# Advancement in Node Positioning Algorithms for the Complex Variable Boundary Element Method

1LT Bryce D. Wilkins[1],
Professor Theodore V. Hromadka II[2]

[1]Carnegie Mellon University
[2]Distinguished Professor, United States Military Academy

*bdwilkins95@gmail.com*

November 20, 2020

# Section 1

## Overview of CVBEM Methodology

# The General CVBEM Approximation Function

- The CVBEM approximation function is a linear combination of complex variable functions that are analytic within a given problem domain, $\Omega$:

$$\hat{\omega}(z) = \sum_{j=1}^{n} c_j g_j(z), \quad z \in \Omega, \tag{1}$$

  where

  - $c_j = \alpha_j + i\beta_j$ are complex coefficients (note: 2 real coefficients),
  - $g_j(z)$ are analytic complex variable basis functions,
  - $n$ is the number of basis functions being used in the approximation

- Each term in the approximation function corresponds to one node and two collocation points.

# Problem Formulation

The Cauchy integral formula:

$$\omega(z) = \frac{1}{2\pi i} \oint_{\Gamma} \frac{\omega(\zeta) d\zeta}{\zeta - z}. \qquad (2)$$

Integration of (2) results in the following sum, which is known as the CVBEM approximation function:

$$\hat{\omega}(z) = \sum_{j=1}^{n} c_j (z - z_j) \ln(z - z_j). \quad (3)$$

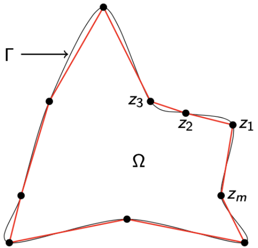



**Figure:** The boundary is discretized using a set of interpolation points. The interpolation points can be connected using straight line segments to create a polygonal representation.

# The CVBEM Modeling Procedure

- The points $z_j$ are the branch points of the logarithm (with branch cuts rotated) and are often referred to as computational nodes.
- The CVBEM can be viewed as a procedure for generating basis functions, such as in (4).
- The generated basis functions are used as inputs for the NPAs.

The CVBEM approximation function is as follows:

$$\hat{\omega}(z) = \sum_{j=1}^{n} c_j (z - z_j) \ln(z - z_j). \tag{4}$$

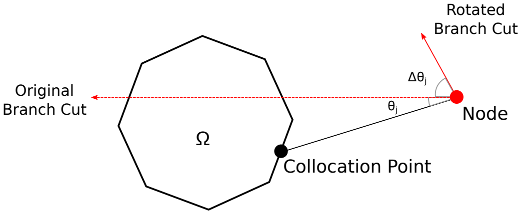



**Figure:** Rotation of a typical branch cut. The branch point of the basis function corresponds to a node for the NPA.

Section 2

# Advancements in Node Positioning Algorithms

# NPA0

Hromadka II, T.V. & Guymon, G.L., A Complex Variable Boundary Element Method: Development. *International Journal for Numerical Methods in Engineering*, **20**, pp. 25-37, 1984.

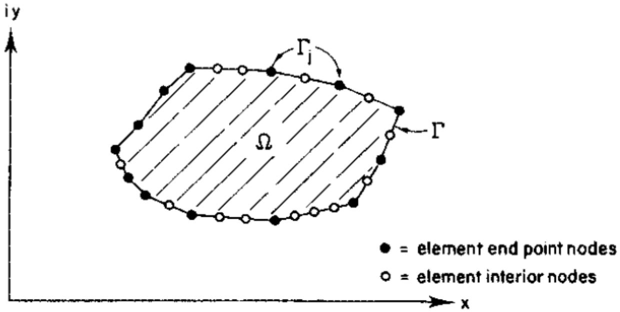



**Figure:** Originally, nodes were located on the problem boundary.

# NPA0.5

Johnson, A.N. & Hromadka II, T.V., Modeling mixed boundary conditions in a Hilbert space with the complex variable boundary element method (CVBEM). *MethodsX*, **2**, pp. 292-305, 2015.

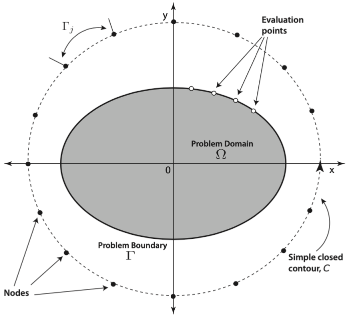



**Figure:** Next, nodes were located in a geometric pattern in the exterior of $\Omega \cup \partial\Omega$.

# NPA1

Demoes, N.J., Bann, G.T., Wilkins, B.D., Grubaugh, K.E. & Hromadka II, T.V., Optimization Algorithm for Locating Computational Nodal Points in the Method of Fundamental Solutions to Improve Computational Accuracy in Geosciences Modeling. *The Professional Geologist*, pp. 6-12, 2019.

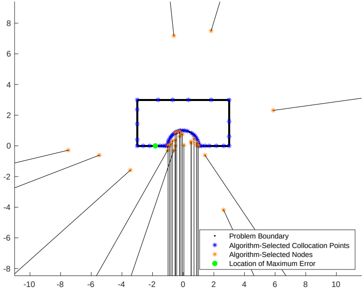



**Figure:** Nodes and collocation points are selected so as to decrease error in fitting boundary conditions.

# NPA2

Wilkins, B.D., Hromadka II, T.V. & McInvale, J., Comparison of Two Algorithms for Locating Nodes in the Complex Variable Boundary Element Method (CVBEM). *International Journal of Computational Methods and Experimental Measurements*, in press.

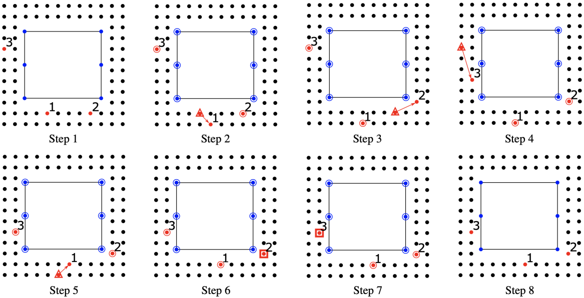



**Figure:** A refinement procedure is added, which allows for the re-location of previously located nodes.

# NPA3

Under current development...

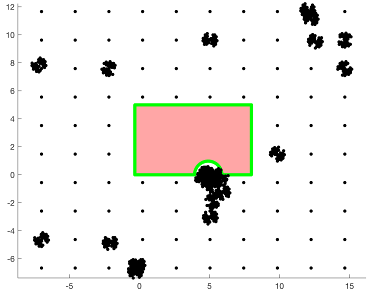



**Figure:** The latest NPA allows for variable candidate node density with increased node density in possible areas of interest.

Section 3

# Example Problem and Results

# Example Problem Details

| | |
|---|---|
| Problem Domain: | $\Omega = \left\{ (x, y) : \ -3 \leq x \leq 3, \ 0 \leq y \leq 3, \right.$ $\left. \text{and } x^2 + y^2 \geq 1 \right\}$ |
| Governing PDE: | $\nabla^2 \psi = 0$ |
| Boundary Conditions: | $\psi(x, y) = \Im[z + \frac{1}{z}], \quad (x, y) \in \partial\Omega$ |
| Number of Candidate Computational Nodes: | 1,000 |
| Number of Candidate Collocation Points: | 500 |

**Table:** Potential Flow Around a Cylindrical Obstacle - Problem Description

# Analytic Solution

- The example problem considers potential flow around a cylinder with the analytic solution given by:

$$\omega(z) = z + \frac{1}{z}$$

- The flow regime approaches potential flow in a 90-degree bend at the stagnation points.
- The stagnation points are difficult to model computationally because of the extreme curvature of the flow regime.

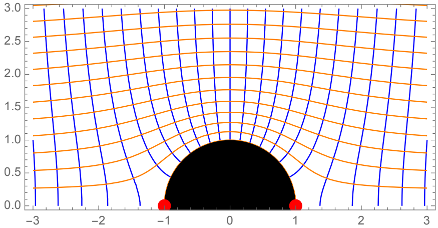



**Figure:** Analytic solution used for comparison between NPA1 and NPA2. The stagnation points are indicated by red points at $(-1, 0)$ and $(1, 0)$.

# NPA Comparisons

Computational results from the use of NPAs 1 and 2 to determine the nodes and collocation points of a CVBEM model with $n = 20$.

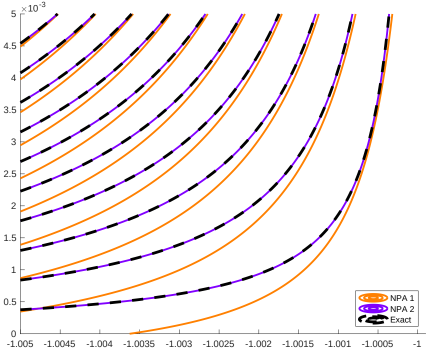



**Figure:** Streamlines.

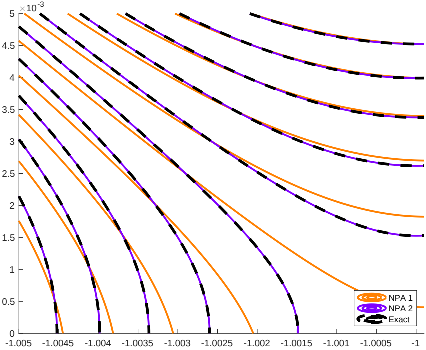
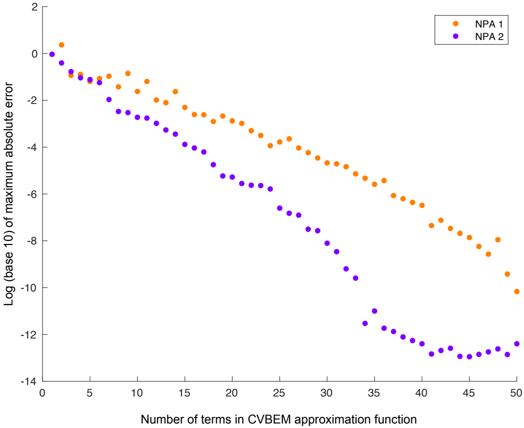



**Figure:** Potential lines.

# Error Results

**Figure:** Maximum absolute error of CVBEM models resulting from the use of NPAs 1 and 2 as each new node is added up to a total of 50 nodes. After $n = 10$, it is clear that the NPA2 approximation is several orders of magnitude more accurate than the NPA1 approximation.



# Time Results

| Number of Basis Functions | Number of d.o.f. | Unrefined Method (NPA1): Maximum Error | Time Elapsed (sec) |
|---|---|---|---|
| 10 | 20 | 2.376217e-02 | 2.600493 |
| 20 | 40 | 1.324917e-03 | 5.413931 |
| 30 | 60 | 2.123033e-05 | 10.021206 |
| 40 | 80 | 3.277547e-07 | 11.846832 |
| 50 | 100 | 6.828804e-11 | 16.865822 |

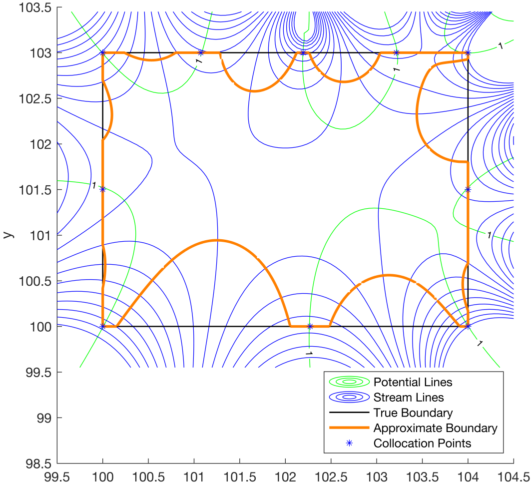
| Number of Basis Functions | Number of d.o.f. | Refined Method (NPA2): Maximum Error | Time Elapsed (sec) |
|---|---|---|---|
| 10 | 20 | 6.731285e-03 | 26.847856 |
| 20 | 40 | 1.639780e-05 | 101.625993 |
| 30 | 60 | 3.783824e-09 | 199.087752 |
| 40 | 80 | 1.816325e-13 | 408.392388 |
| 50 | 100 | 1.163514e-13 | 672.789040 |

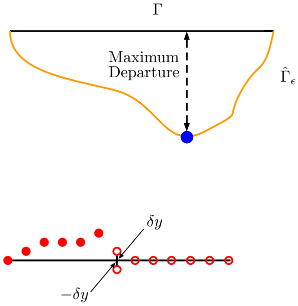
**Table:** Maximum error and time elapsed for various CVBEM models of a Dirichlet boundary value problem.

Section 4

# Final Thoughts - The Approximate Boundary Method

# The Approximate Boundary Method



# Questions