The Journal of Open Source Software

NBI: A library for Nystrom Boundary Integral calculations

Michael J. Carley [1]

1 Department of Mechanical Engineering, University of Bath, UK

DOI: 10.21105/joss.05972

Software

- Review
- Repository
- Archive

Editor: Vissarion Fisikopoulos

Reviewers:

- @mscroggs
- @bonh

Submitted: 12 September 2023
Published: 29 July 2024

Summary

Boundary integral methods are a popular approach to solving potential problems, in particular for the Laplace and Helmholtz problems in such areas as fluid dynamics, acoustics, and electromagnetism. They are a natural choice for the solution of problems in unbounded domains, such as wave scattering, where the radiation boundary condition is automatically satisfied by the nature of the formulation. The Nystrom method is one approach to the solution of boundary integral problems, which lends itself to acceleration using the Fast Multipole Method (FMM). NBI is a library for the solution of boundary integral problems based on the approach of Greengard et al. (2021) and the FMM methods of Gumerov & Duraiswami (Gumerov & Duraiswami, 2003, 2004, 2005, 2009). The code includes a number of executables which can be used to set up and solve problems on realistic geometries, with a number of examples provided for testing of the solver. Results can be visualized using GMSH (Geuzaine & Remacle, 2009), a standard free meshing program.

Statement of need

A number of free boundary integral codes exist that can be used for problems of the type handled by NBI (Betcke & Scroggs, 2021, for example; Kirkup, 2007), with different approaches to discretization and solution, ranging from collocation methods with direct solvers, suitable for relatively small problems, to FMM-accelerated Galerkin techniques with iterative solvers, which can be used on large complex geometries. NBI implements recent work on high-order solution of boundary integral problems (Greengard et al., 2021) in a form that can be used for a range of engineering applications. The principal motivation is the solution of potential problems in acoustics and aerodynamics. As well as the library proper, there are codes for discretization of general geometries supplied in a number of formats; matrix assembly and problem solution; field evaluation; and post-processing and visualization of results. A built-in parser gives a flexible and intuitive means of evaluating boundary conditions, which are supplied as analytical expressions parsed and evaluated on the surface. This allows boundary conditions to be specified in a form that makes it easier to perform parametric studies by systematically modifying internal variables.

Simple geometries, such as spheres and ellipsoids, are implemented as internal functions. More general shapes can be produced using standard GMSH inputs (Geuzaine & Remacle, 2009) or the CST method of Kulfan (2010), which is implemented in an extended and more general form in the AGG library that is included with the NBI distribution. This allows the parametric specification of forms that arise in aerodynamic applications, in a manner that is convenient for optimisation and parametric studies.

Mathematics

The Laplace and Helmholtz potentials generated by a surface source distribution are given by a boundary integral

$$\phi(\mathbf{x}) = \int_S \phi(\mathbf{y}) \frac{\partial}{\partial n} G(\mathbf{x},\mathbf{y}) - \frac{\partial \phi(\mathbf{y})}{\partial n} G(\mathbf{x},\mathbf{y}) \,\mathrm{d}S(\mathbf{y}), \qquad (1)$$

where $G(\mathbf{x},\mathbf{y})$ is the Green's function for the problem. For points lying on the surface, the left hand side is replaced by $\phi/2$ and the equation is interpreted as a boundary integral equation to be solved subject to an appropriate boundary condition. By default, NBI implements a Neumann (surface normal derivative) boundary condition as that most appropriate to acoustic scattering and to potential problems in fluid dynamics.

Features

Details of the problem setup and solution procedure are given in the code documentation, but the basic steps are:

- generation of a surface discretization;
- assembly of the system matrices;
- solution of the problem subject to a specified boundary condition; and
- postprocessing, including evaluation of the potential field, and visualization.

Each of these steps is carried out using an appropriate executable, and a full problem can be solved using a Unix or Linux shell script that runs each step in turn.

The intention has been to develop an implementation of the main elements of Greengard et al. (2021) in a code that is relatively easy to use in applications. The principal differences that a user will note between NBI and the reference implementation of the method of Greengard et al. (2021) are: the range of geometry formats supported, including the widely used GMSH standard, and an extension of Kulfan's CST method for "aerodynamic'' shapes (Kulfan, 2010); the use of the standard PETSc iterative solvers, with the option to select a solver at runtime; and the ability to define boundary conditions using a built-in parser for mathematical expressions, or pre-defined functions for the sources that arise in realistic problems, such as scattering of rotor noise.

Surface generation and representation

Surfaces in NBI are represented as high-order triangular patches, interpolated using Koornwindwer orthogonal polynomials (Greengard et al., 2021). A number of surface generation methods are provided (built-in basic geometries, parametric surfaces, GMSH geo files) allowing considerable flexibility in the workflow for real problems.

Quadrature rules

Two main types of quadrature rule are used in NBI. Surface points are generated on each patch at the nodes of high order rules for triangles (Wandzura & Xiao, 2003; Xiao & Gimbutas, 2010). These nodes are used, via the FMM, to evaluate the far-field interactions in the boundary integrals. Near-field interactions are computed by subtracting the terms generated by the surface sources and adding terms evaluated using adaptive quadrature and specialized rules for quadrature on triangles, pre-computed using methods for generalized Gaussian rules (Bremer & Gimbutas, 2013). These are hard-coded into the library and selected automatically during matrix assembly.

Solving problems

Problems in NBI are solved using iterative solvers with matrix-free evaluation of integrals. The evaluation uses a Fast Multipole Method (Gumerov & Duraiswami, 2003, 2004, 2005, 2009). A GMRES solver is built in and there is an optional interface to the PETSc library and its solvers (Balay et al., 2023). Boundary conditions in NBI are specified using analytical formulae that are parsed and evaluated at each surface node. This allows an intuitive definition of boundary conditions, where user-defined variables simplify parametric variation, and a range of built-in functions for the most common boundary conditions are also available. These include point sources for the Laplace and Helmholtz problems, and ring sources for the simulation of rotor acoustics in aeronautics.

Postprocessing and visualization

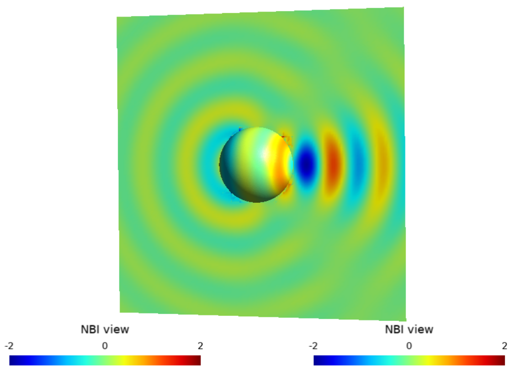

Figure 1: GMSH visualisation of output from sphere scattering example.

Figure 1 shows the GMSH visualisation of the scattered field from a sphere subject to plane wave excitation, one of the test cases included in the package.

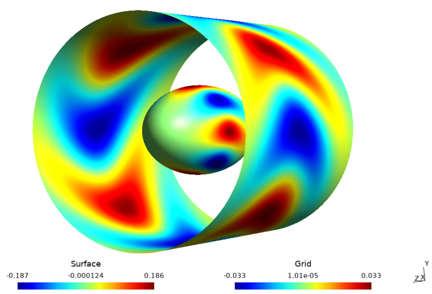

Figure 2: GMSH visualisation of output from rotor noise example.

Figure 2 shows the GMSH visualisation of the field scattered from an ellipsoid, subject to an incident field from a ring source, a simple model for scattering of rotor noise by a nacelle. In this case, the boundary condition is defined using a built-in function in NBI, demonstrating its application to realistic problems with complex boundary conditions.

References

Balay, S., Abhyankar, S., Adams, M. F., Benson, S., Brown, J., Brune, P., Buschelman, K., Constantinescu, E. M., Dalcin, L., Dener, A., Eijkhout, V., Faibussowitsch, J., Gropp, W. D., Hapla, V., Isaac, T., Jolivet, P., Karpeev, D., Kaushik, D., Knepley, M. G., … Zhang, J. (2023). *PETSc Web page*. https://petsc.org/

Betcke, T., & Scroggs, M. W. (2021). Bempp-cl: A fast Python based just-in-time compiling boundary element library. *Journal of Open Source Software*, *6*(59). https://doi.org/10.21105/joss.02879

Bremer, J., & Gimbutas, Z. (2013). On the numerical evaluation of the singular integrals of scattering theory. *Journal of Computational Physics*, *251*, 327–343. https://doi.org/10.1016/j.jcp.2013.05.048

Geuzaine, C., & Remacle, J.-F. (2009). Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, *79*(11), 1309–1331. https://doi.org/10.1002/nme.2579

Greengard, L., O'Neil, M., Rachh, M., & Vico, F. (2021). Fast multipole methods for the evaluation of layer potentials with locally-corrected quadratures. *Journal of Computational Physics: X*, *10*, 100092. https://doi.org/10.1016/j.jcpx.2021.100092

Gumerov, N. A., & Duraiswami, R. (2003). Recursions for the computation of multipole translation and rotation coefficients for the 3-D Helmholtz equation. *SIAM Journal on Scientific Computing*, *25*(4), 1344–1381. https://doi.org/10.1137/S1064827501399705

Gumerov, N. A., & Duraiswami, R. (2004). *Fast multipole methods for the Helmholtz equation in three dimensions*. Elsevier. https://doi.org/10.1016/b978-0-08-044371-3.x5000-5

Gumerov, N. A., & Duraiswami, R. (2005). *Comparison of the efficiency of translation operators used in the fast multipole method for the 3D Laplace equation* (CSTR 4701, UMIACS TR-2005-09). University of Maryland.

Gumerov, N. A., & Duraiswami, R. (2009). A broadband fast multipole accelerated boundary element method for the three dimensional Helmholtz equation. *Journal of the Acoustical Society of America*, *125*(1), 191–205. https://doi.org/10.1121/1.3021297

Kirkup, S. M. (2007). *The boundary element method in acoustics*. Integrated Sound Software. http://www.boundary-element-method.com

Kulfan, B. M. (2010). Recent extensions and applications of the "CST" universal parametric geometry representation method. *Aero*, *114*(1153), 157–176. https://doi.org/10.1017/S0001924000003614

Wandzura, S., & Xiao, H. (2003). Symmetric quadrature rules on a triangle. *Computers and Mathematics with Applications*, *45*, 1829–1840. https://doi.org/10.1016/S0898-1221(03)90004-6

Xiao, H., & Gimbutas, Z. (2010). A numerical algorithm for the construction of efficient quadrature rules in two and higher dimensions. *Computers and Mathematics with Applications*, *59*, 663–676. https://doi.org/10.1016/j.camwa.2009.10.027