Stochastic Computational Electrostatics and Applications: Computing Capacitance

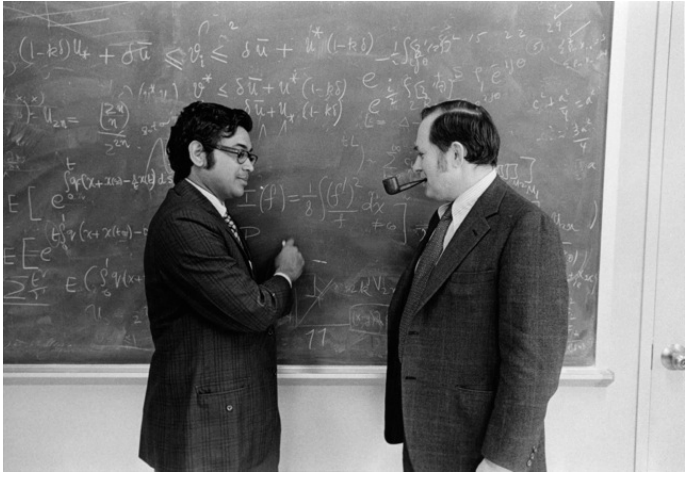
Prof. Michael Mascagni

Department of Computer Science
Department of Mathematics
Department of Scientific Computing
Graduate Program in Molecular Biophysics
Florida State University, Tallahassee, FL 32306 **USA**
AND
Applied and Computational Mathematics Division, Information Technology Laboratory
National Institute of Standards and Technology (NIST), Gaithersburg, MD 20899-8910 **USA**
E-mail: mascagni@fsu.edu or mascagni@nist.gov
URL: http://www.cs.fsu.edu/∼mascagni

In collaboration with Walid Keyrouz and Derek Juba of NIST and John Thrasher of USCB

Outline of the Talk

Thanks to My Probability Professors at Courant: Raghu Varadhan and Monroe Donsker



In Memoriam: Nikolai Simonov, 1956–2019

Early History of MCMs for PDEs

1. Courant, Friedrichs, and Lewy: Their pivotal 1928 paper has probabilistic interpretations and MC algorithms for linear elliptic and parabolic problems
2. Fermi/Ulam/von Neumann: Atomic bomb calculations were done using Monte Carlo methods for neutron transport, their success inspired much post-War work especially in nuclear reactor design
3. Kac and Donsker: Used large deviation calculations to estimate eigenvalues of a linear Schrödinger equation
4. Forsythe and Leibler: Derived a MCM for solving special linear systems related to discrete elliptic PDE problems

The First Passage (FP) Probability is the Green's Function

The canonical elliptic boundary value problem for electrostatics (one of Maxwell's equations):

$$\frac{1}{2}\Delta u(x) = 0, \quad x \in \Omega$$
$$u(x) = f(x), \quad x \in \partial\Omega$$

- Distribution of z is uniform on the sphere
- $u(x)$ has mean-value property and harmonic
- Also, $u(x)$ satisfies the boundary condition

$$u(x) = \mathbb{E}_x[f(X^{\mathbf{x}}(t_{\partial\Omega}))] \tag{1}$$

The First Passage (FP) Probability is the Green's Function

Reinterpreting as an average of the boundary values

$$u(x) = \int_{\partial\Omega} p(x,y)\, f(y)\, dy \tag{2}$$

Another representation in terms of an integral over the boundary

$$u(x) = \int_{\partial\Omega} \frac{\partial g(x,y)}{\partial \mathbf{n}}\, f(y)\, dy \tag{3}$$

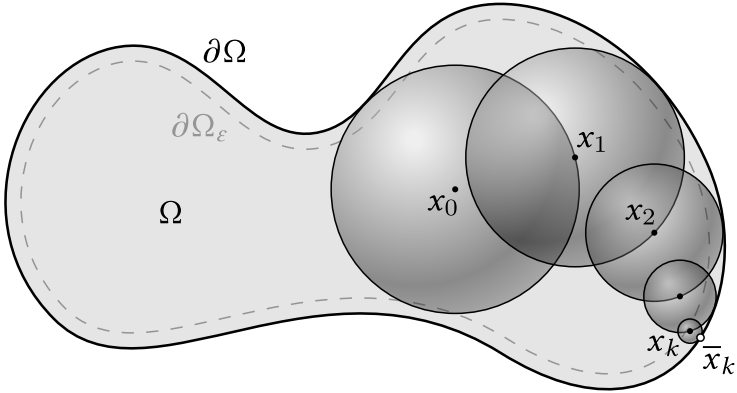
$g(x,y)$ – Green's function of the Dirichlet problem in Ω

$$\Longrightarrow p(x,y) = \frac{\partial g(x,y)}{\partial \mathbf{n}} \tag{4}$$

'Walk on Spheres' (WOS) and 'Green's Function First Passage' (GFFP) Algorithms

- Green's function is known
  $\Longrightarrow$ direct simulation of exit points and computation of the solution through averaging boundary values
- Green's function is unknown
  $\Longrightarrow$ simulation of exit points from standard subdomains of Ω, e.g. spheres
  $\Longrightarrow$ Markov chain of 'Walk on Spheres' (or GFFP algorithm) $x_0 = x, x_1, \ldots, x_N$
  $x_i \to \partial\Omega$ and hits ε-shell is $N = O(|\ln(\varepsilon)|)$ steps
  x_N simulates exit point from Ω with $O(\varepsilon)$ accuracy

The First Passage (FP) Probability is the Green's Function



Computing Permeability via Capacitance

- Hubbard and Douglas from NIST showed that the permeability of a porous medium is functionally related to the capacitance of a nonskew conductor
- Capacitance and permeability are proportionality constants
  1. Capacitance is the ratio of charge to voltage
  2. Permeability is the ratio of pressure gradient to flux
- Capacitance and permeability are completely determined by geometry
- The mathematical problems surrounding both are elliptic
  1. For capacitance it is the Laplace (a Maxwell) equation
  2. For permeability it is the Stokes equation for creeping flow

Computing Capacitance

- Recall that $C = \frac{Q}{u}$, if we hold the conductor, Ω, at unit potential $u = 1$, then $C =$ total charge the conductor's surface, $\partial\Omega$
- The PDE system for the potential is

$$\Delta u = 0, \quad x \notin \Omega; \quad u = 1, \quad x \in \partial\Omega; \quad u \to 0 \ \text{ as } \ |x| \ \to \infty \tag{5}$$

- If we know the charge density, q on the conductor, Ω, then the total charge is just

$$\int_{\partial\Omega} q(s)\, ds$$

- Knowing $u(x)$ gives us $q(x) = -\frac{1}{4\pi}\frac{\partial u}{\partial n}$
- Solving for capacitance requires computing the following for the solution to the above Laplace equation

$$Q = -\frac{1}{4\pi}\int_{\partial\Omega} \frac{\partial u(s)}{\partial n(s)}\, ds \tag{6}$$

Computing Capacitance Deterministically

- The problem we are solving for capacitance the exterior problem, and so one must discretize based on the method of choice
  1. Boundary Element Method (BEM): Must discretize $\partial\Omega$
  2. Finite Element Method (FEM) or Finite Difference Method (FDM): Must discretize $\mathbb{R}^3 \setminus \Omega$
- Solve the exterior Laplace equation (5) for u
- Differentiate u on $\partial\Omega$
- Integrate to compute the total charge using equation (6)

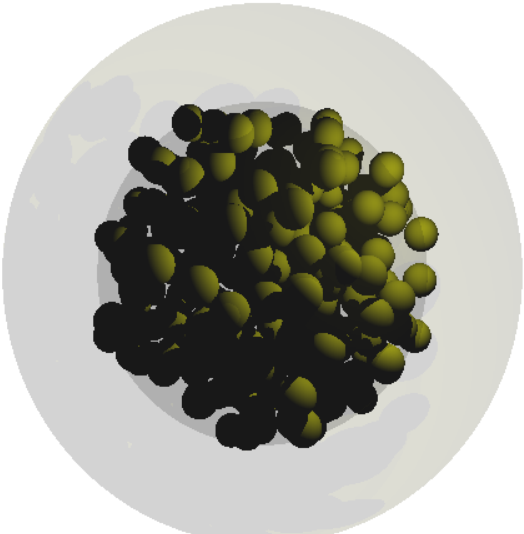
Computing Capacitance Probabilistically

- The PDE system for the potential is

$$\Delta u = 0, \quad x \notin \Omega; \quad u = 1, \quad x \in \partial\Omega; \quad u \to 0 \text{ as } |x| \to \infty$$

- Recall $u(\mathbf{x}) = \mathbb{E}_x[f(X^{\mathbf{x}}(t_{\partial\Omega}))] = 1$, if the walker hits $\partial\Omega$, or 0, if the walker $\to \infty =$ probability of walker starting at $\mathbf{x}$ hitting Ω before escaping to infinity
- Charge density is first passage measure on the exterior of Ω
- Construct a sphere, $S(R)$, such that $\Omega \subset S(R)$
- Capacitance is $C = R \times \mathbb{P}(t_{\partial\Omega} < \infty) = R \times \int_{\partial S(R)} u(\mathbf{x}) ds$, where the starting point of the walk is chosen uniformly on the $\partial S(R)$
- Note, this definition is independent of R

Porous Media: Complicated Interfaces



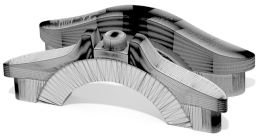
Capacitance as a Computational Paradigm

- We have found that the running time of these codes hinged on the efficiency of geometrical computations
- The probabilistic capacitance computation is key in
  1. Permeability computation via Hubbard-Douglas
  2. Many other electrostatics computations in areas of chemistry, computer science, physics, and materials science, including our own PBE computations
  3. In the ZENO code written for materials property computations at NIST
- We took the NIST ZENO code and analyzed it to determine areas for improvement

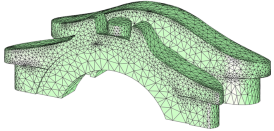
What is the Computational Geometric Problem?

- ZENO is named for the paradox, and getting close the boundary for producing approximate first-passage locations is done with the ZENO algorithm, which is WOS by another name
- The usual geometry in these problems is additive
- As mentioned above the primitive computation geometrically is given a query point, find the closest point to the boundary, and that is used as the radius to construct the WOS radius
- The usual ZENO problem is the so-called exterior problem, which is what is done in permeability: computing the probability of first-passage from ∞ using relative capacitance of a bounding sphere

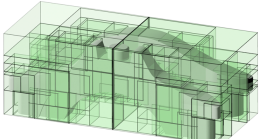
Gridding vs. Hierarchical Decomposition



input
(Thingi10k #996816)



mesh w/ FastTetWild
1 hour 25 minutes



build BVH for WoS
< 1 second

Figure: Gridding for FEM vs. Bounded Volume Hierarchy (BVH) for WOS Geometric Query

Mathematical Model: Molecular Geometry

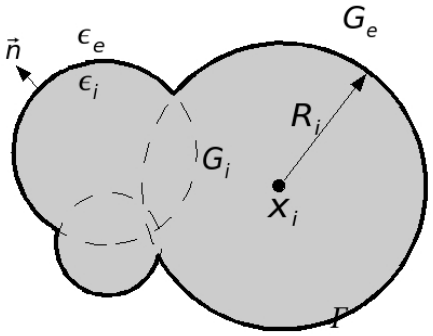


Figure: Biomolecule with dielectric ϵ_i and region region G_i is in solution with dielectric ϵ_e and region G_e. On the boundary of the biomolecule, electrostatic potential and normal component of dielectric displacement continue

Mathematical Model: Partial Differential Equations

- Poisson equation for the electrostatic potential, Φ_i, and point charges, Q_m, inside a molecule (in CGS units):

$$\epsilon_i \Delta \Phi_i(x) + 4\pi \sum_{m=1}^{M} Q_m \delta(x - x^{(m)}) = 0 \ , \ x \in G_i$$

- For 1-1 salt (such as *NaCl*) Poisson-Boltzmann equation (PBE):

$$\Delta \Phi_e(x) - \kappa^2 \sinh(\Phi_e(x)) = 0 \ , \ x \in G_e \ ,$$

but we only consider the linearized PBE:

$$\Delta \Phi_e(x) - \kappa^2 \Phi_e(x) = 0 \ , \ x \in G_e$$

- For one-surface model: continuity condition on the dielectric boundary

$$\Phi_i = \Phi_e \ , \ \epsilon_i \frac{\partial \Phi_i}{\partial n(y)} = \epsilon_e \frac{\partial \Phi_e}{\partial n(y)} \ , \ y \in \Gamma$$

Mathematical Model: Debye-Hückle Parameter

Dependence on salt in the Debye-Hückle parameter (units as per Kirkwood):

$$\kappa^2 = \frac{8\pi N_A e^2 C_s}{\epsilon_e 1000 k_B T}, \text{ where}$$

- C_s – concentration of ions (in moles)
- N_A – Avogadro's number
- e – elementary protonic charge
- k_B – Boltzmann's constant
- ϵ_e – dielectric permittivity outside the molecule

Electrostatic Potential and Energy

- Point values of the potential: $\Phi(x) = \Phi_{rf}(x) + \Phi^c(x)$
  Here, singular part of Φ:
$$\Phi^c(x) = \sum_{m=1}^{M} \frac{Q_m}{|x - x^{(m)}|}$$
- Reaction field electrostatic free energy of a molecule is linear combination of point values of the regular part of the electrostatic potential:
$$W_{rf} = \frac{1}{2} \sum_{m=1}^{M} \Phi_{rf}(x^{(m)}) Q_m \ ,$$
- Electrostatic solvation free energy = difference between the energy for a molecule in solvent with a given salt concentration and the energy for the same molecule in vacuum:
$$\Delta G^{elec}_{solv} = W_{rf}(\epsilon_i, \epsilon_e, \kappa) - W_{rf}(\epsilon_i, 1, 0)$$

'Walk-on-Spheres' Algorithm

- Walk-on-spheres (WOS) algorithm for general domains with a regular boundary
- Define a Markov chain $\{x_i,\ i = 1, 2, \ldots\}$
- Set $x_0 = x^{(m)}$ for some m, $x_i = x_{i-1} + d_i \omega_i\ ,\quad i = 1, 2, \ldots$, where
  1. $d_i = d(x_{i-1})$ is distance from x_{i-1} to Γ
  2. $\{\omega_i\}$ is sequence of independent unit isotropic vectors
  3. x_i is the exit point from the ball, $B(x_{i-1}, d(x_{i-1}))$, for a Brownian motion starting at x_{i-1}
- Outside the molecule, on every step, walk-on-spheres terminates with probability $1 - q(\kappa, d_i)$, where $q(\kappa, d_i) = \dfrac{\kappa d_i}{\sinh(\kappa d_i)}$ to deal with LPBE

'Walk-on-Spheres' and 'Walk-in-Subdomains'

- For general domains, an efficient way to simulate exit points is a combination of
  1. Inside the molecule: 'walk-in-subdomains'
  2. Outside the molecule 'walk-on-spheres'
- The whole domain, G_i, is represented as a union of intersecting subdomains:

$$G_i = \bigcup_{m=1}^{M} G^m$$

- 'Walk-in-Subdomains': Simulate exit point separately in every G^m
  1. $x_0 = x, x_1, \ldots, x_N$ – Markov chain, every x_{i+1} is an exit point from the corresponding subdomain for Brownian motion starting at x_i
  2. For spherical subdomains, $B(x_i^m, R_i^m)$, exit points are distributed in accordance with the Poisson kernel:

$$\frac{1}{4\pi R_i^m} \frac{|x_i - x_i^m|^2 - (R_i^m)^2}{|x_i - x_{i+1}|^3}$$

‘Walk-on-Spheres’ and ‘Walk-in-Subdomains’

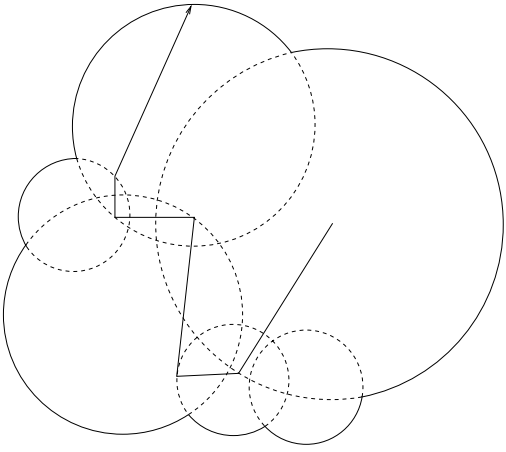


Figure: Walk in subdomains example

Monte Carlo Estimates

- The estimate for the reaction-field potential point value:

$$\begin{aligned}\xi[\Phi_{rf}](x^{(m)}) &= -\Phi^c(x_1^*) \\ &+ \sum_{j=2}^{N_{ins}} F_j(\kappa)\,(\Phi^c(x_j^{ins}) - \Phi^c(x_{j,ins}^*))\end{aligned} \tag{7}$$

- Here $\{x_{j,ins}^*\}$ is a sequence of boundary points, after which the random walker moves inside the domain, G_i, to x_j^{ins}
- The estimate for the reaction-field energy:

$$\xi[W_{rf}] = \frac{1}{2}\sum_{m=1}^{M} Q_m\,\xi[\Phi_{rf}](x^{(m)}) \tag{8}$$

Computational Results: 3 Truncated Arginine-Rich Peptides

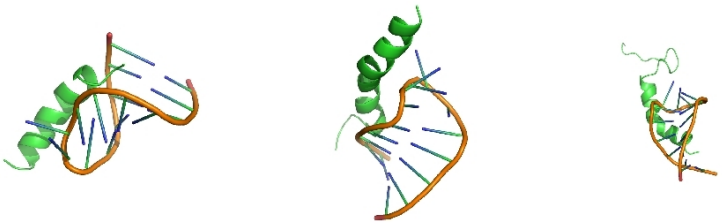


Figure: PDB IDs: 1a4t, 1hji, 1qfq

Accuracy: Monte Carlo vs. Deterministic

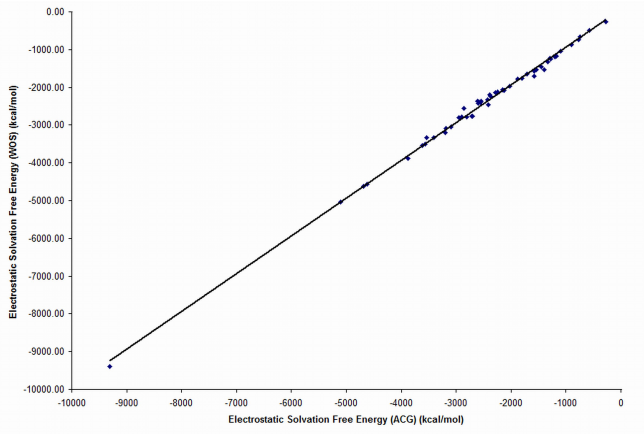


Figure: PDB IDs: 1a4t, 1hji, 1qfq

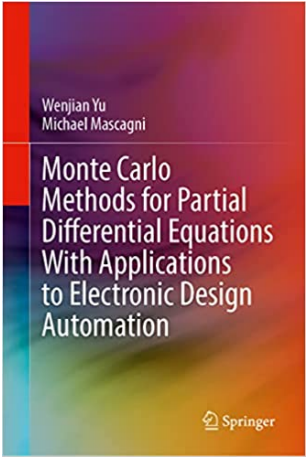
Conclusions

- Monte Carlo can be an efficient method for the numerical solution of PDEs as evidenced by
  1. Financial computing
  2. Numerous problems in electrostatics: mutual capacitance in EDA, biochemical electrostatics
  3. Numerous problems in computational materials: permeability, ZENO
- The computation of capacitance is a good model problem for computational investigation, capacitance is an important functional of geometry
  1. Computational geometry is the bottleneck that can be overcome with proper choice of data-structure/algorithm
  2. Allows for almost perfect parallelization, across cores and multicore nodes
- C++ version of ZENO uses `SPRNG`

Future Work

- Continue work on biochemical electrostatics
- We expect to be able to improve all the codes we have developed
  1. Current C++ version of the WOS-PBE code:
     `https://zeno.nist.gov`
- We want to have a WOS-based implementation of the capacitance code using CUDA on NVidia GPUs
  1. WOS is being used by computer graphics folks
  2. The geometric query in WOS is a generalization of the GPU-based geometry query in ray tracing
- Compare WOS capacitance extraction to:
  1. Deterministic methods (FEM, FDM, BEM)
  2. Machine learning methods

A Book Of Interest?



Bibliography

- C.-O. Hwang, J. A. Given and M. Mascagni (2000)] On the Rapid Calculation of Permeability for Porous Media Using Brownian Motion Paths *Physics of Fluids*, **12**: 1699–1709.
- R. Sawhney and K. Crane (2020)] Monte Carlo Geometry Processing: A Grid-Free Approach to PDE-Based Methods on Volumetric Domains *ACM Trans. Graphics*, **39(4)**: 18 pages. https://doi.org/10.1145/3386569.3392374
- D. Juba, D. J. Audus, M. Mascagni, J. Douglas, W. Keyrouz (2016)] ZENO: Software for calculating hydrodynamic, electrical, and shape properties of polymer and particle suspensions *Journal of Research of the National Institute of Standards and Technology*, **122**. https://doi.org/10.6028/jres.122.020
- P. Hamlin, W. J. Thrasher, W. Keyrouz and M. Mascagni (2019)] Geometry Entrapment in Walk-on-Subdomains *Monte Carlo Methods and Applications*, **24(4)**: 178–193.

Copyright Notice