A Sustainable Software Architecture for Scalable Nonlinear Boundary Element Method Simulations

Jaydeep P. Bardhan
Mechanical & Industrial Engineering, Northeastern University
jbardhan@alum.mit.edu

Matthew G. Knepley
Computational and Applied Mathematics, Rice University
knepley@gmail.com

We acknowledge support from the Department of Energy Office of Advanced Scientific Computing, the NIH R21 Program, and the Rice Intel Parallel Computing Center.

Algorithmic Breakdown

Design goals:

- Ability to interface with parallel N-body algorithms
- Flexbility in discretization (Galerkin, collocation, higher-order methods)
- Ability to couple to other models, including FEM, time-stepping, nonlinear solvers

BIE Architecture:

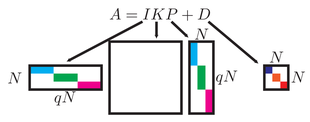
The prototypical linear BEM problem $Ax = b$ can be decomposed into

$$A_{ij} = \int_{\text{panel } i} \beta_i(\vec{r}) \left[\int_{\text{panel } j} G(\vec{r}, \vec{r}')\chi_j(\vec{r}')d\vec{r}' \right] d\vec{r} \tag{1}$$

where

G Green's Function or derivative
χ basis function (usually piecewise constant or linear)
β test function (χ for Galerkin, $\delta(\vec{r} - \vec{r}_i)$ collocation)

In order to compute these parts, we propose a divison of the software into four pieces:



P Sparse, matrix-free **projection** maps basis function weights to source densities at quadrature points

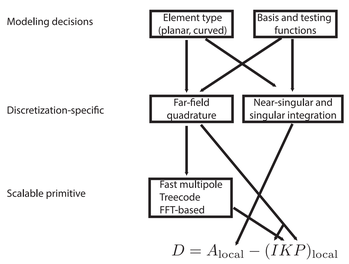
I Sparse, matrix-free **interpolation** evaluates integrals over test functions

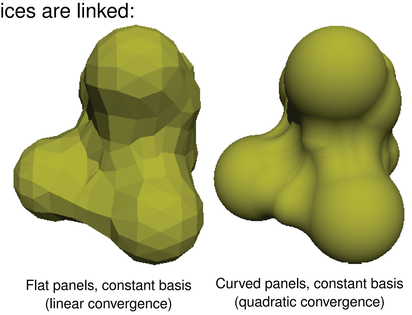
K Scalable N**-body** algorithm (point-to-point kernel)

D Sparse **pre-correction** can be applied matrix-free

Challenge

Software and modeling components in boundary-integral methods are usually tightly coupled, so that the computational choices are linked:





Flat panels, constant basis (linear convergence)

Curved panels, constant basis (quadratic convergence)

Nonlinear BIE

All BEM packages today are based upon linear integral operators. However, nonlinearity can be crucial for model fidelity. In molecular electrostatics, we have shown that replacing the standard Maxwell boundary condition at the molecular surface with

$$(\epsilon_{in} - \Delta\epsilon\, h\left(E_n(\boldsymbol{r}_{S-})\right)) \frac{\partial \phi_{in}}{\partial n}(\boldsymbol{r}_{S-}) = (\epsilon_{out} - \Delta\epsilon\, h\left(E_n(\boldsymbol{r}_{S-})\right)) \frac{\partial \phi_{out}}{\partial n}(\boldsymbol{r}_{S+}) \tag{2}$$

can calculate very accurate energies (matching Generalized Born models wih hundreds of parameters), and also deliver accurate entropy in solution. Here $\Delta\epsilon = \epsilon_{out} - \epsilon_{in}$ and $E_n(\vec{r}_{S-})$ is the normal electric field at $\vec{r}_{S-}$. Note that the electric field just outside the surface does not explicitly enter into the interface condition. The equivalent BIE is then given by

$$\left(\frac{\hat{\epsilon}}{2} I + K' + h\left(E_n\right)\right) \frac{\sigma}{\epsilon_{out}} = \frac{1}{\epsilon_{out}} \sum_i \left(-\frac{\partial G}{\partial n}\right) q_i, \tag{3}$$

in which the nonlinear operator h represents a pointwise multiplication, and is given by

$$h\left(E_n\right) = \alpha \tanh\left(\beta E_n - \gamma\right) + \mu. \tag{4}$$

which we term the Solvation Layer Interface Condition (**SLIC**).

Solver Integration

Our BIE framework now plugs into the PETSc SNES interface for parallel nonlinear algebraic equations. We provide

- the action of the integral operator, perhaps using a fast method
- (Optional) the action of a simplified linearization, perhaps using a cutoff

This affords a range of nonlinear solvers, even using only the operator action,

- Nonlinear Richardson
- Nonlinear Conjugate Gradients
- Nonlinear Krylov and Anderson mixing
- Quasi-Newton methods like BFGS
- Nonlinear Multigrid

Model Coupling

In addition, we can integrate with the PETSc DM framework for specifying data layout, function spaces, and equations. We can use a DMPlex object to specify our boundary mesh and data layout, even for higher order discretizations. This can be combined with other discretized fields which have support on the boundary mesh itself or a volumetric mesh which intersects the boundary. We can use

- finite element fields, using PetscFE
- finite volume fields, using PetscFV
- finite difference fields, using DMDA

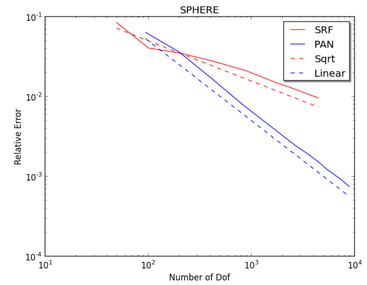
and then the full residual for the combined system can be formed as fed to a PETSc TS or SNES object which solves the full time-dependent system.

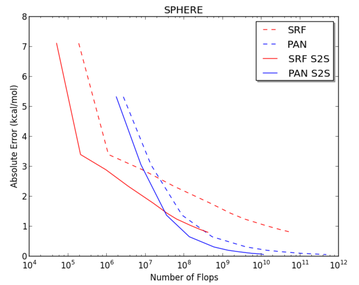
Architecture Capabilities:

- Explicit, specialized panel integrals as needed for pre-correction
- Fast summation algorithm is completely orthogonal to discretization
- Discretization choices (element types, basis and test functions) local to one layer
- Handle Galerkin, collocation, other approaches readily via the $IKPD$ decomposition
- Panel quadrature order can be adjusted as needed
- Multiple panel types and basis/test functions can be used in a single BEM simulation

Panel vs. Point Discretization

Our extensible framework allowed us to investigate the tradeoff between work and accuracy for representative variants of the BEM discretization, and quantify the dependence of this tradeoff on mesh resolution and the geometric complexity of the molecular boundary (https://arxiv.org/abs/1512.08406). Point discretizations are generally assumed to lack the accuracy for practical prediction. This intuition is borne out by simple experiments on a sphere of radius 6 Å and $\epsilon = 4$, immersed in water ($\epsilon = 80$), with 10 random charges inside it. The panel method is superior below 3 Kcal/mol accuracy, but considering only the surface-to-surface operator (S2S) the point method is superior down to 1.5 Kcal/mol.





The convergence for an arginine molecule (ARG) is much noisier due to the geometric complexity, and the panel method does not outperform the point method until 0.75 Kcal/mol (0.25 Kcal/mol for S2S), enough accuracy for many molecular conformational searches.