Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36

TITLE EMPIRICAL RESULTS OF A HYBRID MONTE CARLO METHOD
FOR THE SOLUTION OF POISSON'S EQUATION

AUTHOR(S) Robert E. Hiramoto
Ralph G. Brickner

SUBMITTED TO Applications of Supercomputers in
Engineering '91

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

By acceptance of this article, the publisher recognizes that the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution or to allow others to do so, for U.S. Government purposes.

The Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy.

Los Alamos National Laboratory
Los Alamos, New Mexico 87545

EMPIRICAL RESULTS OF A HYBRID MONTE CARLO METHOD FOR THE SOLUTION OF POISSON'S EQUATION

Robert E. Hiromoto and Ralph G. Brickner

*Computing and Communications Division, Los Alamos National Laboratory,
Los Alamos, New Mexico*

INTRODUCTION

The application of Monte Carlo techniques have considerable importance in solving computational problems that exhibit complex physical interactions and irregular spatial geometries [1,2]. Yet equally important is the large computational parallelism that is inherent in these methods. With the current interest and emphasis on massively parallel computer systems, Monte Carlo methods would seem to be a likely candidate for parallel implementation. However parallelism in itself is only one metric for parallel computational performance. Measures such as load-balancing, data locality, communication latency, and numerical accuracy are equally important metrics for the evaluation of a parallel numerical algorithm.

In this paper, a highly parallel Monte Carlo technique is modified and used in estimating the solution to Poisson's equation. This simple modification incorporating a more efficient use of boundary data and appears to yield a better estimate to the exact solution. As a by product, the new approach is a more efficient sequential algorithm but at the expense of sacrificing parallelism.

DESCRIPTION

We consider the solution of a two-dimensional Poisson equation with Dirichlet boundary conditions given by:

$$\begin{aligned}\nabla^2 \phi(x,y) &= -\rho(x,y), \\ \phi(x_0,y_0) &= f(x_0,y_0).\end{aligned}\tag{1}$$

Only problems with the Dirichlet boundary conditions are discussed here, but the method may be extended to other boundary conditions [3]. Using the five-point finite difference stencil, the discretized form of Equation (1) is given by

$$\phi_{ij} = \frac{1}{4} (\phi_{i-1j} + \phi_{ij-1} + \phi_{i+1j} + \phi_{ij+1} + \frac{\rho_{ij}}{h^2}). \quad (2)$$

Here, the spacing in x and y are both equal to some small grid spacing h .

For Eqn. (2) and correspondingly higher order differencing schemes, the potential at location (i, j) is determined from the average value of the potentials at the appropriate neighboring points. Since the coefficients of the right hand side sum to 1, a probabilistic formulation would prescribe an estimate of ϕ_{ij} to be made by randomly sampling points within the solution domain with a given probability (in this case a uniform probability of 1/4 in each of four directions) to move from one point to any other neighboring point. There are several Monte Carlo techniques discussed in the literature [4,5,6,7,8] that may be used to solve Poisson's equation. We choose to examine a standard Monte Carlo technique [1] that requires sampling the solution domain by means of discrete random walks that score the value of the boundary where the walks terminate. For non-zero ρ , one also scores the value of $\frac{\rho_{ij}}{4h^2}$ at every interior point (i, j) reached during the walk. This entails what is known as a "primary estimate" of the value of ϕ at point (i, j) . For N primary estimates, the arithmetic mean of the N primary estimates gives the final estimate.

In this standard method, estimates for the solution of Poisson's equation at the point (i, j) requires a "particle" or "random walks" to be initialized with its position being equal to the cell indices (i, j) and two tally bins. One bin scores $\frac{\rho_H}{4h^2}$ at each interior point (k, l) of the walk, the second scores the value of $\phi(x_0, y_0)$ when the particle reaches a boundary. The interior tally is initialized with ρ at (i, j) ; the boundary tally is set to zero. As in other Monte Carlo simulations, the use of a pseudo-random number generator determines which of four possible directions the particle is to take for its "step." If the step takes the particle to an interior point, the value for the charge density ρ is tallied at this point. The particle then continues its random walk from this point. If the new cell is a boundary point, the value of ϕ on the boundary is scored in the boundary tally and the particle history terminates. The tallies are multiplied by appropriate factors, giving an estimate for ϕ at this point. To achieve sufficient accuracy to the solution, a large number N of particles are issued from point (i, j) .

The expression for ϕ_{ij} is given as follows:

$$\phi_{ij} = \sum_{n=1}^N \phi_{I(n), J(n)} + \frac{1}{4h^2} \sum_{n=1}^N \sum_{m=1}^{M(n)} \rho_{I'(n, m), J'(n, m)} \quad (3)$$

where N is the number of random walks started from (i, j) , $M(n)$ is the number of cells the N 'th walk comprises, $(I(n), J(n))$ are the cell coordinates of the boundary on which the N 'th walk terminates, and $(I'(n, m), J'(n, m))$ are the cell coordinates of the M 'th cell through which the N 'th walk passes.

The standard Monte Carlo (MC) method may be regarded as a Jacobi-like scheme where the estimate for each interior grid point is determined without reference to estimates obtained at other points. The standard MC method, therefore, exhibits a high degree of parallelism, and appears as a rather good candidate for implementation on a massively parallel system. Yet as a Jacobi-like scheme, some inefficiencies exist in converging the solution to some acceptable accuracy. Standard iterative schemes such as Gauss-Seidel mix both old and new iterate values while other schemes in addition use gradient information to significantly reduce the number of iterations required for convergence over the pure Jacobi scheme. Without major modifications to the standard MC scheme described above, a Gauss-Seidel-like scheme is developed with the hope of minimizing both the number of random walks required to reach the boundary and the total number of particles required to obtain a comparable or better converged solution.

The new technique "propagates" the boundary information into the interior of the solution domain. The procedure consists of initiating random walks from only those grid points adjacent to the boundary (see Fig. 1). When all random walks have terminated, the original boundary is logically replaced by the boundary formed by those points where the solution has just been estimated (i.e., the adjacent points). The procedure continues in this manner where the boundary, incrementally replaced by the solution estimates at the corresponding adjacent points, is in effect "propagated" into the center of the solution domain. We refer to this as a Monte Carlo boundary propagation (MCBP) method. Figure 1 shows the "propagation" of the boundary for a few "iterations." Because each step of the boundary propagation depends on the result of the previous boundary calculation, it is also clear that MCBP exhibits less computational parallelism as compared to the standard MC technique described above.

Fig. 1. Phases of the boundary propagation method.

RATE OF CONVERGENCE

The average number of steps per random walk per mesh point (ω) required to exit a 2-dimensional (2-D) solution domain of size $L \times L$ is determined empirically for both MCBP and the standard MC method. The values of mesh length L range from four (4) to 256. From this data, a least square fit is applied to ω as a function of L .

Following the argument made by R.P. Feynman [9], it can be shown that the average number of steps required by the standard MC method to exit the domain is of $O(L^2)$. A least squares fit of ω applied to αL^2 gives:

$$\omega_{MC} = 0.14 L^2. \quad (4)$$

For a sample size of $N = 50k$ random walks per mesh point, Table I. lists both the measured (ω) and computed (ω_{MC}) values for the average number of steps required to reach the mesh boundary.

TABLE I: Standard MC						
L	4	8	16	32	64	128
ω	2.00	8.19	33.98	139.46	566.30	2282.76
ω_{MC}	2.23	8.91	35.65	142.60	570.38	2281.52

For the boundary method, the functional form of ω is observed to be directly proportional to the length of the mesh size L . Using this fact, a least squares fit of ω applied to αL gives:

$$\omega_{MCBP} = 0.37 L. \quad (5)$$

From Eqn. 5, Table II. lists both the measured (ω) and computed (ω_{MCBP}) values for the average number of steps required to reach the mesh boundary.

TABLE II: MCBP							
L	4	8	16	32	64	128	256
ω	1.44	3.05	6.12	12.14	24.08	47.97	95.63
ω_{MCBP}	1.50	2.99	5.98	11.97	23.94	47.87	95.74

RESULTS

For the test problem, the charge density on the right hand side of Poisson's equation is chosen to be

$$\begin{aligned}\rho(x,y) &= \sin(x)\sin(y), \\ \phi(x_0,y_0) &= 0.\end{aligned}\tag{6}$$

The convergence for the boundary propagation method is studied by running the test problem with different number of samples (or particles) per grid point. The estimated value of the potential is compared against the exact solution. The test problems are solved with mesh sizes of 32×32 and 64×64 (where the boundaries are included in the mesh size). Table III. shows the execution times for the standard MC and the MCBP method on a Sun 3 workstation. The times are measured for an example using 100 random walks initiated from each mesh point.

TABLE III: Execution times for MC vs. MCBP.			
L	MC time (s)	MCBP time (s)	Ratio Mc to MCBP
32	3,981.0	349.6	11.39
64	67,675.6	2,967.8	22.80

A least squares fit of the function $\alpha/\sqrt{N}$, is applied to the maximum error measured for the various grid sizes. Figure 2 shows the correspondence (for $\alpha = 0.82$) between the measured data and the fitted curve. Figures 3 and 4 show the difference in maximum error between the two methods for the same number of random walks, and the norm of the errors for the respective mesh sizes.

Fig. 2. Maximum error as a function of N .

Fig. 3. Comparison of the maximum errors.

Fig. 4. Comparison of the error norms.

Noting that the longest duration of random walks for the MCBP method corresponds to the shortest duration of random walks for the standard MC method (that is, for those walks that start at the grid points adjacent to the original boundary), the boundary method is significantly more efficient than the more parallel (Jacobi-like) method. In fact, in this study the MCBP method is found to be of $O(L)$ times faster. Furthermore for the same number of samples per grid point, the boundary propagation method achieved better estimates of the solution for the example problem studied.

REFERENCES

1. Hammersley, J.M. and D.C. Handscomb, Monte Carlo Methods, Methuen, London (1964).
2. Buslenko, N.P., et al., The Monte Carlo Method: The Method of Statistical Trails, edited by Yu.A. Shreider, Pergamon Press, New York, (1966).
3. Bevensee, R.M. Applications of Probabilistic Potential Theory (PPT) to the Solutions of Electromagnetic Problems, Preprint UCRL-82760, Lawrence Livermore Laboratory, June 1, 1979.
4. Haji-Sheikh, A. Application of Monte Carlo Methods to Thermal Conduction Problems, Ph.D. thesis, University Microfilms, Inc., Ann Arbor, Michigan (1965).
5. DeLaurentis, J.M. and L.A. Romero, A Monte Carlo Method for Poisson's Equation, Journal of Computational Physics, Vol. 90, No. 1, pp 123-140 (Sept. 1990).
6. Booth, T.E. Exact Monte Carlo Solution of Elliptic Partial Differential Equations, Journal of Computational Physics, Vol. 39, No. 2, pp 396-404 (Feb. 1981).
7. Booth, T.E. Regional Monte Carlo Solution of Elliptic Partial Differential Equations, Journal of Computational Physics, Vol. 47, No. 2, pp 281-290 (Aug. 1982).
8. Mascagni, M. A Monte Carlo Method Based on Wiener Integration for Solving Elliptic Boundary Value Problems, submitted to the SIAM Journal on Scientific and Statistical Computing.
9. Feynman, R.P. Lecture on Physics, Vol.I, Section 41-4, Addison-Westley, Reading, MA., (1963).