An improved DRM representation of partial derivatives

B. Natalini¹, V. Popov²
¹Facultad de Ingeniería, Universidad Nacional del Nordeste, Argentina
²Wessex Institute of Technology, United Kingdom

Abstract

The dual reciprocity method (DRM) is one of the most efficient boundary element (BEM) procedures to take the domain integrals to the boundary. The DRM when applied to advection-diffusion problems requires an algorithm to represent the partial derivatives introduced by the convective term. The classical procedure expresses them as a function of the potentials employing an approximation function.

The Dual Reciprocity Method Multy-Domain (DRM-MD) is a technique which combines DRM and domain decomposition for a case when the domain is subdivided in a very large number of subregions and the resulting internal mesh pattern looks like a finite element grid. In the DRM-MD the partial derivatives can be represented either in the classical way or as functions of the normal derivatives.

In this paper a set of examples of advection-diffusion problems which have been solved using DRM-MD codes with both formulations in first partial derivatives are presented. It is shown that the classical formulation produces large numerical errors when the approximation function is \(1 + r\), which can be removed with the new approach, and that the classical formulation provides suitable results when the approximation function is the augmented thin plate spline.

1 Introduction

Boundary elements method (BEM) when applied to problems governed by linear non-homogeneous differential equations or non-linear and time dependent
problems requires evaluation of domain integrals. In early boundary element
analysis the evaluation of domain integrals was done using cell integration, a
technique which, whilst effective and general, made the method lose its
boundary only nature introducing an additional internal discretisation.

Dual reciprocity method (DRM), a technique introduced by Nardini &
Brebbia [1], is one of the most effective procedures to take domain integrals to
the boundary. The DRM uses the concept of particular solutions. It applies the
divergence theorem to the domain integral terms and converts the domain
integrals into equivalent boundary integrals.

Let us consider the linear non-homogeneous differential equation defined in
the domain \( \Omega \) which is enclosed by the contour \( \Gamma \)

\[
\nabla^2 u(x) = b(x,u(x), \frac{\partial u(x)}{\partial x_i})
\]

where:
- \( u \) is a scalar field (potential function)
- \( b \) is the non-homogeneous term
- \( x \) is a position vector in the domain with components \( x_i \).

In order to solve the equation (1) the DRM approximation of the non-
homogeneous term by means of an interpolation function is employed. Although
a variety of functions can in principle be used as an interpolation function,
previous work [2] has shown that good performances were obtained in a variety
of cases with simple expansions, the most popular of which is \( 1+r \), where \( r \)
is the distance between the point \( x \) and a set of field points. In the classical
approach, the partial derivatives in \( b \) are approximated by means of partial
derivatives of the interpolation function used. But Zhu & Zhang [3] have shown
that when performing partial derivatives of the \( 1+r \) interpolation function, the
classical way of handling partial derivatives, the procedure introduces artificial
singularities, as a result of which large numerical errors are produced.

A major problem encountered with the DRM is that the resulting algebraic
system consists of a series of matrix multiplications of fully populated matrices.
When only few internal points are required in the DRM, the resulting computing
time is in general smaller than the one required by the cell integration scheme
but still being costly in comparison with domain approaches. Popov and Power
[4] proposed a technique to overcome this problem and they called it the Dual
Reciprocity Method Multi-Domain (DRM-MD). This technique combines DRM
and domain decomposition for a case when the domain is subdivided in a very
large number of subregions and the resulting internal mesh pattern looks like a
finite element grid.

An important feature of the DRM-MD is that it offers the possibility to
represent the spatial partial derivative as a function of the normal derivatives
instead of approximating it using an interpolation function [5]. In this way any
artificial singularity related to the partial derivatives of the interpolation function
is removed. In this paper this technique is reviewed and results from a set of
DRM-MD codes in which both formulations for first partial derivatives have been implemented are presented.

2 Basic concepts

2.1 Dual Reciprocity Method

The application of the Green’s integral representation formulae to the eqn (1) yields

\[ \lambda(x)u(x) + \int_{\Gamma} q^*(x, y)u(y)d\Gamma_y - \int_{\Omega} u^*(x, y)q(y)d\Omega_y = \int_{\Omega} u^*(x, y)b(y)d\Omega_y \]  

where

\[ q(y) = \frac{\partial u(y)}{\partial n} \] is the normal derivative of the potential

\[ u^*(x, y) = u^*(r) \] is the fundamental solution of the Laplace equation, being \( r = |y - x| \) the distance between the source point \( x \) and the collocation point \( y \)

\[ q^*(x, y) = \frac{\partial u^*(x, y)}{\partial n} \] is the normal derivative of the fundamental solution

\[ b(y) = b(y, u(y), \frac{\partial u(y)}{\partial x_i}). \]

The main idea of the DRM is to approximate \( b \) by a function of the form

\[ b(y) \equiv \sum_{k=1}^{n+m} f(y, z_k)\alpha(z_k) \]  

where

\( f(y, z_k) \) are interpolation functions (i.e., \( f = 1 + r \))

\( \alpha(z_k) \) are unknown coefficients

\( z_k \) are field points.

The functions \( f \) depend only on the geometry of the problem. The approximation is done by means of an arbitrary number of collocation points \( z_k \), usually \( n \) boundary points and \( m \) interior points called DRM nodes. Furthermore, a condition is imposed on \( f \)

\[ \nabla^2 \hat{u}(y, z_k) = f(y, z_k) \]  

where \( \hat{u}(y, z_k) \) is a particular solution of eqn (1).
Replacing \( b(y) \) in eqn (1) by its approximation given by (3) leads to the system of equations

\[
Hu - Gq = (\hat{H}u - \hat{G}q)F^{-1}b
\]

where

- \( H \) and \( G \) are the BEM matrices
- \( \hat{U}, \hat{Q} \) and \( F \) are geometry-only-dependent DRM matrices
- \( u \) and \( q \) are vectors containing the potentials and the normal derivatives respectively
- \( b \) is a vector containing the values of the non-homogeneous term.

### 2.2 Dual Reciprocity Method – Multi Domain

Let us consider the case when the domain, \( \Omega \), of the problem is subdivided into two subregions.

![Figure 1: Domain subdivided in two subregions.](image)

By applying the equation (5) to each subdomain two systems of linear equations are obtained, each one having a fully populated matrix of coefficients. Moreover, as the nodes in the interface introduce more unknowns than equations, because no boundary conditions are defined there, every system has more unknowns than equations. However, when both systems are joined into a single one, and the so-called matching conditions are applied, the system of equations becomes closed.

Matching conditions establish that at every node at the interface

a) The value of the potential is the same for both subdomains

\[
u_i(x_k) = u_2(x_i)
\]

b) The value of the physical flux is the same for both subdomains but with different sign because of the orientation of the normal vectors in the points \( i \) and \( k \)

\[
\Phi_1(u_k, q_k) = -\Phi_2(u_i, q_i)
\]
the form of the function $\Phi$ depending on the physical problem under analysis.

In others words, the node $k$ of $\Omega_1$ introduces two unknowns, the potential and the normal derivative, but contributes only with one equation. On the other hand the point $\gamma$ of $\Omega_2$, which is the same node but seen from $\Omega_2$, introduces another two unknowns and one equation. Finally the matching conditions (6) and (7) provide another two equations that close the system.

From the point of view of the system of equations, (5) when applied to $\Omega_1$ and $\Omega_2$ will produce systems of the form

$$
\begin{bmatrix}
A_1^1 & A_1^0 \\
A_1^0 & A_1^1
\end{bmatrix}
\begin{bmatrix}
p_1^1 \\
p_1^1
\end{bmatrix} = \begin{bmatrix}
b_1^1 \\
b_1^1
\end{bmatrix} \quad (8)
$$

and

$$
\begin{bmatrix}
A_2^2 & A_2^1 \\
A_2^1 & A_2^2
\end{bmatrix}
\begin{bmatrix}
p_2^2 \\
p_2^2
\end{bmatrix} = \begin{bmatrix}
b_2^2 \\
b_2^2
\end{bmatrix} \quad (9)
$$

where $p_1^1$ and $p_2^2$ are unknown nodal potentials and derivatives at the boundaries $\Gamma^1$ and $\Gamma^2$ respectively; and $p_1^1$ and $p_2^1$ are unknown nodal potential and derivatives at the interface $\Gamma_1$. Once the matching conditions are applied, the two vectors of unknowns at the interface, $p_1^1$ and $p_2^1$, can be reduced to a single-one, and the two systems can be assembled together, as:

$$
\begin{bmatrix}
A_1^1 & A_1^0 & 0 \\
A_1^0 & A_1^1 & A_1^2 \\
0 & A_2^1 & A_2^2
\end{bmatrix}
\begin{bmatrix}
p_1^1 \\
p_1^1 \\
p_1^2
\end{bmatrix} = \begin{bmatrix}
b_1^1 \\
b_1^1 \\
b_2^2
\end{bmatrix} \quad (10)
$$

The DRM-MD consists of applying this technique of subdivision of the domain in the limiting case when the resulting internal mesh looks like a finite element grid. The system (10) shows clearly the process by which a system with a sparse matrix is produced from systems with fully populated matrices; and the higher the number of subdomains considered, the higher the trend towards a sparse system. Accordingly, a resulting linear system of equation with sparse matrix of coefficients is characteristic in this approach.

2.3 Treatment of partial derivatives: classical formulation

In the vector $b$ in (5) an algorithm is established to express partial derivatives as functions of either $u$ or $q$. 
In the classical approach the starting point is to express the potential at a point \( \mathbf{x} \) in terms of the approximation function \( f \), in a similar way as it was done for \( b \) in eqn (3)

\[
u(\mathbf{x}) \approx \sum_{k=1}^{n+m} f(\mathbf{x}, \mathbf{z}_k) \beta(\mathbf{z}_k)
\]

or expressed in matrix form

\[
u = \mathbf{F}\beta
\]

where the \( \mathbf{F} \) matrix is the same one that is used in (5).

Differentiation of eqn (12) produces

\[
\frac{\partial \mathbf{u}}{\partial x_i} = \frac{\partial \mathbf{F}}{\partial x_i} \beta.
\]

After re-writing eqn (12) as \( \beta = \mathbf{F}^{-1}\mathbf{u} \), eqn (13) becomes

\[
\frac{\partial \mathbf{u}}{\partial x_i} = \frac{\partial \mathbf{F}}{\partial x_i} \mathbf{F}^{-1}\mathbf{u}.
\]

As the diagonal elements of the \( \frac{\partial \mathbf{F}}{\partial x_i} \) matrix are undefined when \( f = 1 + r \), eqn (14) introduces singularities that lead to large numerical errors. Zhu & Zhang [3] proposed a functional transformation to avoid these singularities. Another possibility to avoid this problem is to use a different interpolation function that does not introduce artificial singularities with differentiation. The DRM-MD offers a third option that consists of expressing the partial derivatives through the normal derivatives that appear in eqn (2).

### 2.4 Treatment of partial derivatives: new formulation

To introduce the proposed algorithm let us consider a quadrilateral 2-D DRM-MD subregion like the one shown in Fig. 2

![Fig. 2: A quadrilateral 2-D DRM-MD subregion with linear elements.](image)
At every corner node there will be three variables, some of which could be given as the boundary conditions: the potential, $u$, the normal derivative before the node, $q_b$, and the normal derivative after the node, $q_a$. Every normal derivative is a scalar product of the gradient, $\nabla u_i$, and the outward unitary vector normal to the boundary, that is

$$q_{ib} = \nabla u_i \cdot n_{ib}$$  \hspace{1cm} (15)$$

and

$$q_{ia} = \nabla u_i \cdot n_{ia}.$$  \hspace{1cm} (16)$$

Eqns (15) and (16) can be re-written as

$$q_{ib} = \frac{\partial u_i}{\partial x_i} n_{1ib} + \frac{\partial u_i}{\partial x_{2i}} n_{2ib}$$  \hspace{1cm} (17)$$

$$q_{ia} = \frac{\partial u_i}{\partial x_i} n_{1ia} + \frac{\partial u_i}{\partial x_{2j}} n_{2ia}.$$  \hspace{1cm} (18)$$

These two equations form a linear system that yields:

$$\frac{\partial u_i}{\partial x_i} = \frac{(q_{ib} n_{2ia} - q_{ia} n_{2ib})}{J_i}$$  \hspace{1cm} (19)$$

and

$$\frac{\partial u_i}{\partial x_{2i}} = \frac{(q_{ia} n_{1ib} - q_{ib} n_{1ia})}{J_i}$$  \hspace{1cm} (20)$$

where

$$J_i = n_{1ib} n_{2ia} - n_{1ia} n_{2ib}.$$  \hspace{1cm} (21)$$

Eqns (19) to (21) define a formulation of partial derivatives as functions of normal derivatives. This formulation has two main advantages in respect to the one given by equation (14). Firstly, it does not introduce new approximations and secondly it involves less number of operations. This algorithm can be implemented in the corner nodes only, and therefore for the DRM nodes, which are located in the interior of the domain, the classical approach must be applied. It is evident that this approach offers substantial benefits when the proportion of corner nodes is high, that is, in the DRM-MD meshes.

3 Numerical examples

Four DRM-MD codes to solve 2-D advection-diffusion problems were compared. The first and second ones had the classical formulation in first partial derivatives and used $1 + r$ and the Augmented Thin Plate Spline (ATPS), see
Golberg & Chen [6], as interpolation functions $f$, respectively. The third and fourth ones used the new formulation in first partial derivatives and $1 + r$ and ATPS as interpolation functions, respectively. These codes solve the equation

$$\nabla^2 u - \vec{V} \cdot \nabla u - ku = 0$$

(22)

where

- $\vec{V}$ is the vector of flow velocity
- $k$ is the reaction constant.

The codes were applied to a 1-D problem with velocity

$$V = \frac{1}{L} \ln \left( \frac{U_1}{U_0} \right) + k \left( x - \frac{L}{2} \right)$$

(23)

and boundary conditions $U_0 = u(x = 0)$; $U_1 = u(x = L)$. Its analytical solution is given by

$$u = U_0 \exp \left[ \left( \frac{kx^2}{2} \right) + \left[ \frac{1}{L} \ln \left( \frac{U_1}{U_0} \right) - L \frac{k}{2} \right] x \right].$$

(24)

To produce equivalent 1-D results from the 2-D codes, rectangular domains of length $L$ in the $x$ direction and width $W$ in the $y$ direction were used. Geometry and boundary condition were symmetric in respect to the $x$ axis, that is $U_0$ and $U_1$ were constant and

$$V_x = V, \quad V_y = 0$$

(25)

and

$$\frac{\partial u}{\partial n} \bigg|_{y=W/2} = 0, \quad \frac{\partial u}{\partial n} \bigg|_{y=-W/2} = 0.$$

(26)

The domain was subdivided in rectangular subdomains and linear elements were used within each subdomain. Fig. 3 shows the analytical solution for the potential and the derivative for three different cases.

Figures 5 to 10 show the error in the potential and the normal derivative using the codes 2 to 4 under different combination of reaction constant and domain width. Two meshes were used: 4x20 (that is 4 subdivisions in the $y$ direction and 20 subdivision in the $x$ direction), and 2x80. In some examples a DRM node was placed at the centre of every subdomain. The maximum value of the errors of the first code are given in the figure caption because they are one or two orders of magnitude bigger than those of the other codes.
Fig. 4: Analytical solution of the examples tested.

Fig. 5: Error distribution of potential and normal derivatives for $L=1$, $W=0.2$, $k=10$ ($V_{\text{max}}=-8.4$), $U_0=300$, $U_1=10$, mesh of 4x20 subregions and DRM nodes included. The maximum errors for the code 1 were 49% and 327% for potential and derivatives respectively.
Fig. 6: Error distribution of potential and normal derivatives for $L=1$, $W=0.2$, $k=20$ ($V_{\max} = -13.4$), $U_0=300$, $U_1=10$, mesh of 4x20 subregions and DRM nodes included. The maximum errors for the code 1 were 84% and 290% for potential and derivatives respectively.

Fig. 7: Error distribution of potential and normal derivatives for $L=1$, $W=0.2$, $k=40$ ($V_{\max} = -23.4$), $U_0=300$, $U_1=10$, mesh of 4x20 subregions and DRM nodes included. The maximum errors for the code 1 were 200% and 590% for potential and derivatives respectively.
The maximum errors for the code 1 were 80% and 547% for potential and derivatives respectively.

**Fig. 8:** Error distribution of potential and normal derivatives for $L=1$, $W=0.2$, $k=10$ ($V_{x_{\max}} = -8.4$), $U_0=300$, $U_f=10$, mesh of 4x20 without DRM nodes.

The maximum errors for the code 1 were 80% and 547% for potential and derivatives respectively.

**Fig. 9:** Error distribution of potential and normal derivatives for $L=1$, $W=0.2$, $k=20$ ($V_{x_{\max}} = -13.4$), $U_0=300$, $U_f=10$, mesh of 4x20 without DRM nodes. The maximum errors for the code 1 were 141% and 922% for potential and derivatives respectively.
From the examples shown, it can be seen that the results from codes 2 to 4 are of the same order of magnitude.

It is clear that using $1 + r$ combined with the classical formulation in first partial derivatives is not suitable at all. When the new formulation is applied to this case it produces a reduction of the error of one and two orders of magnitude, which means that although $1 + r$ gives an acceptable approximation of a particular solution of eqn (1), it becomes very inaccurate when approximating the partial derivatives through (14). By contrast, the approximation (14) provides a good estimation of partial derivatives when using ATPS.

On the other hand the new formulation when combined with ATPS does not produce an improvement of the results. Perhaps the reason is that the new formulation calculates the partial derivatives through a function of the normal derivatives, which are, in the examples shown above, calculated with lower accuracy than the potentials.

The figures show that the magnitude of the errors increase as $k$ increases. Though the errors seem to be large, the variation in potential and its derivatives are of such nature that they would produce significant difficulties for any numerical scheme. However, the errors can be reduced by refining the mesh, see Fig. 6 and 10, which verifies the convergence of the method. A comparison of Fig. 5 and 8 and Fig. 6 and 9 show that the results improve when DRM nodes are used. This last observation is not surprising, because the presence of DRM nodes
improves the ability of the approximation function to represent a particular solution of the eqn (1).

4 Conclusions

The treatment of the partial derivatives appearing in the convective term of the advection-diffusion equation in DRM-MD codes has been reviewed and results from a set of cases has been presented.

It has been shown that the algorithm given by (17) and (18) (new formulation) to represent partial derivatives as a function of normal derivatives removes the main drawback to the use of the DRM-MD codes with $1 + r$ as approximation function, which introduces artificial singularities in the convective term [3]. In this case improvements of one and two orders of magnitude were observed. There was no noticeable improvement when applying the new formulation to codes using ATPS as approximation function, which suggests that accurate representation of first order partial derivatives can be obtained by using (14) and ATPS.

References