JOURNAL OF COMPUTATIONAL PHYSICS 107, 118–123 (1993)

Multi-grid Methods for Steady State Diffusion in Random Media

DIETRICH BRAESS AND MICHAEL BIEBIGHÄUSER

Institute for Mathematics, Ruhr-University Bochum, D-4630 Bochum, Germany

AND

PETER GRASSBERGER AND RICARDO LEUVERINK

Physics Department, University of Wuppertal, D-5600 Wuppertal 1, Germany

Received March 4, 1991

We apply multi-grid methods to study steady state densities for hopping diffusion in two-dimensional random media. We show that these methods are very efficient. For our fastest algorithm, which also uses overrelaxation, CPU times increase as $L^{2.2}$ in order to reach equilibrium on square lattices of size $L \times L$ with a prescribed accuracy. Compared to standard Gauss–Seidel methods, this is at least an improvement by a factor $\propto L$. For $L = 128$, the improvement is about a factor of 10 and even more than 80 if overrelaxation is added.

1. INTRODUCTION

Random walks in random media have been the subject of numerous recent papers [1, 7–9, 12–14, 16, 19]. Interesting observables are the fluctuations of the density in the steady state and transport properties as the mean drift velocity.

In one-dimensional space it is known that random uncorrelated and unbiased forces lead to very slow diffusion, with $\Delta x \sim \log^2 t$ instead of $\Delta x \sim \sqrt{t}$ as in the non-random case [19]. Spatial density correlations in the steady state reached at $t \to \infty$ decay exponentially in this case [9], a situation called *localization* in the physics literature [18, 20].

In $\geqslant 2$ dimensions, the results are conjectured to depend on the nature of the force field. If the field is derived from a potential and if the latter is a Brownian function over $\mathbb{R}^2$, we should again obtain localization and $\Delta x \sim \log^2 t$ [5, 16]. Finally, in two dimensions and for isotropic force–force correlations, one expects $\Delta x \sim \sqrt{t}$ with logarithmic corrections [8, 13, 14]. The corresponding equilibrium distributions seem to be multifractal (i.e., in the continuum limit they seem to be fractal measures concentrated on a dense set in R^2 with Hausdorff dimension strictly less than two [9, 17]).

The last result was obtained only numerically by using a standard Gauss–Seidel algorithm for a (discrete) hopping model on a square lattice. The convergence of this method is very slow. It is roughly as slow as in the computation of the solutions for the Laplace equation (which corresponds to the non-random homogeneous case). In some cases, due to the inhomogenity of the hopping rates, convergence is slightly better, while in other cases the contrary is observed. Overrelaxation provides some improvement, but the computational situation is still unsatisfactory.

Therefore we decided to study the problem by means of multi-grid methods. When doing this we cannot perform the transfer between the grids in the way known from the treatment of partial differential equations. We will modify the restriction and prolongation operators and obtain a method which remains also stable when the hopping rates vary very much. In this way we indeed substantially improve the convergence to the steady-state solution.

2. FORMAL DEVELOPMENT

2.1. *The Problem*

Our configuration space is a square lattice of size $L \times L$ with periodic boundary conditions. In order to be able to use multi-grid methods, we assume L to be a multiple of a suitably high power of 2. The hopping dynamics are described by a random Markov matrix, where $W_{\mathbf{i},\mathbf{k}}$ is the probability for a particle at site $\mathbf{k}$ to jump to site $\mathbf{i}$ in one time step. The matrix elements $W_{\mathbf{i},\mathbf{k}}$ are positive, if $\mathbf{i}$ and $\mathbf{k}$ are neighbours in the lattice. They are positive random variables and time independent. The other elements of $\mathbf{W}$ vanish and

$$\sum_{\langle \mathbf{i},\mathbf{k} \rangle} W_{\mathbf{i},\mathbf{k}} = 1 \qquad \forall \mathbf{k}. \tag{1}$$

Here, the sum runs over $\mathbf{i}$ and includes the four neighbours of $\mathbf{k}$.

0021-9991/93 $5.00

Equation (1) expresses the fact that particles jump at each time step with probability 1 to one of their neighbouring sites. Apart from this constraint, the $W_{i,k}$'s are independently distributed, and the system is statistically homogeneous in the sense that the distribution is the same for all sites. Also, the $W_{i,k}$'s are time-independent, i.e., they correspond to a "frozen" randomness of the substrate. Finally, we assume that they give rise to ergodic diffusion processes with probability 1. That means that they have for nearly all realizations a non-degenerate unit eigenvalue when considered as a matrix as in Eq. (2) below. In the simulations shown below, we obtained them by drawing for each site four pseudorandom numbers r_j uniformly distributed in the interval $[0.1, 0.9]$, and we put the hopping rate for the jth direction equal to $r_j/\sum_{k=1}^{4} r_k$.

For any realization of $\mathbf{W}$, we want to solve the master equation

$$p_i = \sum_k W_{i,k}\, p_k, \tag{2}$$

where $\mathbf{p}$ is a positive vector with normalization

$$\sum_i p_i = 1. \tag{3}$$

In vector notation, we write Eqs. (1)–(3) as

$$\mathbf{p} = \mathbf{W}\mathbf{p} \tag{4}$$

with

$$\mathbf{e}^{\mathrm{T}}\mathbf{p} = 1, \qquad \mathbf{e}^{\mathrm{T}}\mathbf{W} = \mathbf{e}^{\mathrm{T}}, \tag{5}$$

where we use $\mathbf{e}^{\mathrm{T}} = (1, \ldots, 1)$. The vector $\mathbf{p}$ is the (normalized) eigenvector of $\mathbf{W}$ with positive components. We know that a positive eigenvector with eigenvalue one exists since (5) implies that $\mathbf{W}^{\mathrm{T}}$ has the eigenvalue one and that its spectral radius also equals one.

For later convenience, we shall also consider the inhomogeneous problem where Eq. (4) is replaced by

$$\mathbf{p} = \mathbf{W}\mathbf{p} + \mathbf{b}, \tag{6}$$

and the vector $\mathbf{p}$ need neither be normalized nor positive. Problems with $\mathbf{b} \neq 0$ enter into the computation on coarse grids when multi-grid methods are applied. In order that a solution of Eq. (6) exists, the vector $\mathbf{b}$ has only to satisfy $\mathbf{e}^{\mathrm{T}}\mathbf{b} = 0$.

Numerical convergence will be tested for the original equation (2). The criterion is

$$\max_i \frac{|p_i - \sum_k W_{i,k}\, p_k|}{p_i} < \varepsilon$$

with ε typically in the order of 10^{-6}.

A straightforward method to solve Eq. (6) is via the Jacobi iteration $\mathbf{p}^{(k+1)} = \mathbf{W}\mathbf{p}^{(k)} + \mathbf{b}$ after adding a damping factor, or the Gauss–Seidel version of it. More generally, we write

$$\mathbf{p}^{(k+1)} = \mathscr{S}\mathbf{p}^{(k)}, \tag{7}$$

with the goal of computing $\mathbf{p} = \lim_{k\to\infty} \mathbf{p}^{(k)}$. The symbol $\mathscr{S}$ is used since the *smoothing properties* of the classical iterative procedures will play an important role. To be more precise, the error $\mathbf{p} - \mathbf{p}^{(k+1)}$ is smoother than $\mathbf{p} - \mathbf{p}^{(k)}$. In the following, we will always use a symmetric Gauss–Seidel algorithm for $\mathscr{S}$. A natural choice for the initial distribution is $\mathbf{p}^{(0)} = \mathbf{e}/L^2$.

In the symmetric Gauss–Seidel method, forward and backward sweeps are applied alternatingly [22]. It is more efficient than the standard Gauss–Seidel algorithm since computation of hopping fluxes can be simplified: the backward fluxes during the forward sweep are the same as those during the next backward sweep, while the forward fluxes are the same as during the previous backward sweep. If the method is used as a classical iteration, it has the same shortcomings as the standard Gauss–Seidel iteration when L is very large. In order to obtain convergence within the above limit, the average number of sweeps was found to increase almost as L^2, thus giving a CPU time $\sim L^4$.

2.2. *A Multi-grid Method*

The main idea of multi-grid methods [6, 10] can be successfully applied here. Only details on *small length scales* have to be solved on the original lattice, while *long wavelength components* may be computed from approximations on coarser grids. In particular, let $\mathbf{p}'$ be the approximation for the solution of Eq. (6) which is obtained after some smoothing steps (7). Set

$$\mathbf{p} = \mathbf{p}' + \mathbf{y}. \tag{8}$$

Then $\mathbf{y}$ is a smooth vector which satisfies

$$\mathbf{y} = \mathbf{W}\mathbf{y} + \mathbf{d}, \tag{9}$$

where the *defect* $\mathbf{d}$ is computed by

$$\mathbf{d} = \mathbf{b} + \mathbf{W}\mathbf{p}' - \mathbf{p}'. \tag{10}$$

Equation (9) is now approximated by a system

$$\mathbf{y}^g = \tilde{\mathbf{W}}^g \mathbf{y}^g + \mathbf{d}^g. \tag{11}$$

on a lattice of size $L/2 \times L/2$. Here and in the following we denote all quantities on the coarse lattice by a superscript g.

The transition towards a coarser lattice is called

restriction (or *coarse-graining*, *blocking*, or *decimation* in various other contexts). It applies to the defect

$$\mathbf{d}^g = \mathscr{R}\mathbf{d}.$$

The inverse transition to the finer lattice, i.e., to the original one is called *prolongation*

$$\mathbf{y} = \mathscr{P}\mathbf{y}^g.$$

For diffusion problems with smoothly varying hopping rates, one may adopt the multi-grid algorithms which are known for the treatment of differential equations. Since the hopping rates arise from a random process and may vary very strongly, we choose a slightly different approach.

The points of the $L \times L$ lattice are divided into $L/2 \times L/2$ blocks of four lattice points. Each block corresponds to one point of the coarse lattice. Formally, let $\mathbf{j}$ be a vector on the $L/2 \times L/2$ lattice. We write

$$\mathbf{i} \in \mathbf{j} \qquad \text{if} \quad j_x = \lfloor i_x/2 \rfloor, \quad j_y = \lfloor i_y/2 \rfloor.$$

For any probability vector $\mathbf{y}$ and any block $\mathbf{j}$ let $y_\mathbf{j}^g$ be the sum of probabilities for the block $\mathbf{j}$. With this, we obtain the restriction

$$\mathscr{R}: d_\mathbf{j}^g = \sum_{\mathbf{i} \in \mathbf{j}} d_\mathbf{i}.$$

Our transfer from the coarse to the fine grid will be deduced from the hypothesis that the values of $\mathbf{y}$ for the points within each block are (approximately) identical. This defines the prolongation

$$\mathscr{P}: y_\mathbf{i} = \tfrac{1}{4} y_\mathbf{j}^g, \qquad \mathbf{i} \in \mathbf{j}. \tag{12}$$

In the same spirit we obtain for the hopping rate from block $\mathbf{j}$ to block $\mathbf{l}$

$$\tilde{W}_{\mathbf{j},\mathbf{l}}^g := \tfrac{1}{4} \sum_{\mathbf{i} \in \mathbf{j}} \sum_{\mathbf{k} \in \mathbf{l}} W_{\mathbf{i},\mathbf{k}}.$$

Note that the rate $\tilde{W}_{\mathbf{j},\mathbf{l}}^g$ for hopping within the 2×2-block $\mathbf{j}$ may be positive. For each $\mathbf{j}$ we have now five positive entries $\tilde{W}_{\mathbf{j},\mathbf{l}}^g$. Specifically,

$$\tilde{W}_{\mathbf{j},\mathbf{l}}^g > 0 \qquad \text{only if} \quad \langle \mathbf{i}, \mathbf{j} \rangle \text{ or } \mathbf{i} = \mathbf{j}.$$

An advantage of this approach is the fact that $\tilde{\mathbf{W}}^g$ is again a stochastic matrix, i.e.,

$$\sum_\mathbf{j} \tilde{W}_{\mathbf{j},\mathbf{l}}^g = 1 \qquad \text{for each } \mathbf{l}.$$

Summarizing, an approximate solution $\mathbf{y}_{\text{corr}}$ of Eq. (9) on the fine level is obtained from

$$\mathbf{y}_{\text{corr}} = \mathscr{P}\mathbf{y}^g,$$

where $\mathbf{y}^g$ is the solution of the equation

$$\begin{aligned} \mathbf{y}^g &= \tilde{\mathbf{W}}^g \mathbf{y}^g + \mathbf{d}^g, \\ \mathbf{d}^g &= \mathscr{R}\mathbf{d}. \end{aligned} \tag{13}$$

Note that Eq. (13) only provides an approximation. It would yield the exact $\mathbf{y}$ if all $y_\mathbf{j}$'s would be identical for the points within each block.

It would have been a poor strategy to compute a coarse grid approximation $\mathbf{p}^g = \tilde{\mathbf{W}}^g \mathbf{p}^g + \mathbf{b}^g$ for the solution of the original problem. By the decomposition (8), we instead determine only the smooth part of $\mathbf{p}$ on the coarse lattice, and keep the non-smooth part obtained on the fine lattice. The procedure is called *correction scheme* in the multi-grid literature [6].

Before we provide a complete description of the algorithm, we want to simplify Eq. (13). Let $\mathbf{D}$ be the diagonal portion of the matrix $\tilde{\mathbf{W}}^g$, i.e.,

$$D_{\mathbf{j},\mathbf{l}} := \tilde{W}_{\mathbf{j},\mathbf{j}}^g \delta_{\mathbf{j},\mathbf{l}}.$$

Substract $\mathbf{D}\mathbf{y}^g$ on both sides of Eq. (13)

$$(\mathbf{I} - \mathbf{D})\,\mathbf{y}^g = (\tilde{\mathbf{W}}^g - \mathbf{D})\,\mathbf{y}^g + \mathbf{d}^g. \tag{14}$$

We set $\mathbf{z}^g := (\mathbf{I} - \mathbf{D})\,\mathbf{y}^g$ and obtain

$$\mathbf{z}^g = \mathbf{W}^g \mathbf{z}^g + \mathbf{d}^g, \tag{15}$$

where

$$\mathbf{W}^g := (\tilde{\mathbf{W}}^g - \mathbf{D})(\mathbf{I} - \mathbf{D})^{-1}$$

is again a stochastic matrix. Moreover, $\mathbf{W}^g$ has zero entries in its diagonal, i.e., there are only four nonzero entries in each row. Consequently, $\mathbf{W}^g$ has the same structure as the given matrix $\mathbf{W}$.

Up to now we have described a two-level algorithm. In actual computations the problems on the coarse level are not solved by a direct method but only approximately by again going to a coarser lattice. Eventually, we stop on some lattice with less than 20×20 points and look for exact solutions.

Using this idea an improved solution $\mathbf{p}'''$ of Eq. (6) is provided by the following pseudocode. The number of smoothing steps is specified by ν:

```
FUNCTION SOLVE(p, W, b, L)
  p' = S^ν p                          pre-smoothing
  d = b − p' + Wp'                    compute the defect
  d^g = R d                           restrict the defect;
  IF L < L_min THEN
    solve-exact y'^g = W^g y'^g + d^g   Gauss–Seidel or LU-decomposition
  ELSE
    z^g = d^g                         use defect as a first approx. to z
    z'^g = SOLVE(z^g, W^g, d^g, L/2)  apply SOLVE on L/2-grid [REC]
    y'^g = (I − D)^{-1} z'^g          normalize to unit hopping rates
  ENDIF
  y' = P y'^g                         prolongate
  p'' = p' + y'                       add new defect
  RETURN p''' = S^ν p''               post-smoothing
END
```

This algorithm describes the version which is usually called a V-cycle [4, 6]. A W-cycle is obtained when the recursion step in the algorithm marked by [REC] is applied twice; i.e., $\mathbf{z}'^g$ is obtained after two SOLVE-steps on the coarser level.

The complete algorithm consists then in starting from some distribution $\mathbf{p}^{(0)}$ which in the simplest case is the uniform distribution $\mathbf{p}^{(0)} = \mathbf{e}/L^2$, and using SOLVE to pass from $\mathbf{p}^{(k)}$ to $\mathbf{p}^{(k+1)}$. Since the normalization of $\mathbf{p}^{(k)}$ is not preserved during the iteration, we have to normalize at the end,

$$\mathbf{p} = \lim_{k \to \infty} \frac{\mathbf{p}^{(k)}}{\mathbf{e}^{\mathrm{T}} \mathbf{p}^{(k)}}. \tag{16}$$

3. VARIANTS OF THE MULTI-GRID ALGORITHM

1. The idea of the multi-grid method may be used with other smoothing operators, restrictions, and prolongations. Specifically, we have tried transfer operators which are standard in the treatment of elliptic differential equations. In this case the coarse grid gives directly the lattice points whose coordinates are even numbers, and the other values for the fine grid are computed by linear interpolation:

$$\mathscr{P}: \begin{cases} y_{2j_1, 2j_2} = y^g_{j_1, j_2} \\ y_{2j_1+1, 2j_2} = (y^g_{j_1, j_2} + y^g_{j_1+1, j_2})/2 \\ y_{2j_1, 2j_2+1} = (y^g_{j_1, j_2} + y^g_{j_1, j_2+1})/2 \\ y_{2j_1+1, 2j_2+1} = (y^g_{j_1, j_2} + y^g_{j_1, j_2+1} + y^g_{j_1+1, j_2} \\ \qquad + y^g_{j_1+1, j_2+1})/4. \end{cases} \tag{17}$$

The associated restriction is the adjoint operator. If we used only two or three levels, this transfer turned out to be more efficient. On the other hand, the matrices for the coarse grids are no longer stochastic matrices, i.e., often matrices with negative entries are generated. In particular, we observed a loss of stability when many levels but only a small number of smoothing steps were used [2].

The stability was regained when the *incomplete LU-decomposition* was used for smoothing (at least on the coarser grids). The standard LU-decomposition for Gaussian elimination leads to a loss of sparsity when applied to matrices with a small number of nonzero entries. If the fill-in is suppressed, a procedure is obtained called *incomplete LU-decomposition* [21]. It is slightly more expensive than the SSOR-relaxation but more efficient for anisotropic problems.

2. As another alternative we tried a more sophisticated transfer. We abandoned the hypothesis that the corrections for the points within each block are identical, cf. (12). We replaced it by the hypothesis that the corrections (within each block) are proportional to the actual probabilities. Moreover, we thought in terms of multiplicative instead of additive corrections. This leads to

$$\tilde{W}^g_{\mathbf{j},\mathbf{l}}\, p^g_{\mathbf{l}} = \sum_{\mathbf{i} \in \mathbf{j}} \sum_{\mathbf{k} \in \mathbf{l}} W_{\mathbf{i},\mathbf{k}}\, p_{\mathbf{k}}.$$

Indeed, this gave better convergence in terms of cycles. But improvement in terms of CPU time was moderate, since the repeated updating of the restricted hopping matrix was very time consuming.

3. As was pointed out, classical overrelaxation methods converge very slowly. Nevertheless, sometimes overrelaxation may improve the efficiency of multi-grid algorithms, cf. [3, 11]. To be precise, we may replace the coarse-grid correction

$$\mathbf{p}'' = \mathbf{p}' + \mathbf{y}',$$

by

$$\mathbf{p}'' = \mathbf{p}' + \omega \mathbf{y}' \tag{18}$$

with an *overrelaxation factor* $\omega > 1$.

We will return to be the last variant in the discussion of the results while the previous variants seem to be less important and will no longer be considered.

4. RESULTS

In the following we will present results of simulations with the distribution of hopping matrix elements as described in Section 2.1. We have also studied other distributions, but the results were qualitatively similar and will not be discussed.

Our numerical simulations on a VAXstation 2000 were done with lattices of size up to 192×192. So we treated 36864×36864-matrices with ca. 180000 non-zero entries. The coarsest grid typically had 6×6 sites. As expected, the results did not depend much on whether we solved the auxiliary problems on this scale exactly or by repeated

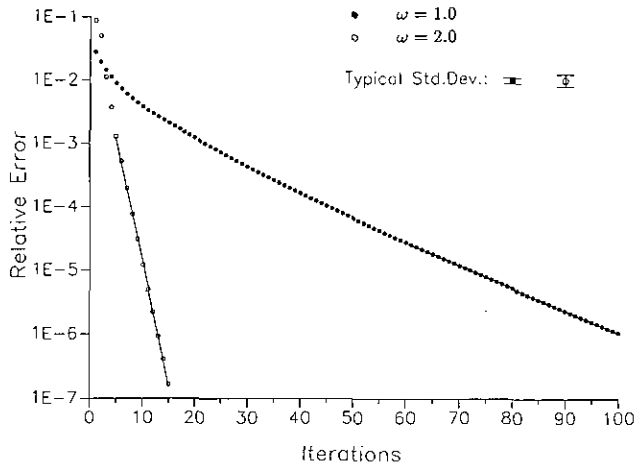

FIG. 1. Convergence for the multi-grid method without (•) and with (∘) overrelaxation. Lattice size was $L = 128$, overrelaxation parameter was $\omega = 2$ (see Eq. (18)), results are averaged over 50 configurations.

application of a Gauss–Seidel algorithm. They also did not depend very much on the depth of the recurrence, whenever it involved more than three levels. Using W-cycles gave, of course, faster convergence per cycle, but again the CPU time did not depend much on whether we used W or V cycles.

Indeed, systematic investigations of CPU times were made difficult by the very large sample-to-sample fluctuations. For instance, in 20 runs on a lattice of size 128×128, the errors (defined by Eq. (7)) after 12 iterations had an average value $\varepsilon = 1.46 * 10^{-3}$ and a standard deviation $\Delta\varepsilon = 4.8 * 10^{-4}$. The variant of the algorithm which gave the best convergence changed from sample to sample.

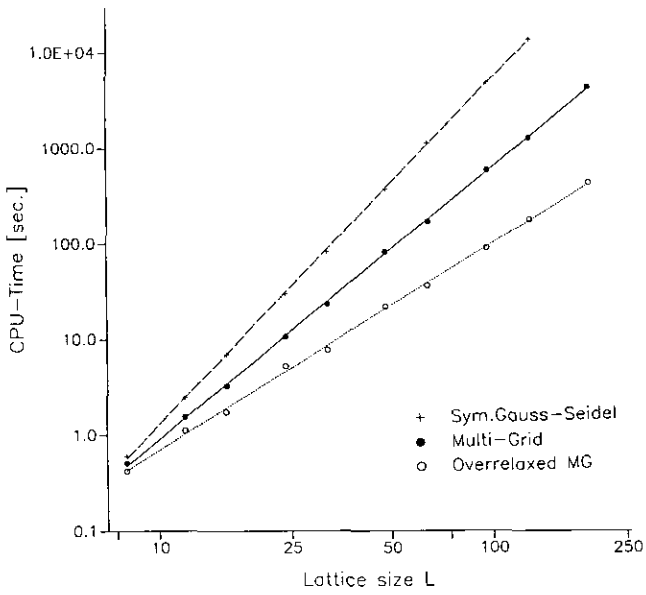

FIG. 2. Average CPU times (on a VAXstation 2000) needed to obtain $\varepsilon = 10^{-6}$, on a log-log scale versus the lattice size L: broken line, symmetric Gauss–Seidel; full line, multi-grid; dotted line, overrelaxed multi-grid with overrelaxation factor $\omega = 2$ (20 configurations).

We found, however, for all samples that the multi-grid method was much faster than the classical Gauss–Seidel algorithm. Asymptotically the errors seemed to decrease exponentially with the number k of cycles (see Fig. 1)

$$\varepsilon \approx e^{-\alpha k}.$$

The exponent α decreased roughly as $1/L$ with increasing lattice size. If this were exact, it would mean that the CPU time to yield a prescribed accuracy would increase as L^3. This is indeed found approximately (see Fig. 2 which shows also a comparison with results with Gauss–Seidel iteration), although the improvement over the Gauss–Seidel algorithm seems closer to a factor $\sim L^{0.7}$ than to $\sim L$.

The observation that the error reduction factor per cycle $e^{-\alpha}$ tends to one with $L \to \infty$ indicates the difference between the present problem and that in the treatment of elliptic problems, where more regular matrices are encountered. We think that this reflects the different diffusion behaviour discussed in the introduction.

The improvement of the multi-grid algorithm when overrelaxation as in (18) was applied is in accordance with this observation. We found that convergence for our random inhomogeneous problems was much enhanced for $\omega > 1$. Best convergence was obtained with ω near 2. For $\omega > 2$, the convergence deteriorated quickly. Again there were large sample-to-sample fluctuations, and the best overall results were obtained with $\omega \approx 1.8 - 2.0$ (see Figs. 1 and 3). The data for $L \leqslant 128$ gave an error reduction factor per cycle of $e^{-\alpha} < 0.37$. The CPU time increases now only as $L^{2.2}$.

If, on the other hand, we encounter inhomogeneous hopping rates, which are not completely at random but had some periodic behaviour, overrelaxation factors close to one are optimal. This reflects the closer similarity to the situation known from partial differential equations.

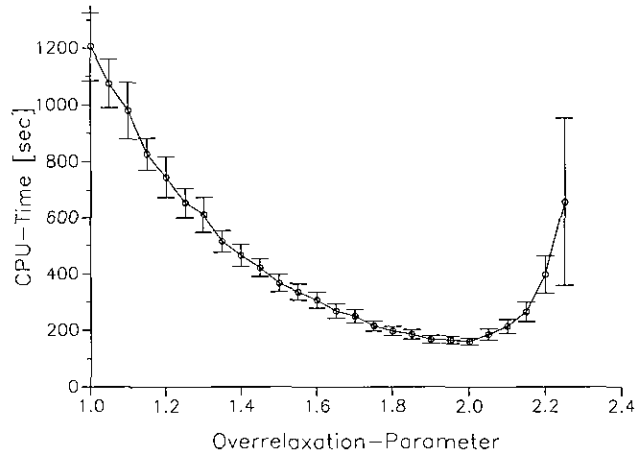

FIG. 3. Dependence of the average CPU time on the overrelaxation-factor ω. Lattice size was $L = 128$ (20 configurations).

5. CONCLUSIONS

The simplest choice of restriction already gave very good multi-grid results for the problem of diffusion with random hopping rates in two dimensions. For this problem, none of the classical methods works efficiently: direct matrix inversion is almost impossible on a workstation for the system sizes we are interested in, while straightforward overrelaxation methods are too slow. Multi-grid methods provide very good results in a small number of cycles.

Locally slow diffusion due to trapping make the multi-grid computations slightly slower than for comparable elliptic problems. This phenomenon is even a bigger problem in one dimension. But in that case, the exact stationary distribution can be determined very easily by other methods, and multi-grid methods are not needed [9].

In addition to the algorithm presented above, we have also tried more sophisticated restriction schemes [10]. Although they gave faster convergence indeed in terms of cycles, the improvement in terms of CPU time was not so evident.

A substantial improvement over the most straightforward multi-grid algorithm was obtained by additional over-relaxation. With this version we obtained roughly another magnitude of improvement for lattices with $L \approx 128$. Overrelaxation factors $\omega \sim 1.8$ seem to be optimal if the hopping rates are random. The present algorithm enabled us to repeat the calculations of Refs. [9, 15] with higher precision and to confirm the results.

ACKNOWLEDGMENTS

This work was supported by the Deutsche Forschungsgemeinschaft, SFB 238. Moreover, the authors would like to thank the referees for their comments.

REFERENCES

1. S. Alexander, J. Bernasconi, W. R. Schneider, and R. Orbach, *Rev. Mod. Phys.* **53**, 175 (1981).
2. M. Biebighäuser, Diplomarbeit, Bochum, 1990.
3. D. Braess, *Comput. Mech.* **3**, 321 (1988).
4. D. Braess and W. Hackbusch, *SIAM J. Numer. Anal.* **20**, 967 (1983).
5. M. Bramson and R. Durrett, *Commun. Math. Phys.* **119**, 199 (1988).
6. A. Brandt, *Math. Comput.* **31**, 333 (1977).
7. B. Derrida and J. M. Luck, *Phys. Rev. B* **28**, 7183 (1983).
8. D. S. Fisher, D. Friedan, Z. Qiu, S. J. Shenker, and S. H. Shenker, *Phys. Rev. A* **31**, 3841 (1985).
9. P. Grassberger and R. Leuverink, *J. Phys. A: Math. Gen.* **23**, 773 (1990).
10. W. Hackbusch, *Multi-Grid Methods and Applications* (Springer-Verlag, New York/Berlin, 1985).
11. W. Hackbusch and A. Reusken, Analysis of a damped multilevel method, *Numer. Math.* **55**, 225 (1989).
12. S. Havlin and D. Ben-Avraham, *Adv. Phys.* **36**, 695 (1986).
13. V. E. Kravtsov, I. V. Lerner, and V. I. Yudson, *J. Phys. A, Math. Gen.* **18**, L703 (1985).
14. J. M. Luck, *Nucl. Phys. B* **225**, 169 (1983).
15. J. M. Luck, *J. Phys. A* **17**, 2069 (1984).
16. E. Marinari, G. Parisi, D. Ruelle, and P. Windey, *Commun. Math. Phys.* **89**, 1 (1983).
17. G. Paladin and A. Vulpiani, *Phys. Rep.* **156**, 147 (1987).
18. T. Schneider, A. Politi, and R. Badii, *Phys. Rev. A* **34**, 2505 (1986).
19. Ya. G. Sinai, in *Proceedings, Berlin Conference on Mathematical Problems in Theoretical Physics*, edited by R. Schrader, R. Seiler, and D. A. Uhlenbrock, Lecture Notes in Physics, Vol. 153 (Springer-Verlag, Berlin, 1982), p. 12.
20. D. J. Thouless, *Phys. Rep.* **13**, 93 (1974).
21. P. Wesseling, *An Introduction to Multigrid Methods* (Wiley, New York, 1991).
22. D. M. Young, *Iterative Solution of Large Linear Systems* (Academic Press, New York, 1971).