid, with coefficients computed by a least square fit. However, because this grid of B-splines is too coarse it cannot provide an accurate density distribution (Figure 27(b)). Multiresolution refinement using a $260 \times 50$ grid of bicubic B-splines results in a more accurate representation of the bone density (Figure 27(c)).

Heat transfer in heterogeneous media

Figure 28 illustrates distribution of heat conductivity of a heterogeneous material. Plots of temperature distributions and magnitudes of their gradients are shown in Figure 29. Although temperature distributions look alike for coarse, fine and refined solutions, the magnitude of the gradient of the coarse solution (Figure 29(b)) is approximately one half of the magnitude of the gradient of more accurate solution shown in Figure 29(c). Application of the meshfree refinement procedure, described in this paper, recovers the temperature gradient as illustrated by the plot in Figure 29(f).

The described multiresolution procedure performs field refinement in one direction: from coarse to fine scale. It works well for linear problems, but refinement of solution to non-linear problems requires two-way communication among different refinement scales: refinement functions should enhance accuracy of the coarse solution within the refinement windows, and at the same time the coarse solution should also be updated outside of the refinement regions. It is straightforward to implement a feedback from fine to coarse scales: once refinement functions at the current refinement level are obtained, the coarse solution has to be recomputed using refinement functions as an enrichment. This iterative procedure would propagate the refinement information from the fine to a coarse scale and assure that the influence of the refinement functions will be accommodated by a coarse solution outside of the refinement zones.

Acknowledgments

This research is supported in part by the National Science Foundation grants DMI-0115133, CCR-0112758, DMI-0323514, and DMI-0621116 and the National Institute of Standards (NIST) grant 60NANB2D0126.

A Meshfree Method with Distance Fields

The meshfree method with distance fields stemmed from Kantorovich’s idea to satisfy Dirichlet boundary conditions [12]. Later, Rvachev and his students generalized Kantorovich’s approach for arbitrary boundary conditions [18, 19]. Over the years, the meshfree method with distance fields has become a mature, theoretically sound engineering analysis method that has been successfully applied to many engineering analysis...
Figure 29: (a) Temperature distribution computed using bicubic B-splines on a $50 \times 25$ grid. (b) Magnitude of gradient of temperature from Figure 29(a) (c) Temperature distribution computed using bicubic B-splines on a $500 \times 250$ grid. (d) Magnitude of gradient of temperature from Figure 29(c) (e) Refined temperature distribution. (f) Magnitude of gradient of the refined temperature from Figure 29(d)
problems including heat transfer, plate bending and vibrations, stress analysis, computational fluid dynamics and others. The salient feature of the method is exact treatment of boundary conditions without using spatial discretizations that conform to the shape of a geometric model. Clean separation of geometric information described by distances to the boundaries and analytic information represented by basis functions and boundary conditions enable complete automation of the solution procedure [29]. In this Section we describe the basic ideas of the method — representing fields by power series of the distances to the boundaries and interpolation of boundary conditions using distance fields.

A.1 Exact Treatment of Boundary Conditions Using Distance Fields

According to Rvachev [18, 19] any field distribution can be represented by a complete power series:

\begin{equation}
\begin{aligned}
  u &= u_0(0) + \sum_{k=1}^{m} \frac{1}{k!} \omega^k u_k(0) + O(\omega^{m+1}), \\
  \end{aligned}
\end{equation}

where \( \omega \) measures a distance (exactly or approximately) from a spatial point \( p \) to the boundary. Similar to the classic Taylor series [4], \( u_k(0) \) represent normal derivatives \( \frac{\partial^k u}{\partial n^k}, k = 1, \ldots, m \) specified on the domain’s boundary, where \( \omega \) takes on zero value. The remainder term \( O(\omega^{m+1}) \) assures completeness of the series. In our paper [19] we proved that it can be represented as a product of \((m+1)\)th degree of the distance \( \omega \) and polynomial \( \Phi \):

\begin{equation}
O(\omega^{m+1}) = \omega^{m+1} \Phi.
\end{equation}

We demonstrated [19] that the polynomial \( \Phi \) in the remainder term could be approximated by a complete linear combination of basis functions \( \Phi = \sum_{i=1}^{N} C_i \chi_i \). In the context of solving boundary value problems the remainder term provides sufficient degrees of freedom to approximate the differential equation of the problem. Expression (30) makes it possible to construct solutions of boundary value problems that satisfy prescribed boundary conditions exactly. Combining boundary conditions, distance to the boundary and basis functions, expression (30) represents a solution structure of a particular boundary value problem. Using a general solution structure (30), we can easily and systematically derive solution structures for different types of boundary conditions. In this Section we will explain derivation of solution structures for Dirichlet and convective boundary conditions.

**Solution structure for Dirichlet boundary condition**

\begin{equation}
  u|_{\partial \Omega} = \varphi
\end{equation}

was proposed by Kantorovich in [12]. He suggested to represent solution of a Dirichlet boundary value problem in the following form:

\begin{equation}
  u = \varphi + \omega \Phi.
\end{equation}

Since a Dirichlet boundary condition only prescribes the value of the solution on the domain’s boundary, the solution structure (31) consists of two terms: the function \( \varphi \) from the boundary condition and the remainder term \( \omega \Phi \). Because the distance field \( \omega \) vanishes on the boundary of a geometric domain, for any choice of the undetermined function \( \Phi \) the solution structure (31) satisfies the Dirichlet boundary condition exactly.

**Solution structure for convective boundary conditions**

\begin{equation}
  u = (\Phi_1 - \omega \nabla \omega \cdot \nabla \Phi_1) + (\psi - h \Phi_1) \omega + \omega^2 \Phi_2 + O(\omega^2)
\end{equation}

defines a family of functions that satisfy the following boundary condition:

\begin{equation}
  \left( \frac{\partial u}{\partial n} + hu \right) |_{\partial \Omega} = \psi.
\end{equation}
In order to represent the normal derivative of $u$ accurately, the distance field $\omega$ must be normalized to the first order. This means that $\omega$ has to have a unit gradient on the domain’s boundary $\partial \Omega$. References [18, 23, 3, 16] describe construction of normalized functions using $R$-functions. The first term $u_0$ in expression (32) represents a value of the function prescribed on the boundary $\partial \Omega$. Since the boundary condition (33) does not explicitly prescribe the value of $u$ on the boundary, $u_0$ is represented by a linear combination of basis functions $\Phi_1 = \sum_{i=1}^{n} C_i \Phi_1$ with undetermined coefficients $C_i$. Subtraction of $\omega \nabla \omega \cdot \nabla \Phi_1$ from $\Phi_1$ ensures that the first normal derivative of $u_0$ vanishes on the zero set of $\omega$ [18, 19]. The second term in expression (32) represents a first order normal derivative of $u$ whose value on the boundary $\partial \Omega$ is derived from the boundary condition (33): \[ \frac{\partial u}{\partial n} \bigg|_{\partial \Omega} = \psi - hu. \] The remainder term $\omega^2 \Phi_2$ guarantees completeness of $u$.

A.2 Transfinite Interpolation with Distances

Using distances to the boundaries it is possible to transfinitely interpolate functions and their derivatives prescribed on these boundaries. Our interpolation approach [20] is a generalization of inverse distance weighting interpolation, known as a Shepard’s method. Shepard used the method in [25] for interpolation of meteorological and geographical/geological data in 1968, but Watson [31] cites much earlier applications of the same technique dating as far back as 1920’s; Rvachev proposed a similar method for interpolating functions in 1967 [17]. In all cases, the interpolating function is constructed as a linear combination of function’s values $f_i$ at points $x$ with weight functions $W_i$:

$$f(x) = \sum_{i=1}^{n} f_i W_i(x),$$

where each weight function $W_i$ is inversely proportional to the distance from the point $x$, where the value $f_i$ is prescribed. In the case of transfinite interpolation [20], where functions and their derivatives are prescribed on the boundaries of a geometric model, the weight functions $W_i$ are inversely proportional to the distances to these boundaries.

Using inverse distance weighting, the functions $W_i, (i = 1, ..., n)$ are constructed by normalizing each inverse distance:

$$W_i(x) = \frac{\omega_i^{-\mu_i}(x)}{\sum_{j=1}^{n} \omega_j^{-\mu_i}(x)},$$

where $\omega_i(x)$ is a distance from a spatial point $x$ to the $i$th boundary. Expression (35) is inconvenient for evaluation on a computer because it leads to division by zero at boundary points. Such numerical problems are easily avoided by rewriting the weight functions in the equivalent but numerically stable form:

$$W_i(x) = \frac{\prod_{j=1,j \neq i}^{n} \omega_j^{\mu_j}(x)}{\sum_{k=1}^{n} \prod_{j=1,j \neq k}^{n} \omega_j^{\mu_j}(x)},$$

Properties of inverse distance weighting transfinite interpolation have been extensively studied by the authors [20]. As shown in [20] transfinite interpolation can be used to interpolate functions with their derivatives prescribed on different portions of the boundary. In the context of solving boundary value problems, this means that solution structure that satisfies mixed boundary conditions can be obtained by transfinite interpolation of individual solution structures. For example, if $u_1 = \varphi + \omega_1 \Phi_1$ is a solution structure satisfying the Dirichlet boundary condition given on the boundary $\partial \Omega_1$

$$u_1 \big|_{\partial \Omega_1} = \varphi,$$
and \( u_2 = \Phi_2 - \omega_2 \nabla \omega_2 \cdot \nabla \Phi_2 + (\psi - h\Phi_2) \omega_2 + \omega_2^2 \Phi_3 \) is a solution structure satisfying a convective boundary condition on boundary \( \partial \Omega_2 \)

\[
\left( \frac{\partial u_2}{\partial n} + hu_2 \right) \bigg|_{\partial \Omega_2} = \psi,
\]

then the solution structure that satisfies mixed boundary conditions

\[
u|_{\partial \Omega_1} = \varphi, \quad \left( \frac{\partial u}{\partial n} + hu \right) \bigg|_{\partial \Omega_2} = \psi
\]
is a transfinite interpolation of \( u_1 \) and \( u_2 \):

\[
u = u_1 \omega_2^2 + u_2 \omega_1 \omega_2 = (\varphi + \omega_1 \Phi_1) \frac{\omega_2^2}{\omega_1 + \omega_2} + (\Phi_2 - \omega_2 \nabla \omega_2 \cdot \nabla \Phi_2 + (\psi - h\Phi_2) \omega_2 + \omega_2^2 \Phi_3) \frac{\omega_1}{\omega_1 + \omega_2}.
\]

### A.3 Solution Method

Solution structures define sets of functions that satisfy prescribed boundary conditions exactly. In order to satisfy additional constraints one needs to determine functions \( \Phi_k \) in the solution structures. Since in most cases it is impossible to determine these functions exactly, \( \Phi_k \) are represented by a linear combination of basis functions:

\[
\Phi_k = \sum_{i=1}^{n} C_{ki} \chi_{ki}.
\]

In this case solution of a boundary value problem is translated into determination of the numerical values of the coefficients \( C_{ki} \) by any appropriate method.

Let us illustrate application of meshfree method with distance fields to solving the Poisson equation (3) with homogeneous Dirichlet boundary condition specified on the boundary of a circle with unit radius. The function \( f \) in equation (3) is given by the expression \( f = 10^4 \exp(-10^4(x^2 + y^2)) \). This function takes on the value of \( 10^4 \) at the origin and rapidly decays away of it.

According to [12, 18] we can represent a solution \( u \) as a product of two functions \( \omega \) and \( \Phi \):

\[
u = \omega \Phi.
\]

The function \( \omega \) is constructed in a such way that its zero set coincides with the boundary of the geometric domain [18, 23, 29]. For a circular domain with center at the origin and unit radius the function \( \omega \) can be written as follows: \( \omega = \frac{(1-x^2-y^2)}{2} \). Since the function \( \Phi \) in the solution structure cannot be determined exactly, it is represented by a linear combination of basis functions \( \Phi = \sum_{i=1}^{N} C_i \chi_i \). In order to find a solution that satisfies the differential equation of the problem we can apply any standard solution method. For example, to solve the boundary value problem (3) we will use the classical Ritz method that requires minimization of a quadratic functional:

\[
F = \int_\Omega u_0 \nabla^2 u_0 d\Omega - 2 \int_\Omega (-f) u_0 d\Omega.
\]

Substitution of \( u_0 \), given by expression (37), into the functional \( F \), application of the Gauss theorem to decrease the order of the derivatives, and differentiation of \( F \) with respect to the coefficients \( \{C_i\}_{i=1}^{N} \) results in a system of linear algebraic equations \( \mathbf{A} \mathbf{C} = \mathbf{B} \). Elements of the matrix \( \mathbf{A} \) and vector \( \mathbf{B} \) are computed as follows:

\[
a_{lm} = \int_\Omega \nabla (\omega \chi_l) \cdot \nabla (\omega \chi_m) d\Omega;
\]

\[
b_l = \int_\Omega (\omega \chi_l) 10^4 e^{-10^4(x^2+y^2)} d\Omega.
\]
When the algebraic system has been solved, numerical values of the coefficients \( \{C_i\}_{i=1}^N \) are substituted into the expression for \( \Phi \) which in turn is substituted into the solution structure (37). The latter can be evaluated and visualized. Figure 9(a) presents a plot of the solution obtained by using bicubic B-splines defined on a \( 30 \times 30 \) uniform Cartesian grid.

References


