Local defect correction for boundary integral equation methods

by

G. Kakuba, M.J.H. Anthonissen
Local Defect Correction for Boundary Integral Equation Methods

Department of Mathematics and Computer Science
Eindhoven University of Technology

CASA REPORT

by

G. Kakuba and M. J. H. Anthonissen

May 12, 2013
Abstract

This paper presents a new approach to gridding for problems with localised regions of high activity. The technique of local defect correction has been studied for other methods as finite difference methods and finite volume methods. In this paper we develop the technique for the boundary element method, an integral equation method. The technique offers an iterative way for obtaining the solution on an equivalent composite grid. It uses two grids: a global uniform coarse grid covering the whole boundary and a local fine grid covering the local active boundary. The solution of the local problem on the local fine grid is used to estimate the defect on the fine grid. The defect is then added to the right hand side of the global coarse grid problem which is then solved again to obtain a better updated solution. We demonstrate the technique's strength using an example and show that it offers a cheaper alternative to either solving on a global uniform grid or directly on a composite grid.

Keywords: integral equations, single layer potential, double layer potential, boundary elements, local error, defect, local activity, gridding.

1 Introduction

Often boundary value problems have small localised regions of high activity where the solution varies very rapidly compared to the rest of the domain. This behaviour maybe either due to boundary conditions or due to an irregular boundary. One therefore has to use relatively fine meshes to capture the high activity. Since the activity is localised, one may also choose to solve on a uniform structured grid. That is, instead of a uniform global grid, the solution is approximated using several uniform grids with different grid sizes that cover different parts of the domain. The size of each grid is chosen in agreement with the activity of the solution in that part of the domain. This refinement strategy is called local uniform grid refinement [Ferret and Reusken, 1996]. The solution is approximated on a composite grid which is the union of the various uniform local grids. One way of approximating this composite grid solution that is simple and less complex is by Local Defect Correction (LDC).

In LDC, at least one grid, the global coarse grid, covers the entire domain. Then a uniform local fine grid is used in a small part of the domain containing the high activity. In [Ferret and Reusken, 1996, Hackbusch, 1984], LDC has been shown to be a useful way of approximating the composite grid solution in which a global coarse grid solution is improved by a local fine grid solution through a process whereby the right hand side of the global coarse grid prob-
lem is corrected by the defect of a local fine grid approximation. This method has been well explored for numerical methods such as finite differences and finite volumes, see [Ferret and Reusken, 1996, Hackbusch, 1984, Anthonissen, 2001, Minero et al., 2006]. In this paper we explore potential analogues and develop an LDC strategy for boundary integral equation (BIE) methods, in particular, the boundary element method (BEM). The BEM is an increasingly popular numerical method for potential problems because it leads to a reduction in problem domain and provides very accurate solutions. However, although reduction in problem domain is guaranteed, the accuracy depends a lot on the mesh used [Guiggiani, 1990, Liapis, 1994]. Although a lot has been done for the finite element methods, BEM’s main competitor, comparatively little has been done on adaptive mesh refinement in BEM [Carsten and Stephan, 1995, Carsten and Stephan, 1996]. This paper therefore seeks to add to a gradually growing literature on mesh refinement in BEM. The initial attempts on LDC for BEM were in [Kakuba et al., 2006] and [Kakuba and Mattheij, 2007] where an algebraic approach was suggested and studied. However, in this paper we develop an integral approach to the basic steps of LDC. This approach should be preferred as it takes into account the integral properties of BEM. Since in BEM we discretise the boundary, we are concerned with problems in which the high activity occurs at the boundary.

One of the most important steps in adaptive refinement is error estimation. One way of estimating the error in collocation BEM is by using higher order interpolation at the elements to estimate the exact solutions and then use an appropriate norm of the difference between the BEM solution and the estimated exact solution to measure the error [Kita and Kamiya, 2001]. In this paper, however, for the sake of developing the LDC algorithm, we use simple examples whose exact solutions are known. Then we use the infinity norm $\| \cdot \|_\infty$ of the difference between the exact and the BEM solutions as the measure for the error.

The paper is organised as follows: First, we introduce the BIE and its discretisation using BEM in Section 2. In Section 3, we develop an LDC strategy for BEM alongside an example. In Section 4 we test the strategy on a typical example and discuss results. In Section 5 we give the advantages of the algorithm and finish in Section 6 with a summary of the concepts and results presented in the paper.
2 The BIE and BEM discretisation

The BEM results from a numerical discretisation of a BIE. Consider a domain \( \Omega \) with boundary \( \partial \Omega \) on which we have the following Laplace problem:

\[
\begin{align*}
\nabla^2 u(r) &= 0, \quad r \in \Omega, \\
u(r) &= g(r), \quad r \in \partial \Omega^1, \\
\frac{\partial u}{\partial n}(r) &= h(r), \quad r \in \partial \Omega^2,
\end{align*}
\]

where \( \partial \Omega^1 \cup \partial \Omega^2 = \partial \Omega \) and \( \partial \Omega^1 \cap \partial \Omega^2 = \emptyset \) and \( g \) and \( h \) are given functions, see Figure 1. If \( \partial \Omega^1 \equiv \partial \Omega \) we have a Dirichlet problem and if \( \partial \Omega^2 \equiv \partial \Omega \) we have a Neumann problem, otherwise we have a mixed problem.

2.1 Boundary Integral Equations

In this section we present integral relations for the potential \( u(s) \) at different locations of the domain \( \Omega \). These relations have been abundantly derived in literature and are readily available in various books on boundary element methods such as [Paris and Canas, 1997, Katsikadelis, 2002, Pozrikidis, 1992]. They are presented here for completeness. The relations are derived starting from the following Green's identity. Let \( \phi \), defined in \( \Omega \), and \( \psi \), defined in \( \Omega \times \Omega \), be two scalar functions which are twice continuously differentiable in \( \Omega \) and once on \( \partial \Omega \), then

\[
\int_{\Omega} (\phi(z)\nabla^2_z \psi(s;z) - \psi(s;z)\nabla^2 \phi(z)) \, d\Omega(z) = \int_{\partial \Omega} \left( \phi(\chi) \frac{\partial \psi}{\partial n}(s;r(\chi)) - \psi(s;r(\chi)) \frac{\partial \phi}{\partial n}(r(\chi)) \right) \, d\chi, \tag{2.2}
\]

where \( \chi \) is an arc length coordinate in \( \partial \Omega \). In (2.2), \( n \) is the outward unit normal at \( \partial \Omega \). To derive the integral relations for the Laplace equation, use is made of the identity (2.2). We take \( \phi \) as the unknown function \( u \) and \( \psi \) the fundamental solution. The fundamental solution of the Laplace equation is the solution of the singularly forced Laplace equation

\[
\nabla^2 v(s;r) + \delta(s;r) = 0, \quad s, r \in \Omega^\infty, \tag{2.3}
\]
where \( r \) is the variable field point, \( s \) is the fixed location of the source point or pole and \( \Omega^\infty \) denotes the infinite domain which is the whole plane in 2D. The subscript \( r \) on the operator \( \nabla \) means differentiation is with respect to \( r \). The Dirac delta distribution \( \delta(s;r) \) satisfies the following properties [Pozrikidis, 2002, Katsikadelis, 2002]:

\[
\int_{\Omega} \delta(s;r) \, d\Omega = \begin{cases} 
1, & \text{if } s \in \Omega, \\
0, & \text{if } s \notin \Omega.
\end{cases}
\] (2.4)

In 2D we have

\[
v(s;r) = \frac{1}{2\pi} \log \frac{1}{r}
\] (2.5)

where \( r = (x,y), \ s = (x_s,y_s) \) and \( r := \sqrt{(x-x_s)^2 + (y-y_s)^2} \) is the Euclidean distance from \( s \) to \( r \). Note that the fundamental solution \( v \) is symmetric, that is \( v(s;r) = v(r;s) \), also, \( v \) is singular at the point \( r = s \) and the integrals are computed as Cauchy principle value integrals. Taking these into consideration and using Green’s identity one obtains the following integral relation for the Laplace equation:

\[
c(s)u(s) = \int_{\partial\Omega} \left[ v(s;r(\chi)) \frac{\partial u}{\partial n}(r(\chi)) - u(r(\chi)) \frac{\partial v}{\partial n}(s;r(\chi)) \right] d\chi,
\] (2.6)

where the coefficient \( c(s) \) is given by

\[
c(s) := \begin{cases} 
1, & s \in \Omega, \\
\frac{\alpha(s)}{2\pi}, & s \in \partial\Omega, \\
0, & s \in \Omega^c,
\end{cases}
\] (2.7)

\( \alpha(s) \) is the internal angle at \( s \) and \( \Omega^c = \mathbb{R}^2 \setminus \Omega \) is the complement of \( \Omega \). We now introduce the following definitions of the single and double layer potentials:

\[
\mathcal{K}^s q(s) := \int_{\partial\Omega} v(s;r(\chi)) q(r(\chi)) d\chi,
\] (2.8a)

\[
\mathcal{K}^d u(s) := \int_{\partial\Omega} \frac{\partial v}{\partial n}(s;r(\chi)) u(r(\chi)) d\chi.
\] (2.8b)

These integrals are called single and double layer potentials respectively by making an analogy with the corresponding boundary distributions of electric charges and charge dipoles in electrostatics, see [Atkinson, 1997, Kabara, 2005, Kress, 1989, Pozrikidis, 2002, Schull and Steinbach, 2000]. Using (2.8), the integral equation (2.6) can be written as

\[
(cI + \mathcal{K}^d)u(s) = \mathcal{K}^s q(s).
\] (2.9)
The operators $K_s$ and $K_d$ are called the single and double layer operator respectively and $I$ is the identity operator. When $s$ is located at the boundary the integral equation is referred to as a boundary integral equation. The functions

$$k_s(s; r) := v(s; r),$$

(2.10a)

$$k_d(s; r) := \frac{\partial v}{\partial n}(s; r),$$

(2.10b)

are called the kernel of the integral operators $K_s$ and $K_d$ respectively.

### 2.2 The Boundary Element Method

The integral equation (2.6) expresses the value of the potential $u$ at any point $s$ in terms of its values and normal derivative at the boundary. A discretisation of this equation leads to the Boundary Element Method (BEM) system of algebraic equations. To this end, the physical boundary $\partial \Omega$ is partitioned into $N$ parts $\partial \Omega_j$, $j = 1, 2, \ldots, N$, see Figure 2.

![Figure 2: A discretisation of an ellipse into $N = 9$ elements.](image)

Each physical partition $\partial \Omega_j$ is represented by a numerical partition $\Gamma_j$. The union $\Gamma := \bigcup_{j=1}^{N} \Gamma_j$ is what is called a numerical boundary. For $\Gamma_j$ we use rectilinear elements in which the ends of each partition $\partial \Omega_j$ are connected by a straight line of length $L_j$ to be called a boundary element, here simply called element and $L_j$ the element size. A grid that has the same element size $L$ for all the elements is called a uniform grid of size $L$. We consider here constant elements which means that the functions $u$ and $q$ are assumed to be constant on each element. An element is represented by a single node placed at the midpoint of the element. The element containing the $j$-th node is denoted $\Gamma_j$. The BIE is
then applied at the $i$-th node, that is $s = r_i$, and integration over $\partial \Omega$ is estimated by the sum of the integrations on all the $N$ elements. Thus we have,

$$c(r_i)u(r_i) + \sum_{j=1}^{N} \int_{r_i} u(r(\chi)) \frac{\partial v}{\partial n}(r_i;r(\chi)) \, d\chi = \sum_{j=1}^{N} \int_{r_i} q(r(\chi)) v(r_i;r(\chi)) \, d\chi, \quad (2.11)$$

where nodes $r_i$, $i = 1, 2, \ldots, N$, are the midpoints of the elements, see Figure 3.

The functions $u$ and $q$ are assumed to be constant on each element and equal to their nodal values. Let us introduce the definitions

$$u_j := u(r_j), \quad q_j := q(r_j), \quad (2.12)$$

where $r_j$ is the midpoint of $\Gamma_j$. Now (2.11) gives

$$c_i u_i + \sum_{j=1}^{N} u_j \int_{r_i} \frac{\partial v}{\partial n}(r_i;r(\chi)) \, d\chi = \sum_{j=1}^{N} q_j \int_{r_i} v(r_i;r(\chi)) \, d\chi, \quad (2.13)$$

for all the collocation points $i = 1, 2, \ldots, N$. Define

$$\hat{H}_{ij} := \int_{r_i} \frac{\partial v}{\partial n}(r_i;r(\chi)) \, d\chi, \quad \text{and} \quad G_{ij} := \int_{r_i} v(r_i;r(\chi)) \, d\chi. \quad (2.14)$$

With this notation, equation (2.13) becomes

$$c_i u_i + \sum_{j=1}^{N} \hat{H}_{ij} u_j = \sum_{j=1}^{N} G_{ij} q_j. \quad (2.15)$$

Setting

$$H_{ij} := c_i \delta_{ij} + \hat{H}_{ij}, \quad (2.16)$$
where \(\delta_{ij}\) is the Kronecker delta. (2.15) can be written as

\[
\sum_{j=1}^{N} H_{ij} u_j = \sum_{j=1}^{N} G_{ij} q_j, \tag{2.17}
\]

Let \(H\) and \(G\) be \(N \times N\) matrices whose elements are given by (2.16) and (2.14) respectively. Also introduce the following vectors of length \(N\):

\[
\mathbf{u} := (u_1, u_2, \ldots, u_N)^T, \quad \mathbf{q} := (q_1, q_2, \ldots, q_N)^T. \tag{2.18}
\]

Then applying equation (2.17) for all \(r_i\) yields

\[
Hu = Gq. \tag{2.19}
\]

In (2.19) we have a system of \(N\) algebraic equations in \(2N\) unknowns \(u_j\) and \(q_j\), \(j = 1, 2, \ldots, N\), which, as it is now, is an underdetermined system. The extra \(N\) relations needed to solve the system uniquely must come from the boundary conditions. For each element \(j\), either \(u_j\) or \(q_j\) is known through boundary conditions.

For generality, let us assume a part \(\partial \Omega^1\) of the boundary on which \(u(r)\) is given and a part \(\partial \Omega^2\) on which \(q(r)\) is given. Let these two parts be discretised into \(N_1\) and \(N_2\) constant elements respectively such that \(N_1 + N_2 = N\). Thus, there are still \(N\) unknowns, \(N_2\) values of \(u(r)\) on \(\partial \Omega^2\) and \(N_1\) values of \(q(r)\) on \(\partial \Omega^1\), which are to be determined from the system (2.19). Before solving the system, the unknown quantities need to be separated from the known quantities. To this end, partition the matrices \(H\) and \(G\) and write (2.19) as

\[
\begin{pmatrix} H_1 & H_2 \end{pmatrix} \begin{pmatrix} \hat{u}_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} G_1 & G_2 \end{pmatrix} \begin{pmatrix} q_1 \\ q_2 \end{pmatrix}, \tag{2.20}
\]

where \(\hat{u}_1\) and \(\hat{q}_2\) denote the known quantities on \(\Gamma^1\) (representing \(\partial \Omega^1\)) and \(\Gamma^2\) (representing \(\partial \Omega^2\)) respectively. Then carry out the multiplications and move all the unknowns to the left hand side of the equation to obtain

\[
Ax = b \tag{2.21}
\]

where

\[
A := (-G_1 \ H_2), \quad x := \begin{pmatrix} q_1 \\ u_2 \end{pmatrix}, \quad b := -H_1 \hat{u}_1 + G_2 \hat{q}_2. \tag{2.22}
\]

The solution of the system (2.21) gives a BEM approximation of the unknowns in \(x\) in the grid nodes \(r_i\). We denote by \(x^L\) a BEM approximation on a grid of size \(L\). Thus \(u^L_i\) (or \(q^L_i\)) is a BEM approximation of \(u_i\) (or \(q_i\)) using a grid of size \(L\). Solving (2.21) gives the unknown boundary quantities of \(u\) and \(q\). Therefore we now have all the boundary quantities. The solution \(u(r)\) can then be computed at any point \(r \in \Omega\) using (2.13) with \(c(r) = 1\).
3 Local Defect Correction

Consider the Neumann problem (3.1) whose domain is a unit square in two dimension. That is,

\[
\begin{align*}
\nabla^2 u(r) &= 0, \quad r \in \Omega := [0, 1] \times [0, 1], \\
q(r) &= h(r), \quad r \in \Gamma,
\end{align*}
\]

where

\[
h(r) = \frac{(r - r_s) \cdot n(r)}{||r - r_s||^2}, \quad r_s = (0.5, -0.02)^T.
\]  (3.2)

The Neumann problem (3.1) results in a Fredholm integral equation of the first kind [Atkinson, 1997, Hackbusch, 1995, Kress, 1989]. It is singular and hence has no unique solution [Atkinson, 1967]. To ensure a unique solution in the discretised problem, a value of \(u\) is prescribed in one of the nodes [Chen and Zhou, 1992, Strese, 1984]. In the implementations in this paper, the value of \(u\) in the first element node is prescribed. This will also help us compare the numerical solutions directly with the exact solutions for error measurement. The solution in \(\Omega\), shown in Figure 4, has a small area close to the boundary where it changes rapidly. As a result, the solution \(u(r)\) in the boundary has a region of high activity in a small part of the boundary, see Figure 4. Therefore we can identify a small region inside \(\Omega\) which contains the high activity. This region we call the local domain and denote it by \(\Omega_{\text{local}}\), see Figure 5. Its boundary \(\Gamma_{\text{local}}\), which we will call the local boundary, consists of two parts: a part \(\Gamma_{\text{active}}\) that is also part of the global boundary and a part \(\Gamma_{\text{inside}}\) that is contained in the global domain \(\Omega\), Figure 5b. We will call the part \(\Gamma_{\text{active}}\) the local active boundary. For instance, in the problem corresponding to the solutions shown in Figure 4, the

![Figure 4: Solution in the domain and solution at the boundary](image)

(a) Solution in \(\Omega\) with small region of high activity.  
(b) The solution \(u(r)\) in part of \(\partial \Omega\) that borders the high activity.
boundary $\Gamma_{\text{active}}$, may be identified as $\Gamma_{\text{active}} = \{(x, y) : y = 0, x \in [0.2, 0.8]\}$. The part of the global boundary $\Gamma$ that is outside the active region $\Gamma_{\text{active}}$ will be denoted $\Gamma_c$, that is, $\Gamma_c := \Gamma \setminus \Gamma_{\text{active}}$.

(a) Identify local high activity area $\Omega_{\text{local}}$.

(b) A local domain $\Omega_{\text{local}}$.

Figure 5: An example of a multiscaled solution with localised high activity in Figure 5a and, in Figure 5b, an illustration of a local problem domain. The boundary of $\Omega_{\text{local}}$ is $\Gamma_{\text{local}} := \Gamma_{\text{active}} \cup \Gamma_{\text{inside}}$.

The interest in BEM is to compute a numerical approximation of $u(r)$ at the boundary as accurately as possible. For such kind of multiscaled variations one is faced with the option of using a global uniform grid with a mesh of relatively small size $l$ in order to capture the high activity. This would result in very large systems which are computationally expensive since BEM matrices are full matrices. Besides, outside the local active boundary $\Gamma_{\text{active}}$, the variation of the solution is smooth and a relatively coarse grid would suffice. The other option is to use a uniform structured grid designed to capture the different activities. This would be a composite grid with a relatively fine mesh of size $l$ in the local active region and a coarse grid of size $L$ elsewhere.

With LDC we approximate the solution on a composite grid in an iterative way that involves solving a so called local problem which is a boundary value problem defined on the local domain. The local problem is solved on a fine mesh whose size is chosen in agreement with the local activity. The solution on the local fine grid is combined with the solution on the global coarse grid through defect correction to obtain a composite grid solution on $\Gamma$. The advantage of this approach is that instead of solving a large composite grid system, two smaller systems; a global coarse grid system and a local fine grid system, are solved independently. For problems with various local activities the local problems can be solved separately in parallel giving a tremendously cheaper way of obtaining a composite grid solution other than solving directly on the composite grid.

As introduced in Section 2.2, let $\Gamma$ be the numerical representation of $\partial \Omega$ in BEM. The global coarse grid $\Gamma^k$ is a uniform mesh of $N$ elements each of size $L$.
covering the whole of $\Gamma$, that is,
\[ \Gamma^l := \{ \Gamma^l_1, \Gamma^l_2, \ldots, \Gamma^l_N \} \tag{3.3} \]
where $|\Gamma^l_j| = l$ for all $j$. The *local fine grid* $\Gamma^l_{\text{local}}$ is a uniform mesh of $N_{\text{local}}$ elements each of size $l$ covering $\Gamma_{\text{local}}$, that is,
\[ \Gamma^l_{\text{local}} := \{ \Gamma^l_{\text{local},1}, \Gamma^l_{\text{local},2}, \ldots, \Gamma^l_{\text{local},N_{\text{local}}} \} \tag{3.4} \]
where $|\Gamma^l_{\text{local},i}| = l$ for all $i = 1, 2, \ldots, N_{\text{local}}$. The size of the local fine grid $l$

is chosen in agreement with the activity of the solution in $\Gamma_{\text{active}}$. Since the solution varies much more rapidly in $\Gamma_{\text{active}}$ than elsewhere, we expect $l$ to be much smaller than $L$. Part of the grid $\Gamma^l_{\text{local}}$ belongs to $\Gamma_{\text{active}}$ and part belongs to $\Gamma_{\text{inside}}$. The part that belongs to $\Gamma_{\text{active}}$ is denoted $\Gamma^l_{\text{active}}$ and that that belongs to $\Gamma_{\text{inside}}$ is denoted $\Gamma^l_{\text{inside}}$. That is
\[ \Gamma^l_{\text{active}} := \{ \Gamma^l_{\text{active},1}, \Gamma^l_{\text{active},2}, \ldots, \Gamma^l_{\text{active},N_{\text{active}}} \} \tag{3.5a} \]
\[ \Gamma^l_{\text{inside}} := \{ \Gamma^l_{\text{inside},1}, \Gamma^l_{\text{inside},2}, \ldots, \Gamma^l_{\text{inside},N_{\text{inside}}} \} \tag{3.5b} \]
where $\Gamma^l_{\text{active}} \cup \Gamma^l_{\text{inside}} = \Gamma^l_{\text{local}}$ and $N_{\text{active}} + N_{\text{inside}} = N_{\text{local}}$. In constant elements that we discuss here, the collocation nodes are the midpoints of the elements, where the solution is computed. Let us denote the set of nodes of the coarse grid as $r^l$,
\[ r^l := \{ r^l_1, r^l_2, \ldots, r^l_N \}. \tag{3.6} \]
Similarly we denote the set of nodes of the local fine grid as $r^l_{\text{local}}$,
\[ r^l_{\text{local}} := \{ r^l_{\text{local},1}, r^l_{\text{local},2}, \ldots, r^l_{\text{local},N_{\text{local}}} \}. \tag{3.7} \]
The sets \( r_{\text{active}}^l \) and \( r_{\text{inside}}^l \) are defined similarly. We assume that all the grid nodes of \( r^l \cap r_{\text{active}}^l \) belong to \( r_{\text{active}}^l \), see Figure 6. The composite grid nodes \( r_{\text{active}}^{l,l} \) are the union \( r^l \cup r_{\text{active}}^l \) of the global coarse grid nodes \( r^l \) and the active local fine grid nodes \( r_{\text{active}}^l \). The composite grid \( \Gamma_{\text{active}}^{l,l} \) consists of the finest elements that correspond to \( r_{\text{active}}^{l,l} \).

First we discretise the BIE on \( \Gamma^l \) to yield

\[
\frac{1}{2} u^l + \sum_{j=1}^{N} u_j^l \int_{r_i^l} \frac{\partial v}{\partial n}(r_i; r(\chi)) \, d\chi = \sum_{j=1}^{N} \int_{r_i^l} q(r(\chi)) v(r_i; r(\chi)) \, d\chi, \tag{3.8}
\]

which gives the initial global coarse grid system of equations

\[
A^l u_0^l = b^l. \tag{3.9}
\]

Once we have solved (3.9), the next step is to use the initial global coarse grid solution \( u_0^l \) to formulate a local problem on \( \Omega_{\text{local}} \). This local problem on \( \Omega_{\text{local}} \) satisfies the same operator as in the global problem. The boundary conditions on \( \Gamma_{\text{active}} \) are the same as those in the global problem, that is, \( q(r) = h(r) \), since \( \Gamma_{\text{active}} \subseteq \Gamma \). On \( \Gamma_{\text{inside}} \) we prescribe an artificial boundary condition \( \tilde{g}(r) \) defined below. So we have

\[
\begin{cases}
\nabla^2 u(r) = 0, & r \in \Omega_{\text{local}}, \\
q(r) = h(r), & r \in \Gamma_{\text{active}}, \\
u(r) = \tilde{g}(r), & r \in \Gamma_{\text{inside}}.
\end{cases} \tag{3.10}
\]

The function \( \tilde{g}(r) \) is piecewise constant on \( \Gamma_{\text{inside}} \) and takes on values of \( u_{\text{inside}}(r_i) \) where \( r_i \) is a node of \( \Gamma_{\text{inside},i} \), that is,

\[
\tilde{g}(r) := u_{\text{inside}}(r_i), & r \in \Gamma_{\text{inside},i} \subset \Gamma_{\text{inside}}. \tag{3.11}
\]

To compute \( u_{\text{inside}}(r_i) \) we use the relation (2.13) with \( c_i = 1 \) since \( r_i \in \Omega \),

\[
u_{\text{inside}}(r_i) := \sum_{j=1}^{N} \int_{r_i^l} q(r(\chi)) v(r_i; r(\chi)) \, d\chi - \sum_{j=1}^{N} u_j^l \int_{r_i^l} \frac{\partial v}{\partial n}(r_i; r(\chi)) \, d\chi, & r_i \in \Gamma_{\text{inside}}. \tag{3.12}\]

Equation (3.12) means that we use the solution of the initial global coarse grid problem to obtain artificial Dirichlet boundary conditions at \( \Gamma_{\text{inside}} \). Since at \( \Gamma_{\text{active}} \) \( q \) is known, the local problem is mixed and the BIE for (3.10) is, for \( r, r(\chi) \in \Gamma_{\text{local}} \),

\[
\frac{1}{2} u(r) + \int_{\Gamma_{\text{active}}} u(r(\chi)) \frac{\partial v}{\partial n}(r; r(\chi)) \, d\chi + \int_{\Gamma_{\text{inside}}} \tilde{g}(r(\chi)) \frac{\partial v}{\partial n}(r; r(\chi)) \, d\chi = \int_{\Gamma_{\text{active}}} q(r(\chi)) v(r; r(\chi)) \, d\chi + \int_{\Gamma_{\text{inside}}} q(r(\chi)) v(r; r(\chi)) \, d\chi. \tag{3.13}
\]
Discretising (3.13) on the local fine grid defined in (3.4) and (3.5) we have

\[
\frac{1}{2} u_{\text{local},i} + \sum_j u_{\text{active},j} \int_{\Gamma_{\text{active},j}} \frac{\partial \nu}{\partial n}(r_i; r(\chi)) \, d\chi + \sum_j u_{\text{inside},j} \int_{\Gamma_{\text{inside},j}} \frac{\partial \nu}{\partial n}(r_i; r(\chi)) \, d\chi = \\
\sum_j q_{\text{active},j} \int_{\Gamma_{\text{active},j}} \nu(r_i; r(\chi)) \, d\chi + \sum_j q_{\text{inside},j} \int_{\Gamma_{\text{inside},j}} \nu(r; r(\chi)) \, d\chi. \tag{3.14}
\]

In (3.14) we have two vectors on \(\Gamma_{\text{local}}\): \(u_{\text{local}}\) and \(q_{\text{local}}\), where

\[
u_{\text{local}} = \begin{bmatrix} u_{\text{active}} \\ u_{\text{inside}} \end{bmatrix}, \quad q_{\text{local}} = \begin{bmatrix} q_{\text{active}} \\ q_{\text{inside}} \end{bmatrix}. \tag{3.15}
\]

The vector \(u_{\text{inside}}\) is known through (3.12) and the vector \(q_{\text{active}}\) is known because \(q(r(\chi))\) is given on \(\Gamma_{\text{active}}\). So if we repeat (3.14) for all the local nodes we obtain an algebraic system of \(N_{\text{local}}\) equations. We rearrange the system in matrix form by putting the known quantities on one side to obtain the initial local problem algebraic system of equations

\[
A_{\text{local}}^l \mathbf{x}_{\text{local}}^l = b_{\text{local}}^l \tag{3.16}
\]

where

\[
\mathbf{x}_{\text{local}}^l = \begin{bmatrix} u_{\text{active}}^l \\ q_{\text{inside}}^l \end{bmatrix}.
\]

The solution \(u_{\text{active}}^l\) is expected to be more accurate than the coarse grid solution \(u_{j}^l\) in \(\Gamma_{\text{active}}\). The next step of LDC is to use the local fine grid solution to update the global coarse grid problem. In updating, the right hand side of the global coarse grid problem is corrected by the defect of the local fine grid approximation, we will call this step the *defect correction* step. The two approximations are then used to define a composite grid approximation of \(u(r)\). The question now is: how do we compute the defect?

Consider the coarse grid discretisation (3.8). If we knew the exact continuous function \(u(r)\) and hence the exact solution \(u_i := u(r_i)\) in the nodes we would use it in (3.8) to obtain

\[
\frac{1}{2} u_i + \sum_{j=1}^{N} u_j \int_{\Omega_j} \frac{\partial \nu}{\partial n}(r_i; r(\chi)) \, d\chi = \sum_{j=1}^{N} q(r(\chi)) \nu(r_i; r(\chi)) \, d\chi + d_i^l. \tag{3.17}
\]

where \(d_i^l\) is the local defect for the \(i\)-th equation. We also have the exact BIE as

\[
\frac{1}{2} u_i + \sum_{j=1}^{N} u(r(\chi)) \frac{\partial \nu}{\partial n}(r_i; r(\chi)) \, d\chi = \sum_{j=1}^{N} q(r(\chi)) \nu(r_i; r(\chi)) \, d\chi. \tag{3.18}
\]
Subtracting (3.18) from (3.17) gives

\[
\sum_{j=1}^{N} u_i \int_{\Gamma_j^\delta} \frac{\partial v}{\partial n}(r_i; r(\chi)) \, d\chi - \sum_{j=1}^{N} \int_{\delta \Omega_j} u(r(\chi)) \frac{\partial v}{\partial n}(r_i; r(\chi)) \, d\chi = d_i^\delta. \tag{3.19}
\]

From (3.19) we define the local defect per element \(j\) as

\[
d_{ij} := u_i \int_{\Gamma_j^\delta} \frac{\partial v}{\partial n}(r_i; r(\chi)) \, d\chi - \int_{\delta \Omega_j} u(r(\chi)) \frac{\partial v}{\partial n}(r_i; r(\chi)) \, d\chi, \tag{3.20}
\]

so that the total defect at \(r_i\) is given by

\[
d_i^\delta := \sum_j d_{ij}, \quad i = 1, 2, \ldots, N. \tag{3.21}
\]

Therefore if we know the exact continuous function \(u(r)\) we can compute the local defect \(d_{ij}^\delta\), add it to the right hand side of (3.8) and solve for the exact solution \(u_i\) on each element. However \(u(r)\) is not known and therefore we cannot compute the defect using (3.20). All we can do is estimate \(d_{ij}^\delta\) as accurately as possible using the best solution available, which is

\[
u_{\text{best},i} = \begin{cases} u_i^\delta, & \Gamma_j^\delta \subset \Gamma_e, \\ u_{\text{active},i}, & \Gamma_j^\delta \subset \Gamma_{\text{active}}. \end{cases} \tag{3.22}
\]

So for elements in the high activity region we have the fine grid solution which we can use to estimate the local defect as follows.

In the case of a square, \(\Gamma_j^\delta = \delta \Omega_j\). Suppose that in the local fine grid \(\Gamma_{\text{active}}^j\) a global coarse grid element \(\Gamma_j^\delta\) is divided into \(k\) fine elements \(\Gamma_{\text{active},jk}^j\), such that \(\Gamma_j^\delta = \bigcup_k \Gamma_{\text{active},jk}^j\), see an illustration in Figure 7 for \(k = 3\). Then the best approximations of the integrals in (3.20) are

\[
u_i \int_{\Gamma_j^\delta} \frac{\partial v}{\partial n}(r_i; r(\chi)) \, d\chi \approx u_{\text{active},i} \int_{\Gamma_j^\delta} \frac{\partial v}{\partial n}(r_i; r(\chi)) \, d\chi, \tag{3.23a}
\]

\[
\int_{\delta \Omega_j} u(r(\chi)) \frac{\partial v}{\partial n}(r_i; r(\chi)) \, d\chi \approx \sum_k u_{\text{active},jk} \int_{\Gamma_{\text{active},jk}^j} \frac{\partial v}{\partial n}(r_i; r(\chi)) \, d\chi. \tag{3.23b}
\]

Therefore, using the initial fine grid solution, we have the following best approximation of the initial defect per element

\[
d_{ij}^0 \approx u_{\text{active},i} \int_{\Gamma_j^\delta} \frac{\partial v}{\partial n}(r_i; r(\chi)) \, d\chi - \sum_k u_{\text{active},jk} \int_{\Gamma_{\text{active},jk}^j} \frac{\partial v}{\partial n}(r_i; r(\chi)) \, d\chi, \tag{3.24}
\]

for \(\Gamma_j^\delta \subset \Gamma_{\text{active}}\) and

\[
d_{ij}^0 \approx 0.
\]
for $\Gamma^L_j \subset \Gamma_c$. We can then compute the defect
\begin{equation}
\mathbf{d}_0^i \approx \sum_j \mathbf{d}_{0ij}, \quad \Gamma^L_j \subset \Gamma_{active}, \quad \text{for all } i = 1, 2, \ldots, N. \tag{3.25}
\end{equation}

By default, integration in the BIE is global. Each node of the global coarse grid communicates with the active region through integration. So although the activity is local, it affects the entire system of equations. The defect $\mathbf{d}_0^i$ is therefore computed for all nodes of the global coarse grid to generate the local defect vector
\begin{equation}
\mathbf{d}_0^i := (d_{01}^i, d_{02}^i, \ldots, d_{0N}^i)^T. \tag{3.26}
\end{equation}

The next step now is the updating step. The global coarse grid discretisation is updated with the defect of the local fine grid solution. So we have
\begin{equation}
\mathbf{A}^L \mathbf{u}^L_1 = \mathbf{b}^L + \mathbf{d}_0^L. \tag{3.27}
\end{equation}

Solving (3.27) gives the updated coarse grid solution $\mathbf{u}^L_1$. At this stage we use the fine grid solution on $\Gamma^L_{active}$ and the global coarse grid solution to form a composite grid solution $\mathbf{u}_1^{L,L}$ as
\begin{equation}
\mathbf{u}_{0,j}^{1,1}(r) = \begin{cases}
\mathbf{u}^L_{active,j}(r), & r \in \Gamma_{active}, \\
\mathbf{u}^L_1(r), & r \in \Gamma_c.
\end{cases} \tag{3.28}
\end{equation}

The composite grid solution (3.28) can now be used to compute better boundary conditions on $\Gamma_{inside}$ and then form and solve the updated fine grid problem
\begin{equation}
\mathbf{A}_{local} \mathbf{x}_{local}^{1,1} = \mathbf{b}_{local}^{1,1}. \tag{3.29}
\end{equation}

Then we obtain the updated composite grid solution given by
\begin{equation}
\mathbf{u}_{1,1}^{1,1}(r) = \begin{cases}
\mathbf{u}_{active}^1(r), & r \in \Gamma_{active}, \\
\mathbf{u}_1^1(r), & r \in \Gamma_c.
\end{cases} \tag{3.30}
\end{equation}
This step marks the end of one complete cycle of the LDC algorithm. The iteration process is summarised in the following algorithm:

**BEM-LDC Algorithm**

(i) **Initialisation**
- Solve the global coarse grid system
  \[ A^1 u^1_L = b^1_0 \]
- Solve the initial fine grid system
  \[ A^l x^1_{local} = b^1_0 \]
- Compute the initial defect
  \[ d^1_L \]

(ii) For \( i = 1, 2, \ldots \)
- Solve for the updated course grid solution \( u^i_L \) in
  \[ A^i u^i_L = b^1 + d^{i-1}_L. \]
- Solve for the updated fine grid solution \( x^i_{active} \) in
  \[ A^i_{local} x^i_{local} = b^1_{local}. \]
- Form the updated composite grid solution \( u^i_{L+1} \) and compute the new defect \( d^i_L \).

4 **Examples and results**

The LDC procedure above has been used to solve the problem (3.1). The results are shown in Figure 8 and Figure 9. The figures show the solution on the side \( y = 0 \) of the unit square. The course grid used is of size \( L = 0.2 \) and the fine grid is of size \( l = 0.2/9 \). Figure 8 shows how the initial results compare with the exact solution (the solid line), the initial coarse grid solution in Figure 8(a) and the initial fine grid solution in Figure 8(b). Figure 9 shows the results after the first update, the updated coarse grid solution in Figure 9(a) is better than the initial one in Figure 8(a) and the updated local fine grid solution is better than the initial one in Figure 8(b). Figure 10 shows how fast the global error decreases. Basically the algorithm has converged already in the first iteration since the error reduction between successive iterations after the first one is much smaller compared to that in the first iteration.
Figure 8: Results of a typical LDC process for a Neumann problem in one iteration. The solid line is the exact solution.

Figure 9: Results of a typical LDC process for a Neumann problem in one iteration. The solid line is the exact solution.
In brief, the LDC iterative process involves the following: solve the global coarse grid problem on $\Gamma$, compute $u$ on $\Gamma_{\text{inside}}$, solve a fine grid problem on $\Gamma_{\text{local}}$ and then update the global coarse grid problem.

Suppose we have $p$ locally active small regions and thus $p$ local problems. Let, for each local problem, $M^l$ be the number of elements on $\Gamma_{\text{local}}$ and $M^m$ the number of elements on $\Gamma_{\text{inside}}$. For instance in the illustration in Figure 6, $M^l = 3$ and $M^m = 9$. We can increase $M^l$ without necessarily increasing $M^m$ since the activity is on $\Gamma_{\text{local}}$. Let $M^m$ be so small compared to $M^l$ that the size of the local problem is $M \approx M^l$. Let $N$ be the size of the global problem and $N^l_{\text{local}}$ the number of global elements in $\Gamma_{\text{local}}$. We assume $\Gamma_{\text{local}}$ is such a small part of the global boundary that $N - N^l_{\text{local}} \approx N$. Then the equivalent composite grid on $\Gamma$ would be of size $pM + N$. The operational count for LU-decomposition is $N^3/3$ for a size $N$ matrix. So the complexity of the equivalent composite grid problem would be

$$\frac{1}{3}(pM + N)^3 \approx \frac{(p + 1)^3}{3}N^3 \text{ if } M \approx N.$$  \hfill (5.1)

The BEM-LDC algorithm converges in one step which involves solving two coarse grid problems and $p$ local problems and so has total complexity

$$2 \cdot \frac{1}{3}N^3 + \frac{p}{3}M^3 \approx \frac{N^3}{3}(2 + p).$$  \hfill (5.2)

So when we compare (5.1) with (5.2) we see that the composite problem is $(p + 1)^3/(2 + p)$ times more expensive than BEM-LDC. Suppose instead we were
to refine globally to a grid of size equal to that of the local problems. Then if the refinement ratio is say $\alpha$, that is, $L/l = \alpha$, the resulting problem would be of complexity $\frac{1}{3} \alpha^3 N^3$. So the resulting problem would be a factor $\alpha^3/(p + 2)$ times more complex than using LDC. For instance in the modest case of $\alpha = 2$, this factor is more than one for up to $p = 5$ local problems. Thus BEM-LDC is cheaper than both of its obvious alternatives of either composite gridding or refining uniformly.

Another advantage of LDC over solving on direct composite or fine uniform grids is the memory required. LDC requires less memory than the equivalent composite or uniform grid problems. This is because instead of handling large matrices and vectors of sizes say $(M+N)$ or $\alpha N$, it handles smaller matrices of sizes $N$ and $M$ at a time.

6 Conclusions

We have presented the technique of Local Defect Correction for BEM, a technique for solving problems with high local activity in the boundary using BEM. This technique offers an alternative to solving directly on a composite grid or a uniform fine grid both of which would result in large matrices that are more expensive than using LDC. In previous works a similar technique was suggested but that depended on the decomposition of the global system matrix itself, a strategy that did not take into account the integral properties of the BIE as is done here. What is also interesting to note is that one iteration of the algorithm suffices.

References


## Previous Publications in This Series:

<table>
<thead>
<tr>
<th>Number</th>
<th>Author(s)</th>
<th>Title</th>
<th>Month</th>
</tr>
</thead>
<tbody>
<tr>
<td>13-07</td>
<td>M.E. Hochstenbach</td>
<td>Probabilistic upper bounds for the matrix two-norm</td>
<td>Apr. ’13</td>
</tr>
<tr>
<td>13-08</td>
<td>M.E. Hochstenbach, D.A. Singer, P.F. Zachlin</td>
<td>Numerical approximation of the field of values of the inverse of a large matrix</td>
<td>Apr. ’13</td>
</tr>
<tr>
<td>13-09</td>
<td>I. Zisis, B.J. van der Linden, C.G. Giannopapa</td>
<td>Towards a smoothed particle hydrodynamics algorithm for shocks through layered materials</td>
<td>May ’13</td>
</tr>
<tr>
<td>13-10</td>
<td>G. Kakuba, M.J.H. Anthonissen</td>
<td>Local defect correction for boundary integral equation methods</td>
<td>May ’13</td>
</tr>
</tbody>
</table>