Monte Carlo Method for Calculating the Electrostatic Energy of a Molecule

Michael Mascagni[1],[2] and Nikolai A. Simonov[2],[*]

[1] Department of Computer Science,
Florida State University, Tallahassee, FL, 32306, USA,
`mascagni@cs.fsu.edu`
[2] CSIT, Florida State University, Tallahassee, FL, 32306, USA,
`simonov@csit.fsu.edu`

Abstract. The problem of computing the electrostatic energy of a large molecule is considered. It is reduced to solving the Poisson equation inside and the linear Poisson-Boltzmann equation in the exterior, coupled by boundary conditions. A Monte Carlo estimate for the potential point values, their derivatives, and the energy is constructed. The estimate is based on the walk on spheres and Green's function first passage algorithms; the walk in subdomains technique; and finite-difference approximations of the boundary condition. Results of some illustrative calculations are presented.

1 Introduction

Electrostatics plays a fundamental role in intermolecular interactions and to a large extent determines molecular properties [2, 13]. There are different approaches to the description of the electrostatic field on the molecular level. One of the possible and widely used models is a continuum model. For a given charge distribution $\rho(x)$, the electrostatic potential is determined as a solution to Poisson's equation

$$-\nabla\epsilon\nabla u(x) = \rho(x) \ , \ x \in R^3 \ , \tag{1}$$

where ϵ is a position-dependent permittivity. In bio-molecular applications, the geometry of a problem is taken into account by thinking of a molecule as a cavity with point charges and constant ϵ inside. The exterior is modelled as a dielectric medium with different permittivity and some charge distribution. Clearly, certain boundary conditions on the surface of a molecule have to be added.

The common approach to solving these kinds of problems is the finite-difference technique; boundary element and finite element methods are successfully used as well.

---

[*] *Permanent address:* ICM&MG, Lavrentjeva 6, Novosibirsk 630090, Russia, `nas@osmf.sscc.ru`

Another possible way of treating computationally these problems comes from the probabilistic representation of solutions to elliptic and parabolic partial differential equations as functionals of diffusion process trajectories [10]. Direct computational simulation of physical diffusion in this case coincides with the approximation of a Brownian motion as the solution to a stochastic differential equation via a first-order Euler scheme (see e.g. [11]). This approach was applied [4] to simulation studies of diffusion-limited reactions. Though computationally far from being optimal, it allows one to include different physical phenomena (hydrodynamic, electrostatic, etc.) into the computational scheme, and, what is essential, it is efficient enough to be competitive with deterministic methods.

It is worth noting that probabilistic representations strongly depend on boundary conditions. In particular, taking into account reflective surfaces incorporates the local time spent by the diffusion process on the boundary [6]. Up to now, there has been no effective way to construct a computational algorithm based on this representation. So, the only conceivable method to incorporate the reflective conditions into the computational algorithm is to simulate them directly.

There are a large number of mathematical articles and books describing Monte Carlo methods for solving boundary value problems for the heat, Laplace and other diffusion equations [3, 5, 14, 15]. In general, these methods are based on the recasting of a problem in the form of an integral equation and application of the Monte Carlo technique for constructing estimates for their solution [1]. The walk on spheres (WOS) method (described in [12]) is based on such an approach. The Green's function first passage Monte Carlo method is the natural extension of WOS. The simulation-tabulation technique [9], and last passage variants of the algorithm [7], further extended the capabilities of stochastic computational methods when applied to solving electrostatics problems.

This paper presents the application of different ideas that come from computational mathematics to the construction of Monte Carlo algorithms for solving boundary value problems that are of interest for physical biochemistry.

The rest of the paper is organized as follows. The next section is devoted to the mathematical statement of the problem. Next we describe the algorithm for the simplest case, and in Section 4 we present the description of the algorithm we use for computing the internal electrostatic energy of a molecule immersed in a dielectric media. The problem of coupling different equations through continuity boundary conditions was never solved via the Monte Carlo method before.

In the last part of the article results of some numerical experiments are presented and, finally, some conclusions and possible further directions are discussed.

2   Mathematical Statement of the Problem

In this section, we consider the problem of calculating the internal energy of a molecule. To be more exact, we will calculate the electrostatic energy – the internal energy for non-bonded electrostatic interactions between atoms constituting a large molecule.

Assume that the molecule in question can be described as a compact set $G \in R^3$ constructed of a large number of intersecting balls (atoms):

$$G = \bigcup_{m=1}^{M} B(x_m, r_m) ,$$

where $B(x_m, r_m) \bigcap \left(\bigcup_{n \neq m}^{M} B(x_n, r_n) \right) \neq \emptyset, m = 1, \ldots, M.$

Every spherical atom has its electrical charge, q_m, which is positioned at its center, x_m, and r_m is the radius of this atom (ball). Hence, the electrostatic potential, $u(x)$, satisfies Poisson's equation (1) inside G for the particular charge density $\rho(x) = \sum_{m=1}^{M} q_m \delta(x - x_m)$. Here, the dielectric permittivity, $\epsilon = \epsilon_i$, is constant.

The potential can be represented as the sum of two functions:

$$u(x) = u^{(0)}(x) + g(x) , \tag{2}$$

where $g(x) = \sum_{m=1}^{M} \frac{q_m}{4\pi\epsilon_i} \frac{1}{|x - x_m|}.$

From (2) and (1) it follows that $\Delta u^{(0)}(x) = 0$ in G, and the boundary values of $u^{(0)}$ are equal to $u(x) - g(x)$.

First suppose that the molecule is grounded. Physically, this means that G is surrounded by an ideally conducting material, and the problem of computing the potential reduces to the interior Dirichlet boundary value problem for Poisson's equation (1):

$$u(y) = 0 , \quad \text{or} \quad u^{(0)}(y) = -g(y) , \ y \in \partial G . \tag{3}$$

In the general case, the molecule is thought of as being immersed in some dielectric (e.g. water). The classical approach is to treat the surrounding medium as continuous with some constant permittivity, ϵ_e. The distribution of dissolved ions determines the charge density, ρ, outside G. From this it follows that the electrostatic potential in the exterior of the molecule also satisfies Poisson's equation. From statistical mechanical considerations, ions should be distributed in accordance with the Boltzmann law. With some assumptions [2],

$$\rho(x) = - \sum_j q_j^{ion} e n_j^0(x) \sinh(q_j^{ion} e \, u(x)/k_b T) , \ x \in G_1 ,$$

where $q_j^{ion} e$ is the charge of j-th type of ion (e is the unit electrical charge), $n_j^0(x)$ is the bulk number density of these ions, k_b is Boltzmann's constant, and T is the absolute temperature (see e.g. [13]).

With this right hand side, equation (1) becomes a non-linear Poisson-Boltzmann equation for the potential, u, in the domain $G_1 \equiv R^3 \setminus \overline{G}$. For small potential values, it may be linearized, thus leading to the equation

$$\Delta u(x) - k^2 u(x) = 0 , \ x \in G_1 , \tag{4}$$

where k is a positive constant (its square, k^2, is known as the Debye-Hückel screening parameter).

Equations (1) and (4) must be coupled by the continuity conditions on the boundary of the molecule:

$$u_i(y) = u_e(y) \ , \ \epsilon_i \frac{\partial u_i}{\partial n(y)} = \epsilon_e \frac{\partial u_e}{\partial n(y)} \ , \ y \in \partial G \ . \tag{5}$$

For convenience, we denote u_i as the solution to (1) in the interior of G, and u_e as the solution of (4) in the exterior of the molecule. We suppose also that $u_e(x) \to 0$ as $|x|$ goes to infinity.

It is a common knowledge (see e.g. [13]) that the expression for electrostatic energy depends on the thermodynamic process that led to the resulting charge distribution. In the case when all response functions are linear, the electrostatic free energy of the molecule is given by

$$E = \frac{1}{2} \sum_{m=1}^{M} u_m q_m \ , \tag{6}$$

where u_m is the non-singular part of the electrostatic potential at the center of the m-th atom. This means that in the calculation of E we exclude the infinite self-energy of the point charges, and take $u_m = u^{(0)}(x_m)$.

3   Monte Carlo Algorithm: Grounded Molecule

To find the energy, E, we construct a Monte Carlo algorithm based on the properties of Brownian motion. Denote by $\xi[E]$ a Monte Carlo estimate for E. We represent it as a weighted sum of estimators for point values of the potential:

$$\xi[E] = \frac{1}{2} \sum_{m=1}^{M} \xi[u_m] \ q_m \ .$$

Construction of every $\xi[u_m] = \xi[u](x_m)$ is based on the simulation of a Markov chain $\{x_m^{(j)}, j = 0, 1, \ldots, N_m\}$ with initial point $x_m^{(0)} = x_m, m = 1, \ldots, M$ and random length N_m.

Here, we shall make use of the strong Markov property of Brownian motion [8]. This makes it possible to avoid the simulation of the whole path, and to jump from an interior point of a domain directly to its boundary. To do this, we have to know the exact distribution of the exit point, or, which is equivalent, the boundary Green's function for this domain. It is clear; however, that there is no chance of using this approach for all of G. Even the walk on spheres approximation cannot be directly applied in this case, since when finding the distance from a point to the boundary of the domain we have to take into account all intricate curves formed by the intersections of the bounding surfaces.

The most efficient way to simulate exit points on ∂G is to use the natural representation of G as a union of intersecting spherical subdomains. The exit

point from every subdomain can be modelled efficiently, either by the walk on spheres method or directly. It was proved [16] that this provides a Monte Carlo estimate that has the same properties as an estimate based on direct simulation of the exit point.

Every subdomain of G can be thought of as a ball in this particular case. By Poisson's formula for a function, u, that satisfies the Laplace equation, at every point $x \in B(x_c, r)$ we have

$$u(x) = \int_{S(x_c, r)} p_p(x \to y) \ u(y) d\sigma(y) \ , \tag{7}$$

where $p_p(x \to y) = \dfrac{1}{4\pi r} \dfrac{r^2 - |x - x_c|^2}{|x - y|^3}$ is the Poisson kernel. If we put $x_m^{(0)} = x_m$ and choose $x_m^{(1)}$ randomly on the spherical surface $S(x_m, r_m)$ in accordance with the density $p_p(x_m^{(0)} \to x_m^{(1)})$, which in this particular case corresponds to the isotropic distribution since $x_m^{(0)}$ lies at the center of the sphere. By definition, $x_m^{(1)}$ either hits the boundary or there exists a number j_1 that $x_m^{(1)} \in S(x_{j_1}, r_{j_1})$. In the latter case we simulate the next point of the Markov chain in accordance with the Poisson kernel density on the surface $S(x_{j_1}, r_{j_1})$. Finally, with probability one, the path of the Markov chain terminates at the boundary after a finite number of steps, N_m.

Consider the sequence $\{u^{(0)}(x_m^{(i)}), i = 0, 1, \ldots, N_m\}$. It forms a martingale, since it is constructed by randomization of (7). Hence, $\mathbb{E}(u^{(0)}(x_m^{(i+1)})|x_m^{(i)}) = u^{(0)}(x_m^{(i)})$, and we have

$$u^{(0)}(x_m) = \mathbb{E} u^{(0)}(x_m^{(N_m)}) \ . \tag{8}$$

In the simplest case of Dirichet boundary values (3), $u^{(0)}$ on the boundary surface can be calculated exactly. Hence, we can set $\xi[u_m] = -g(x_m^{(N_m)})$, which gives us

$$\xi[E] = -\frac{1}{2} \sum_{m=1}^{M} g(x_m^{(N_m)}) \ q_m \ . \tag{9}$$

Proposition 1. *Let G be a compact set constructed from a finite number of intersecting balls. Suppose that the function $u^{(0)}$ is a solution to the Dirichlet problem inside this domain with finite boundary values, and E is a weighted sum of its point values (6).*

Then (9) is an unbiased Monte Carlo estimator for E with finite variance.

Computation of Derivatives

Poisson's formula (7) makes it possible to calculate not only the solution of the Laplace equation but also its derivatives (see ([3])). Let $x = x_c \equiv (x_{(1)}, x_{(2)}, x_{(3)})$. Then

$$\frac{\partial u}{\partial x_{(i)}}(x) = \int_{S(x,r)} \frac{3(y_{(i)} - x_{(i)})}{r^2} p_p(x \to y) \ u(y) d\sigma(y) \ , \ i = 1, 2, 3 \ . \tag{10}$$

From the above and (8) it follows that

$$\frac{\partial u^{(0)}}{\partial x_{(i)}}(x_m) = -\mathbb{E} \; \frac{3(x_{m,(i)}^{(1)} - x_{m,(i)})}{r_m^2} \; g(x_m^{(N_m)}) \; . \tag{11}$$

4   Monte Carlo Algorithm: General Case

Now consider $u^{(0)}$ to be a solution of the Laplace equation with the boundary values $u(x) - g(x)$. Point values of $u^{(0)}$ inside G can be evaluated through relation (8), where, due to the double randomization principle (see e.g. [3]), the unknown numbers $u^{(0)}(x_m^{(N_m)})$ can be substituted by their independent estimates.

By definition
$$\xi[u^{(0)}](y) = \xi[\psi](y) - g(y) \; , \; y \in \partial G \; , \tag{12}$$

where by $\psi(y)$ we mean the unknown boundary value of $u(y)$. To construct a Monte Carlo estimate for ψ, we make use of the boundary condition (5). Let $n(y)$ be the external normal vector at the point $y \equiv y_0$ on the domain boundary. Then, through a finite difference approximation we have

$$\psi(y) = pu(y - hn) + (1 - p)u(y + hn) + O(h^2) \; , \tag{13}$$

where u equals u_i at points inside G, and is equal to u_e in G_1. The coefficient $p = \dfrac{\epsilon_i}{\epsilon_i + \epsilon_e}$ may be thought of as the probability of jumping back inside ($y_1 = y - hn$), whereas with the complementary probability that of the next point being chosen as $y + hn$ (outside). Hence,

$$\psi(y_0) = \mathbb{E}(u(y_1)|y_0) + O(h^2) \; . \tag{14}$$

For internal points we have

$$u(y_1) = \mathbb{E}(u^{(0)}(y_2) + g(y_1)|y_1) \; , \tag{15}$$

where $y_2 \in \partial G$ is the last point of the Markov chain of the walk in subdomains constructed in accordance with the algorithm of the previous section.

To construct a Monte Carlo estimator in the case when $y_1 \in G_1$, we can use the walk on spheres algorithm [3]. Set $y_{1,0} = y_1$. Next, on every step of the algorithm we find the distance from the point, $y_{1,i}$, to the boundary: $d_i = dist(y_{1,i}, \partial G)$. Then the next point is chosen isotropically on $S(y_{1,i}, d_i)$, and so on, until either the distance becomes less than a prescribed number, ε, or the chain is terminated in G_1.

We have [3]

$$u(y_{1,i}) = \mathbb{E}(p_i u(y_{1,i+1})|y_{1,i}) \; , i = 0, 1, \dots, N - 1 \; ,$$

where $p_i = kd_i / \sinh(kd_i) < 1$ can be thought of as a survival probability. Hence,

$$u(y_{1,i}) = \mathbb{E}(Q_i u(y_{1,i+1})|y_{1,i}) \; , \tag{16}$$

where $Q_i = 1$ with probability p_i and $Q_i = 0$ with the complementary probability. If $d_N < \varepsilon$ and $Q_N = 1$ we can set $y_2 \in \partial G$ to be the nearest to $y_{1,N}$. Thus, we have

$$u(y_1) = I\!E(Q_N(\psi(y_2)|y_1) + O(\varepsilon))) . \tag{17}$$

If $Q_N \neq 0$ we return to relations (12), (14), and therefore, simulate a jump from the boundary in accordance with (13).

So, we see that the probability of termination of the constructed Markov chain depends only on kd_i. This means that the larger the Debye-Hückel parameter, k^2, the shorter the chain and the smaller the computational cost of the algorithm. The mean number of steps can be exactly evaluated in the simplest cases. For a sphere or other convex G, the probability of going to infinity is $O(h)$. So, the mean number of points on the boundary that come from outside N_e will be $O(h^{-1})$ if the influence of k is not taken into account. The length of every walk on spheres that returns to the boundary is $O(\log(\varepsilon))$ (see [3]). Hence, the overall number of points in the Markov chain is $O(h^{-1}\log(\varepsilon)f(k))$ where $f(k)$ is a decreasing function of the Debye-Hückel parameter.

The resulting bias in the estimator comes from the boundary condition (5) and (17). Therefore, we have to take $\varepsilon = h^2$ to provide an $O(h)$ bias.

From this it follows

Proposition 2. *Let G be a compact set constructed from a finite number of intersecting balls in such a way that every part of the boundary is reachable for a Brownian motion starting at infinity, with probability proportional to its surface area.*

Suppose that the function $u^{(0)}$ is defined by (2), where u is a solution to the boundary value problem (1) , (4), (5), and g is bounded on ∂G.

Then the linear recurrence (12), (14), (15), (17) defines a δ-biased Monte Carlo estimator for the weighted sum of the potential $u^{(0)}$ point values (6); and the variance of this estimator is finite.

Here $\delta = O((N_e + N_i)h^2 + N_e\varepsilon)$ or $O(h)$ for $\varepsilon = h^2$ (N_i is the number of reflections inside G).

5 Results of Computational Experiments

To test the proposed algorithms we applied the constructed Monte Carlo estimators to solving several simple model problems. For some of them analytical solutions are available. All calculations were carried out on an ordinary desktop computer: a Dell PC with 1.3 GHz P4 processor running Windows 2000.

5.1. We consider first the simplest case, a spherical (one-atom) molecule. The analytical solution scaled by $\dfrac{q^2}{4\pi\epsilon_i R}$ is

$$E = \frac{\epsilon_i}{\epsilon_e(1 + kR)} - 1 .$$

For the chosen parameters $\epsilon_i = 4.0$, $\epsilon_e = 78.5$, $k = 0.104$ one obtains the exact value, -0.9538. Our calculations provided the result -0.9536 in 418 seconds. For

the calculation parameters $\varepsilon = h^2 = 10^{-6}$ and $N = 10000$, both the bias and the statistical error are equal to 9.6×10^{-4} (i.e. 0.1%). (We consider the statistical error to be equal to two standard errors, σ). Note that the computation ensuring one per cent error ($\varepsilon = 10^{-4}$, $N = 120$) takes only 0.4 seconds.

It is clear that the efficiency of the algorithm depends on the parameters of the problem. Table 1 shows the dependence of the computational time on the Debye-Hückel parameter for an accuracy of 0.1%. The ratio of dielectric

Table 1. Normalized energy for different values of k.

k	exact	estimate	N	h	time
0.104	-0.9538	-0.9536	1.0×10^4	1.0×10^{-3}	418
1.04	-0.9750	-0.9753	2.8×10^3	1.9×10^{-3}	30
10.4	-0.9954	-0.9948	4.0×10^2	1.0×10^{-2}	0.2

permittivities is also of great importance. Compare the fact that 0.4 seconds were needed for computing the solution to 1% accuracy when $\epsilon_e = 78.5$ while 272 seconds were required for $\epsilon_e = 7.85$.

5.2. Further, we consider a model molecule constructed of two equal unit spheres with unit charges of opposite signs. The distance between the centers is set to be 1.5. In the case of a grounded molecule there is no bias and 10^6 trajectories ensure the statistical error of 0.1% for the energy. Its value (normalized by $1/4\pi\epsilon_i$), -0.3649, was obtained in 39 seconds. The same statistics provide estimates for the potential and the electrostatic field point values as well. We have $u^0(x_1) = 0.3645$, $u^0(x_2) = -0.3652$ with $\sigma = 0.0002$. The estimates for the only non-zero component of the field at points $X1$ and x_2 give $\partial u^{(0)}/\partial x_{(1)}(x_1) = -0.3548$, $\partial u^{(0)}/\partial x_{(1)}(x_2) = -0.3561$ with $\sigma = 0.0007$.

With a finite permittivity in the exterior we obtained the result -0.3430 with 1% accuracy in 488 seconds.

After testing the algorithm on simple model configurations, we applied it to calculating the energy of real molecules (proteins) constructed from several hundreds of atoms. These computational experiments were conducted in collaboration with biophysicists, and their results will be presented elsewhere.

6   Discussion and further directions

The computational results presented in the previous section show that the constructed Monte Carlo algorithms can be efficiently applied for calculating the internal energy, point values of the electrostatic potential and its derivatives, when 1% accuracy in the result is sufficient.

It is essential to note, however, that it is much simpler for the random walk methods than for conventional deterministic algorithms to take into account the

real complicated geometrical properties of this computational domain. The implementation of the presented stochastic methods is natural and straightforward.

Considering the problem of free energy calculation, we discovered that our method works very well for a grounded molecule and, more generally, in the case when the Debye-Hückel parameter is greater than 1. Clearly, the applicability of the linearized Poisson-Boltzmann equation becomes questionable for large values of k. So, the next step in our investigations will be the extension of our algorithms to the non-linear case.

Also, it would be interesting to eliminate the time-consuming random walks in the exterior of the molecule when k is less than 1. We have some ideas along these lines and are currently working on their development.

Acknowledgements

References

1. Curtiss, J.H.: Monte Carlo methods for the iteration of the linear operators. J. Math. Phys. **32** (1954) 209–232
2. Davis, M.E. and McCammon, J.A.: Electrostatics in biomolecular structure and dynamics. Chem. Rev. **90** (1990) 509–521
3. Elepov, B.S., Kronberg, A.A., Mikhailov, G.A. and Sabelfeld, K.K.: Solution of boundary value problems by the Monte Carlo method. Nauka, Novosibirsk (1980) (in Russian)
4. Ermak, D.L. and McCammon, J.A.: Brownian dynamics with hydrodynamic interactions. J. Chem. Phys. **69** (1978) 1352–1360
5. Ermakov, S.M., Nekrutkin, V.V. and Sipin, A.S.: Random processes for classical equations of mathematical physics. Kluwer Academic Publishers, Dordrecht (1989)
6. Freidlin, M.: Functional integration and partial differential equations. Princeton University Press, Princeton (1985)
7. Given, J.A., Mascagni, M. and Hwang, C.-O.: Continuous path Brownian trajectories for diffusion Monte Carlo via first- and last-passage distributions. In: Margenov, S., Wasniewski, J., Yalamov, P. (eds.). Lecture Notes in Computer Science, Vol. 2179. Springer-Verlag, Berlin Heidelberg New York (2001) 46–57
8. Hida, T.: Brownian motion. Springer-Verlag, Berlin Heidelberg New York (1979)
9. Hwang, C.-O., Given, J.A. and Mascagni, M.: The simulation-tabulation method for classical diffusion Monte Carlo. J. Comput. Phys. **174** (2001) 925–946
10. Kac, M.H.: On some connections between probability theory and differential and integral equations. In: Proc. 2nd Berkeley Symp. on Math. Statist. and Probability. Univ. of California Press, Berkeley (1951) 189–215
11. Kloeden, P.E. and Platen, E.: Numerical solution of stochastic differential equations. Springer-Verlag, Berlin Heidelberg New York (1992)

12. Muller, M.E.: Some continuous Monte Carlo methods for the Dirichlet problem. Ann. Math. Statist. **27** (1956) 569–589
13. Naray-Szabo, G. and Warshel, A. (eds.): Computational Approaches to Biochemical Reactivity. Series: Understanding Chemical Reactivity, Vol. 19. Kluwer Academic Publishers, New York (2002)
14. Sabelfeld, K.K.: Monte Carlo Methods in Boundary Value Problems. Springer-Verlag, Berlin Heidelberg New York (1991)
15. Sabelfeld, K.K. and Simonov, N.A.: Random Walks on Boundary for solving PDEs. VSP, Utrecht (1994)
16. Simonov, N.A.: A random walk algorithm for the solution of boundary value problems with partition into subdomains. In: Methods and algorithms for statistical modelling. Akad. Nauk SSSR Sibirsk. Otdel., Vychisl. Tsentr, Novosibirsk (1983) 48–58 (in Russian)