The Method of Meshless Fundamental Solutions with Sources at Infinity

Peter Mitic
Positive Corporation Ltd., Hampshire, U.K.
peter.mitic@virgin.net
Youssef F. Rashed
Wessex Institute of Technology, Southampton, U.K.
youssef@wessex.ac.uk

1 Abstract

The method of external source collocation is used to solve a discretised boundary value problem, $\nabla^2 U=0$, where *U* is the potential in a two-dimensional simply-connected region *D*, subject to a mixture of Neumann and Dirichlet boundary conditions. Numerical analysis has, to date, been hindered by an accumulation of round-off error, which has made it impossible to investigate accuracy of the Meshless Fundamental Solutions method unless sources are near the boundary. Symbolic analysis allows a full investigation of ill-conditioned systems in which sources can be placed "at infinity". This analysis provides an indication of how many sources must be used and where they should be placed.

2 Introduction

Solutions to the equation $\nabla^2 U=0$, where *U* is the potential in a two-dimensional simply-connected region *D*, with *U* or $Q=\nabla U$ specified on the boundary, are often impractical or impossible to obtain analytically if *D* is not a simple shape, or if the boundary conditions are complex. In these cases, the usual approach is to discretise the boundary into segments and apply numerical techniques. Severe ill-conditioning results if many sources are used or if some of them are far from the boundary. In this paper we consider few sources with a simple boundary, and use symbolic analysis to avoid accumulating round-off error.

3 Outline and Analysis of the Method

Figure 1 shows a general case of such a region, *D*, with boundary ∂D and $\nabla^2 U=0$ in D. The boundary is discretised into *Z* elements such as (B_{i1}, B_{i2}) with midpoint m_j. *D* is surrounded by *W* external sources, $S_1, S_2, \ldots S_W$, each with potential given by a fundamental solution:

$$U(n_{ij}) = \log(n_{ij}) /(2\pi) \qquad \text{......................(1)}$$

where n_{ij} is the distance from m_j to S_i. (See [1], [2] or [3])

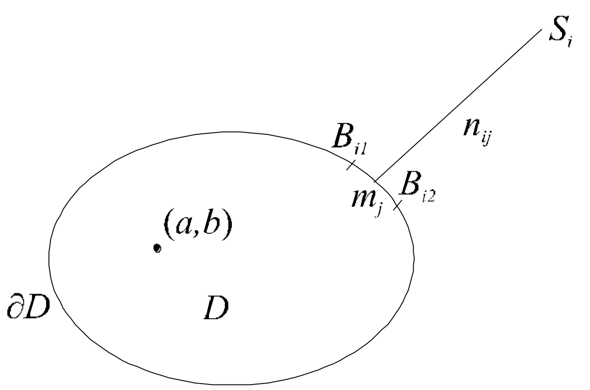

Figure 1

The justification for the Meshless Method may be found in any discussion of the Method of Fundamental Solutions.[4] is a detailed example.The main principles are:

- A source *S*, external to the boundary, gives rise to a potential at a given point *X* on or inside the boundary.
- The potential is given by the fundamental solution for the problem which depends on the problem geometry and the physical characteristics of the materials concerned.
- Several sources, $S_1 ... S^n$, external to the boundary, give rise to different potentials at the same point.
- The actual potential at *X* is a linear combination of the potentials due to the sources concerned, but boundary conditions must be satisfied.

The number of sources need not be the same as the number of boundary elements, but there must be enough to 'represent' all boundary conditions. If there are too few, the solutions obtained may be wrong. If there are too many, the correct linear combination of sources will be determinable by a least squares (or equivalent) process. The method of solution is described briefly in [5]. The details that appear here are relevant to the ensuing discussion of convergence and ill-conditioning, and are necessarily more detailed than described in [5]. The *Mathematica* implementation is coded in package `Meshless.m`.

4 Symbolic Analysis of a Simple Problem

Only the simplest of BEM problems may be analysed symbolically or with rational arithmetic. Computation in more complex problems is too slow. Hence, we use the *Brebbia Square* problem, described in [6] and [7], with four sources. Figure 2 shows the *Brebbia Square*, for which $\nabla^2 U=0$ in the region *D*, and boundary values for *U* and *Q* are as shown. Each of four sources is a distance *x* from the boundary.

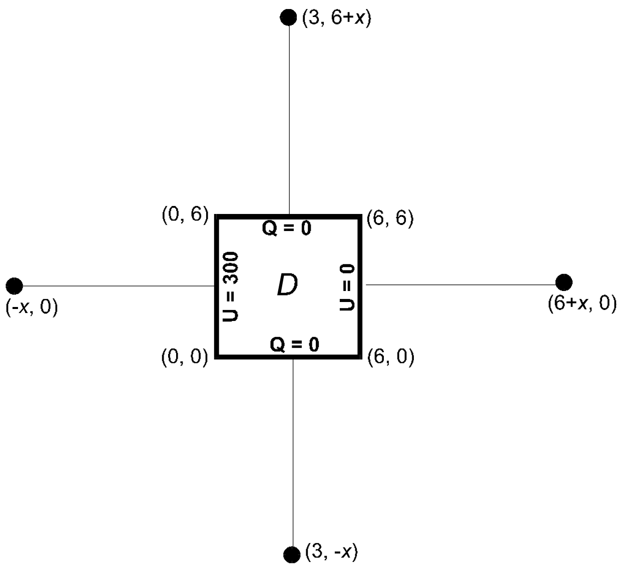

Figure 2

The Constant Element BE analysis in [6] and [7] yields the following values for the potential *U*, and fluxes Q_x and Q_y at the interior points *P*.

P	*U*	*Qx*	*Qy*
(2,2)	200.28	-50.303	-0.14976
(2,4)	200.28	-50.303	0.14976
(3,3)	150.01	-50.215	-0.00004
(4,2)	99.74	-50.306	0.14564
(4,4)	99.74	-50.306	-0.14564

Table 1

4.1 Overall implementation in Mathematica

The most important reason for using *Mathematica* is to allow analysis of extremely ill-conditioned systems. The source code is also particularly compact. Package `Meshless.m` exports the function `MeshlessBoundarySolutionMatrix`, which does the bulk of the computation. It implements a BEM and returns the computed BE matrix.

4.2 Detailed implementation in Mathematica

The *Mathematica* inputs, (collectively equation 2), for this problem are below. They define the boundary and sources, compute the BE midpoints, define the boundary conditions and their type and compute the required results.

```
BoundaryEndPointsNumber = 4;
BoundaryEndPoints = {{6, 0}, {6, 6}, {0, 6}, {0, 0}};

Sources2[x_] := {{6 + x, 3}, {3, 6 + x}, {-x, 3}, {3, -x}}
BoundaryElements = Append[
  Map[Part[BoundaryEndPoints, {#, # + 1}] &, Table[i, {i, 1, BoundaryEndPointsNumber - 1}]], {Last[BoundaryEndPoints], First[BoundaryEndPoints]}];
BoundaryNodes = Map[MidPoint, BoundaryElements];

UQ = {0, 0, 300, 0};
UQType = {0, 1, 0, 1};
MboundarySolM =
MeshlessBoundarySolutionMatrix[BoundaryElements, UQ,  UQType, Sources2[x]] // Simplify // N

DetMBoundarySolM = Det[MBoundarySolM];
Table[DetMBoundarySolM, {x, 1, 100, 5}]
Plot[DetMBoundarySolM, {x, 1, 100}]
```

........................(2)

The output corresponding to MboundarySolM is a complicated 4-by-4 matrix. A plot of its determinant against x demonstrates progressive ill-conditioning of the matrix. When $x \sim 100$, DetMBoundarySolM ~ 10^{-8}. The following inputs, (collectively equation 3), compute the potential, (solPab), and x- and y- fluxes, (solUab), at an interior point (a,b) of the square.

```
MBoundarySolution[x_] :=
  MeshlessBoundarySolution[BoundaryElements, UQ, UQType, Sources2[x],
  "UseInverse"] // Simplify // N
MBoundarySolution[x];

InteriorPoints = {{a, b}};
solPab = First[intSolPotential[MBoundarySolution[x],
      InteriorPoints, x]]
solUab = intSolFlux[MBoundarySolution[x], InteriorPoints, x]
```

........................(3)

The unpalatable results solPab and solUab form the basis for further analysis.

4.3 Limiting expressions when x~0 and x→∞

Taking the limit in the outputs from (3) as $y \to \infty$ should yield a simple function for the potential and flux at (a,b). The results are inconclusive because *Mathematica* displays a 'division by zero' error. Despite this, the correct potential, $U(a,b)=300-50a$, and fluxes, $Ux=-50$, $Uy=0$, are obtained. These results show that the limiting potential is independent of b and that the limiting flux is independent of a and b. This is confirmed by numerical examples.

Further numerical trials show that it is not, in general, safe to place the sources too close to the boundary. The result is grossly inaccurate for $x<20$, but can be improved by increasing the number of sources. If more sources are used, symbolic computations fail due to insufficient memory. Symbolic analysis is therefore quite limited.

4.4 Condition Number of the Boundary Element matrix

The Condition Number (CN) of the boundary element matrix, (MboundarySolM in equation 2), provides some insight into how ill-conditioning varies with the distance, *x*, of the sources from the boundary. *Mathematica* provides the function SingularValues, which implements Singular Value Decomposition. Using it to compute the CN is therefore easy:

```
ConditionNumber[n_]:=
 Module[{mbs, usv},
         mbs = MBoundarySolutionMatrix[x]  /. {x->n};
         usv =  SingularValues[mbs] [[2]];
         Max[usv]/Min[usv] ]
```

.........................(4)

Figure 3 shows that the CN increases approximately quadratically as *x* increases. The largest CN, (x = 100), corresponds to a BE matrix determinant of the order of 10^{-9}, which is something that C++ or Fortran can cope with. Problems arise with more sources: the columns of the resulting BE matrices represent nearly-parallel vectors. For example, determinants of the order of 10^{-195} occur if 20 sources are used with x = 100. In such cases, obtaining equivalent results using C++ or Fortran is not possible due to severe round-off error.

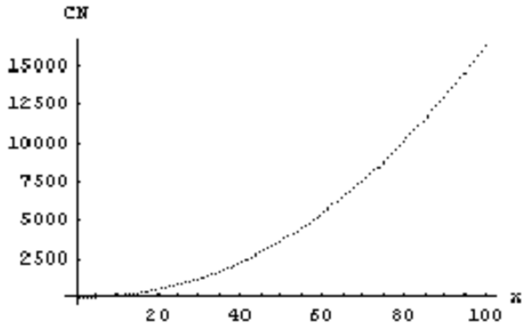

Figure 3

5 Numerical Investigations

The variables in a numerical analysis of this problem are the number of sources, the number of boundary elements and source placement. The latter comprises the distance of the sources from the boundary and their positions relative to each other. In order to mirror the symbolic analysis of the previous section, we use 4 sources, each a variable distance *x* from the boundary. Each side of the square has 4 boundary elements. Table 2 shows the results of calculations of potential and flux at (2,4) as *x* varies from 10 to 100 (‘infinity’!) in steps of 10.

x	Potential	x□Flux	y□Flux
10	199.031	□49.0782	□0.558438
20	199.663	□49.7132	□0.186076
30	199.834	□49.8615	□0.0911756
40	199.901	□49.9186	□0.053867
50	199.935	□49.9465	□0.0355091
60	199.954	□49.9622	□0.0251509
70	199.966	□49.9718	□0.0187411
80	199.973	□49.9782	□0.0145017
90	199.979	□49.9826	□0.0115532
100	199.983	□49.9859	□0.00942013

Table 2

These results show that the computed solutions converge to the exact solutions, (200, -50 and 0), and that convergence is fast. Therefore, in order to compute a 'solution', it is sufficient to compute one solution using a sufficiently large value for *x*. The results in Table 2 indicate that 'sufficiently large' means, in practice, 5 to 10 times the dimension of the boundary. This gives a maximum error of 2.5% in the *y*-flux. For comparison, using $x = 500$ gives {199.999, -49.9994, -0.000395234} for the potential, *x*-flux and *y*-flux respectively. Although this appears to be an improvement, when more sources and/or boundary elements are used ill-conditioning limits further improvement in accuracy.

6 Discussion

The limiting process of the previous section shows that it is possible to obtain an accurate symbolic solution for the *Brebbia Square* problem using only a few sources. This is advantageous because more sources increase ill-conditioning effects. *Mathematica*'s Limit function does not work smoothly in all cases we have tried, but the results are easily backed up by numerical calculations. This method is also inherently efficient because the prerequisites for calculating the potential and flux at an interior point are determinable without knowing the co-ordinates of the interior point. 'Traditional' boundary element methods need to know target point co-ordinates in advance in order to compute results, and the computation must be reformulated to accommodate new target points. Numerical computations execute much faster than part symbolic computations, and it is sufficient to place sources sufficiently far from the boundary to achieve a given accuracy. Ill-conditioning effects can be reduced by using a minimal number of sources and boundary elements.

We have done many other numerical investigations, varying the number of sources, the number of boundary elements and source placement. The main outcome of these experiments is that the actual configuration of sources is immaterial: the shape profile of the sources need not be like the shape of the boundary.

7 Conclusion

The analysis in this paper demonstrates that 'sources at infinity' is a viable configuration for the method of Meshless Fundamental Solutions. The number of sources required will be discussed elsewhere, but it should be noted that the number of sources must reflect the degrees of freedom inherent in the boundary conditions of the problem. This determines the minimum number of sources required. Beyond that, extra sources produce more accurate results but ill-conditioning effects increase with the number of sources.

References

[1] Mathon, R. and Johnston, R. The Approximate Solutions of Elliptic Boundary Value problems by Fundamental Solutions, in SIAM J. Numerical Analysis, vol.14,#4,Sept 1977

[2] Karageorghos, A. and Fairweather, G. The method of Fundamental Solutions for the solution of Plane Potential Problems, in IMA J.of Numerical Analysis, vol.9,pp231-42,1989

[3] Cartwright, D.J. Underlying principles of the Boundary Element Method. WIT Press 2001

[4] Goldberg, M.A. and Chen, C.S. The method of Fundamental Solutions for potential, Helmholtz and diffusion problems, in Boundary Integral Methods - Numerical and Mathematical Aspects (ed. Goldberg, M.A.) pp103-76, CMP 1998

[5] Burgess, G. and Mahajerin, E. The Fundamental Collocation Method Applied to the Nonlinear Poisson Equation in two Dimensions, in Computers and Structures, vol. 17, #6, pp 763-767. 1987 Pergamon Journals
[6] Brebbia, C.A. The Boundary Element Method for Engineers. Pentech Press 1980
[7] Brebbia, C.A. and Dominguez, J. Boundary Elements: an Introductory Course. McGraw-Hill/CMP 1989