Journal of Statistical Physics, Vol. 92, Nos. 3/4, 1998

Numerical Simulations of Random Walk in Random Environment

Joseph G. Conlon[1] and Brian von Dohlen[1]

Received January 26, 1998; revised May 4, 1998

This paper is concerned with the numerical simulation of a random walk in a random environment in dimension $d=2$. Consider a nearest neighbor random walk on the 2-dimensional integer lattice. The transition probabilities at each site are assumed to be themselves random variables, but fixed for all time. This is the random environment. Consider a parallel strip of radius R centered on an axis through the origin. Let X_R be the probability that the walk that started at the origin exits the strip through one of the boundary lines. Then X_R is a random variable, depending on the environment. In dimension $d=1$, the variable X_R converges in distribution to the Bernoulli variable, $X_\infty = 0, 1$ with equal probability, as $R \to \infty$. Here the 2-dimensional problem is studied using Gauss–Seidel and multigrid algorithms.

KEY WORDS: Numerical simulations; random walks.

1. INTRODUCTION

In this paper we are concerned with random walks on the one and two dimensional integer lattice. These walks are nearest neighbor walks, but the transition probabilities are random variables themselves. The goal is to understand how these walks behave at large time scales compared to Brownian motion. In dimension $d=1$, it has been shown [S] that the large time behavior is strongly subdiffusive. In constrast, for $d \geqslant 3$ and transition probabilities of small noise it has been shown [BK] that the large time behavior is diffusive with probability 1. This paper is mainly concerned with the $d=2$ case. We report on numerical simulations which indicate that $d=2$ behavior is different from $d=1$ or $d \geqslant 3$ behavior.

[1] Department of Mathematics, University of Michigan, Ann Arbor, Michigan 48109.

0022-4715/98/0800-0571$15.00/0

We consider an elliptic problem which can be stated in any dimension. Let $\mathbb{Z}^d$ be the d dimensional integer lattice. At a lattice point $n \in \mathbb{Z}^d$, there are d directions which we denote by $e_1,..., e_d$. Thus $|e_i| = 1$, $i = 1,..., d$ and the vectors $e_1,..., e_d$ span a d dimensional vector space. With each $n \in \mathbb{Z}^d$ and direction j, $1 \leqslant j \leqslant d$, we associate a Bernoulli random variable $b_{n,j}$. Thus $b_{n,j} = 1$ with probability 1/2, $b_{n,j} = -1$ with probability 1/2, and the $b_{n,j}$ are independent for different values of n and j. Let $\varepsilon > 0$ be a fixed number, $0 < \varepsilon < 1$. We shall be interested in walks $Y(t)$, $t = 0, 1, 2,...$ on $\mathbb{Z}^d$ with transition probabilities given by the formula,

$$P(Y(t+1) = n + e_j \mid Y(t) = n) = \frac{1}{2d}(1 + \varepsilon b_{n,j})$$

$$P(Y(t+1) = n - e_j \mid Y(t) = n) = \frac{1}{2d}(1 - \varepsilon b_{n,j})$$

Next, for integer $M \geqslant 1$ let D_M be the set $D_M = \{n = (n_1,..., n_d): |n_d| \leqslant 2^M\}$. Consider now the walk Y started at the origin and let X_M be the probability it exits D_M through the boundary $\{n: n_d = 2^M\}$. Evidently X_M is a random variable since it depends on the random environment determined by the variables $b_{n,j}$. It is also clear that $0 < X_M < 1$ and $E[X_M] = 1/2$. We are interested in the asymptotic behavior of X_M as $M \to \infty$. Following the method of Sinai [S], we prove in Section 2 a rigorous result for the $d = 1$ case.

Theorem 1. If $d = 1$ and $0 < \varepsilon < 1$, then X_M converges in distribution as $M \to \infty$ to the Bernoulli variable X_∞ given by $X_\infty = 1$ with probability 1/2, $X_\infty = 0$ with probability 1/2.

Observe that since we know $0 < X_M < 1$ and $E[X_M] = 1/2$, the variable X_∞ has maximum possible variance. In view of Theorem 1.1 it is natural to ask if in all dimensions X_M converges in distribution to a variable X_∞ and to enquire about its variance. For $d \geqslant 3$ scaling arguments [BK, F, DL] suggest that $X_\infty = 1/2$ with probability 1. In this paper we report on numerical simulations for the $d = 2$ case. Taking $\varepsilon = 1/2$ and $M = 4, 5, 6$ we find that the distribution of X_M does not decrease significantly over the three scales $M = 4, 5, 6$. In fact we find that up to an error of 0.025 we have $P(X_M < 0.2) = P(X_M > 0.8) = 0$, $P(0.2 < X_M < 0.3) = P(0.7 < X_M < 0.8) = 0.05$, $P(0.3 < X_M < 0.4) = P(0.6 < X_M < 0.7) = 0.15$, $P(0.4 < X_M < 0.5) = P(0.5 < X_M < 0.6) = 0.30$. Hence we expect that for $d = 2$ the variance of X_∞ is nonzero as in the one dimensional case. In contrast to the one dimensional situation, we expect the probability that X_∞

is close to 1 or 0 is small. In Section 4 we give the details of our numerical simulations.

In [F] a renormalisation group argument suggests that the mean square fluctuation of the walk at large time t is linear in t, even in dimension $d=2$. Therefore, at the level of mean square fluctuation, the walk behaves diffusively at large time. Our results are not inconsistent with this claim. They do suggest however, that the recurrence properties of the $d=2$ walk differ from those of the standard walk in $\mathbb{Z}^2$.

To see this, let S_M denote the circles in $\mathbb{R}^2$ centered at the origin and with radius 2^M, $M=0, 1, 2,....$ For $M=1, 2,...,$ let p_M be the probability that the walk started at $(2^M, 0)$ crosses S_{M+1} before crossing S_{M-1}. The probability p_M is, like X_M, a random variable of the environment. Furthermore, p_M and $p_{M'}$ are independent if $|M-M'|\geqslant 2$. We expect that as $M\to\infty$, the variable p_M converges in distribution to a variable p_∞ with $E[p_\infty]=1/2$, corresponding to the zero noise case, and that p_∞ has finite variance.

Consider the walk started at the origin. For $R\geqslant 2$, let N_R be the number of recurrences to the origin before the walk has gone a distance R. In the standard walk case one has $N_R\sim c\ln R$ for some constant c when R is large. We can infer from Sinai's results [S] what N_R should be when there is a random environment. We do this by thinking of the walk as inducing a walk on the circles S_M, $M=0, 1, 2....$ Since the p_M are approximately independent with expectation 1/2, we are roughly in the situation studied by Sinai. According to his method, the number of recurrences to the origin before the walk crosses the circle S_N is of order $\exp[c\sqrt{N}]$ when N is large, for some constant c. Putting $R=2^N$, we conclude that N_R is of order $N_R\sim\exp[c\sqrt{\ln R}]$ when R is large. This is a marked contrast to the case of the standard walk.

One can see the connection to Fisher's result by considering the question of the number of recurrences to a line. Thus consider the walk starting at the origin again and let $\mathcal{N}_R$ be the number of recurrences to the line $\{(n, 0): n\in\mathbb{Z}\}$ before the walk has gone a distance $R\sim 2^N$. In the standard walk one has $\mathcal{N}_R\sim cR$ for large R. We expect that this continues to be the case when there is a random environment. To see this, let $p_M(n)$, $n\in\mathbb{Z}$, $M=1, 2,...$ be the probability that the walk started at the point $(n, 2^M)$ hits the line $\{(r, 2^{M+1}: r\in\mathbb{Z}\}$ before hitting the line $\{(r, 2^{M-1}): r\in\mathbb{Z}\}$. In view of our numerical results it is reasonable to expect that $p_M(n)$ and $p_M(n')$ are approximately independent if $|n-n'|\gg 2^M$. Consider now the walk as inducing a walk on the lines $\{(n, \pm 2^M): n\in\mathbb{Z}\}$, $M=0, 1, 2,...., N$. In the standard walk situation the line $\{(n, 2^M): n\in\mathbb{Z}\}$ is visited over a length scale of 2^N during the order 2^N recurrences to the line $\{(n, 0): n\in\mathbb{Z}\}$. If this continues to be the case for the random environment then the weak

correlation of the probabilities $p_M(n)$ on length scales larger than 2^M can reduce fluctuations sufficiently to still give order 2^N recurrences.

Our method of obtaining the value of X_M for a given environment $b_{n,j}$ is to use a Gauss–Seidel algorithm [C]. This works well for $M=4, 5$ but for $M=6$ is too slow to enable one to make a large number of simulations. We therefore use for $M=6$ a multi grid algorithm. This algorithm works much more efficiently than the Gauss–Seidel algorithm but can diverge if not used appropriately. The divergence problem comes from the fact that the coarse grid operators are no longer elliptic and hence can have spectral radius larger than 1. In Section 3 we describe the Gauss–Seidel and multigrid algorithms we use. We also discuss an interesting 9×9 matrix which is connected to the invariance of coarse grid operators in the multigrid algorithm.

2. THE $d=1$ CASE

We follow the methodology of Sinai [S]. For $n\in\mathbb{Z}$, let $u(n)$ be the probability that the walk Y started at $n\in D_M$ exits D_M through the boundary $n=2^M$. If we set $R=2^M$, then u satisfies the finite difference equation,

$$u(n)=\tfrac{1}{2}(1+\varepsilon b_n)\,u(n+1)+\tfrac{1}{2}(1-\varepsilon b_n)\,u(n-1), \qquad -R<n<R$$

with the boundary conditions $u(-R)=0$, $u(R)=1$. An expression for the solution of this problem can be given in closed form as

$$u(n)=\sum_{k=-R}^{n-1} v(k)\Big/\sum_{k=-R}^{R-1} v(k), \qquad -R<n\leqslant R \tag{2.1}$$

where $v(k)$ is given by $v(0)=1$,

$$v(k)=\prod_{m=1}^{k}\frac{1-\varepsilon b_m}{1+\varepsilon b_m}, \qquad k\geqslant 1$$

$$v(k)=\prod_{m=k+1}^{0}\frac{1+\varepsilon b_m}{1-\varepsilon b_m}, \qquad k\leqslant -1$$

Let us denote the solution (2.1) at $n=0$ by $u_R(0)$ to indicate the dependence on R. Theorem 1.1 follows then from:

Proposition 1. For any δ, $0<\delta<1$, one has

$$\lim_{R\to\infty} P(u_R(0)>1-\delta)=1/2$$

Proof. It is clear that

$$P(u_R(0) > 1-\delta) = P\left(\sum_{k=-R}^{-1} v(k) > \frac{1-\delta}{\delta} \sum_{k=0}^{R-1} v(k)\right)$$

$$\geqslant P\left(\sup_{-R \leqslant k \leqslant -1} v(k) > \frac{(1-\delta)\,R}{\delta} \sup_{0 \leqslant k \leqslant R-1} v(k)\right) \tag{2.2}$$

We shall use the reflection principle to bound below the RHS of (2.2). For $m \in \mathbb{Z}$, let Y_m be the random variables

$$Y_M = \ell n \frac{1-\varepsilon b_m}{1+\varepsilon b_m}, \qquad m \geqslant 1$$

$$= \ell n \frac{1+\varepsilon b_m}{1-\varepsilon b_m}, \qquad m \leqslant 0$$

Thus the Y_m are independent Bernoulli variables, $Y_m = \pm \ell n((1+\varepsilon)/(1-\varepsilon))$ with probability 1/2. For $k \in \mathbb{Z}$ let $S_k = \ell n\, v(k)$. We can write the S_k as sums of the variables Y_m,

$$S_k = \sum_{m=1}^{k} Y_m, \qquad k \geqslant 1$$

$$S_0 = 0$$

$$S_k = \sum_{m=k+1}^{0} Y_m, \qquad k \leqslant -1$$

Evidently we have

$$P\left(\sup_{-R \leqslant k \leqslant -1} v(k) > \frac{(1-\delta)\,R}{\delta} \sup_{0 \leqslant k \leqslant R-1} v(k)\right)$$

$$= P\left(\sup_{-R \leqslant k \leqslant -1} S_k > \ell n \frac{(1-\delta)\,R}{\delta} + \sup_{0 \leqslant k \leqslant R-1} S_k\right)$$

Observe now that the reflection principle [D] yields for any positive integer r,

$$P\left(\sup_{-R \leqslant k \leqslant -1} S_k \geqslant r\,\ell n \frac{1+\varepsilon}{1-\varepsilon}\right) \geqslant 2P\left(S_{-R} \geqslant (r+1)\,\ell n \frac{1+\varepsilon}{1-\varepsilon}\right)$$

Hence,

$$P\left(\sup_{-R\leqslant k\leqslant -1} v(k) > \frac{(1-\delta)\,R}{\delta} \sup_{0\leqslant k\leqslant R-1} v(k)\right)$$

$$\geqslant 2P\left(S_{-R} \geqslant \ell n \frac{(1-\delta)\,R}{\delta} + 2\,\ell n \frac{1+\varepsilon}{1-\varepsilon} + \sup_{0\leqslant k\leqslant R-1} S_k\right)$$

$$= 2\left[1 - P\left(\sup_{0\leqslant k\leqslant R-1} S_k > S_{-R} - \ell n \frac{(1-\delta)\,R}{\delta} - 2\,\ell n \frac{1+\varepsilon}{1-\varepsilon}\right)\right] \tag{2.3}$$

The reflection principle also yields for any nonnegative integer r the inequality,

$$P\left(\sup_{0\leqslant k\leqslant R-1} S_k \geqslant r\,\ell n \frac{1+\varepsilon}{1-\varepsilon}\right) \leqslant 2P\left(S_{R-1} \geqslant r\,\ell n \frac{1+\varepsilon}{1-\varepsilon}\right)$$

We conclude then that

$$P\left(\sup_{0\leqslant k\leqslant R-1} S_k > S_{-R} - \ell n \frac{(1-\delta)\,R}{\delta} - 2\,\ell n \frac{1+\varepsilon}{1-\varepsilon}\right)$$

$$\leqslant P\left(S_{-R} - \ell n \frac{(1-\delta)\,R}{\delta} - 2\,\ell n \frac{1+\varepsilon}{1-\varepsilon} < 0\right)$$

$$+ P\left(S_{-R} - \ell n \frac{(1-\delta)\,R}{\delta} - 2\,\ell n \frac{1+\varepsilon}{1-\varepsilon} \geqslant 0\right)$$

and

$$S_{R-1} \geqslant S_{-R} - \ell n \frac{(1-\delta)\,R}{\delta} - 2\,\ell n \frac{1+\varepsilon}{1-\varepsilon}$$

It is easy to see from the central limit theorem that

$$\lim_{R\to\infty} P\left(S_{-R} - \ell n \frac{(1-\delta)\,R}{\delta} - 2\,\ell n \frac{1+\varepsilon}{1-\varepsilon} < 0\right) = \frac{1}{2}$$

$$\lim_{R\to\infty} P\left(S_{-R} - \ell n \frac{(1-\delta)\,R}{\delta} - 2\,\ell n \frac{1+\varepsilon}{1-\varepsilon} \geqslant 0\right.$$

$$\left.\text{and } S_{R-1} \geqslant S_{-R} - \ell n \frac{(1-\delta)\,R}{\delta} - 2\,\ell n \frac{1+\varepsilon}{1-\varepsilon} < 0\right) = \frac{1}{4}$$

The second identity above follows from the fact that S_{R-1} and S_{-R} are independent and hence converge to independent Gaussians. In view of (2.2), (2.3) it follows that

$$\liminf_{R\to\infty} P(u_R(0) > 1-\delta) \geqslant \tfrac{1}{2}$$

By symmetry we also have that

$$\liminf_{R\to\infty} P(u_R(0) < \delta) \geqslant \tfrac{1}{2}$$

It follows that

$$\begin{aligned}\limsup_{R\to\infty} P(u_R(0) > 1-\delta) &\leqslant \limsup_{R\to\infty} [1 - P(u_R(0) < \delta)] \\ &= 1 - \liminf_{R\to\infty} P(u_R(0) < \delta) \leqslant \tfrac{1}{2}\end{aligned}$$

The result follows.

3. NUMERICAL ALGORITHMS FOR $d=2$

For integers $M, L > 0$ we consider the region $D_{M,L} \subset \mathbb{Z}^2$ defined by $D_{M,L} = \{(n_1, n_2) \in \mathbb{Z}^2 : |n_1| \leqslant L2^M, |n_2| \leqslant 2^M\}$. For functions $v: D_{M,L} \to \mathbb{R}$ we define a finite difference operator $\mathscr{L}$ by

$$\begin{aligned}\mathscr{L}v(i,j) = {} & a(i,j)\, v(i,j) + b(i,j)\, v(i,j+1) + c(i,j)\, v(i,j-1) \\ & + d(i,j)\, v(i+1,j) + e(i,j)\, v(i-1,j) + f(i,j)\, v(i+1,j+1) \\ & + g(i,j)\, v(i-1,j+1) + h(i,j)\, v(i+1,j-1) \\ & + k(i,j)\, v(i-1,j-1) \qquad (3.1)\end{aligned}$$

Hence the functions $a, b, c, d, e, f, g, h, k$ are specified. In our case they are given by the formulas

$$\begin{gathered}a(i,j) = 4, \qquad b(i,j) = -1 - 2\varepsilon\rho(i,j) + \varepsilon \\ c(i,j) = -1 + 2\varepsilon\rho(i,j) - \varepsilon, \qquad d(i,j) = -1 - 2\varepsilon\sigma(i,j) + \varepsilon \\ e(i,j) = -1 + 2\varepsilon\sigma(i,j) - \varepsilon, \qquad f \equiv g \equiv h \equiv k \equiv 0\end{gathered}$$

Here $\rho(i,j)$ and $\sigma(i,j)$ are randomly generated bits which are 0 or 1 with probability 1/2.

We shall be interested in solving a boundary value problem on $D_{M,L}$ given by

$$\begin{aligned} \mathscr{L}v(i,j) &= 0, \qquad |i| < L2^M, \qquad |j| < 2^M \\ v(i,-2^M) &= 0, \qquad |i| \leqslant L2^M \\ v(i,2^M) &= 1, \qquad |i| \leqslant L2^M \\ v(-L2^M, j) &= (j+2^M)/2^{M+1}, \qquad |j| \leqslant 2^M \\ v(L2^m, j) &= (j+2^M)/2^{M+1}, \qquad |j| \leqslant 2^M \end{aligned} \tag{3.2}$$

Thus the boundary conditions are given by v is 0 or 1 on the horizontal boundaries and linear on the vertical boundaries. Having solved the problem (3.2) numerically we shall take $v(0,0)$ as our approximate value of the exit probability X_M. Evidently L needs to be sufficiently large to approximate an infinite strip. Empirically we find that $L=3$ is sufficient for good accuracy.

For $M=4,5$ we approximately solve (3.2) by using a Gauss–Seidel algorithm. Thus for one iteration of the algorithm we run the loop

$$\begin{aligned} v(i,j) = -a(i,j)^{-1}\,[&b(i,j)\,v(i,j+1) + c(i,j)\,v(i,j-1) + d(i,j)\,v(i+1,j) \\ &+ e(i,j)\,v(i-1,j) + f(i,j)\,v(i+1,j+1) + g(i,j)\,v(i-1,j+1) \\ &+ h(i,j)\,v(i+1,j-1) \\ &+ k(i,j)\,v(i-1,j-1)], \qquad |i| < L2^M, \quad |j| < 2^M \end{aligned} \tag{3.3}$$

When $M=6$, this algorithm is too slow and we use a multigrid algorithm. To describe this consider the coarse grid region $D_{M,L,\text{coarse}}$ corresponding to $D_{M,L}$, defined by

$$D_{M,L,\text{coarse}} = \{(i,j) \in D_{M,L} : i, j \text{ even}\}$$

Observe that if $M \geqslant 1$, then $D_{M,L,\text{coarse}}$ includes boundary points of $D_{M,L}$. For a function v with domain $D_{M,L,\text{coarse}}$ we can define a new function Qv with domain $D_{M,L}$ by linear interpolation. Thus

$$\begin{aligned} Qv(i,j) &= v(i,j) && \text{if both } i, j \text{ even} \\ Qv(i,j) &= \tfrac{1}{2}[v(i+1,j) + v(i-1,j)] && \text{if } i \text{ is odd, } j \text{ even} \\ Qv(i,j) &= \tfrac{1}{2}[v(i,j+1) + v(i,j-1)] && \text{if } i \text{ is even, } j \text{ odd} \\ Qv(i,j) &= \tfrac{1}{4}[v(i+1,j+1) + v(i+1,j-1) + v(i-1,j+1) \\ &\quad + v(i-1,j-1)] && \text{if both } i, j, \text{ odd} \end{aligned} \tag{3.4}$$

If v is a function with domain $D_{M,L}$ then we can define a function Q^*v with domain $D_{M,L,\text{coarse}}$ by full weighting [B]. Thus

$$\begin{aligned} Q^*v(i,j) = &\tfrac{1}{4}v(i,j) + \tfrac{1}{8}v(i+1,j) + \tfrac{1}{8}v(i-1,j) + \tfrac{1}{8}v(i,j+1) \\ &+ \tfrac{1}{8}v(i,j-1) + \tfrac{1}{16}v(i+1,j+1) + \tfrac{1}{16}v(i-1,j+1) \\ &+ \tfrac{1}{16}v(i+1,j-1) + \tfrac{1}{16}v(i-1,j-1) \end{aligned} \tag{3.5}$$

Since Q takes functions with domain $D_{M,L,\text{coarse}}$ to functions with domain $D_{M,L}$ and Q^* takes functions with domain $D_{M,L}$ to functions with domain $D_{M,L,\text{coarse}}$ it follows that the operator $\mathscr{L}_{\text{coarse}} = 4Q^*\mathscr{L}Q$ takes functions with domain $D_{M,L,\text{coarse}}$ to themselves. The operator $\mathscr{L}_{\text{coarse}}$ is the operator $\mathscr{L}$ transferred to the coarse grid in the language of multigrid.

Since the coefficients in the operator $\mathscr{L}$ are highly oscillatory we need to carefully calculate the coefficients of $\mathscr{L}_{\text{coarse}}$. It turns out that $\mathscr{L}_{\text{coarse}}$ has the same form as $\mathscr{L}$ in (3.1), whence we can write

$$\begin{aligned} \mathscr{L}_{\text{coarse}}v(i,j) = &\, a_{\text{coarse}}(i,j)\, v(i,j) + b_{\text{coarse}}(i,j)\, v(i,j+2) \\ &+ c_{\text{coarse}}(i,j)\, v(i,j-2) + d_{\text{coarse}}(i,j)\, v(i+2,j) \\ &+ e_{\text{coarse}}(i,j)\, v(i-2,j) + f_{\text{coarse}}(i,j)\, v(i+2,j+2) \\ &+ g_{\text{coarse}}(i,j)\, v(i-2,j+2) + h_{\text{coarse}}(i,j)\, v(i+2,j-2) \\ &+ k_{\text{coarse}}(i,j)\, v(i-2,j-2), \qquad i,j \text{ even} \end{aligned}$$

The coefficients a_{coarse}, b_{coarse}, c_{coarse}, d_{coarse}, e_{coarse}, f_{coarse}, g_{coarse}, h_{coarse}, k_{coarse} are given by the following formulas,

$$\begin{aligned} a_{\text{coarse}}(i,j) = &\, a(i,j) + \tfrac{1}{2}b(i,j) + \tfrac{1}{2}c(i,j) \\ &+ \tfrac{1}{2}d(i,j) + \tfrac{1}{2}e(i,j) + \tfrac{1}{4}f(i,j) + \tfrac{1}{4}g(i,j) + \tfrac{1}{4}h(i,j) \\ &+ \tfrac{1}{4}k(i,j) + \tfrac{1}{4}a(i+1,j) + \tfrac{1}{8}b(i+1,j) + \tfrac{1}{8}c(i+1,j) \\ &+ \tfrac{1}{2}e(i+1,j) + \tfrac{1}{4}g(i+1,j) + \tfrac{1}{4}k(i+1,j) + \tfrac{1}{4}a(i-1,j) \\ &+ \tfrac{1}{8}b(i-1,j) + \tfrac{1}{8}c(i-1,j) + \tfrac{1}{2}d(i-1,j) + \tfrac{1}{4}f(i-1,j) \\ &+ \tfrac{1}{4}h(i-1,j) + \tfrac{1}{4}a(i,j+1) + \tfrac{1}{8}d(i,j+1) + \tfrac{1}{8}e(i,j+1) \\ &+ \tfrac{1}{2}c(i,j+1) + \tfrac{1}{4}k(i,j+1) + \tfrac{1}{4}h(i,j+1) + \tfrac{1}{4}a(i,j-1) \\ &+ \tfrac{1}{8}d(i,j-1) + \tfrac{1}{8}e(i,j-1) + \tfrac{1}{2}b(i,j-1) + \tfrac{1}{4}g(i,j-1) \\ &+ \tfrac{1}{4}f(i,j-1) + \tfrac{1}{16}a(i+1,j+1) + \tfrac{1}{8}c(i+1,j+1) \end{aligned}$$

$$+\tfrac{1}{8}e(i+1,j+1)+\tfrac{1}{4}k(i+1,j+1)+\tfrac{1}{16}a(i-1,j-1)$$
$$+\tfrac{1}{8}b(i-1,j-1)+\tfrac{1}{8}d(i-1,j-1)+\tfrac{1}{4}f(i-1,j-1)$$
$$+\tfrac{1}{16}a(i-1,j+1)+\tfrac{1}{8}c(i-1,j+1)+\tfrac{1}{8}d(i-1,j+1)$$
$$+\tfrac{1}{4}h(i-1,j+1)+\tfrac{1}{16}a(i+1,j-1)+\tfrac{1}{8}b(i+1,j-1)$$
$$+\tfrac{1}{8}e(i+1,j-1)+\tfrac{1}{4}g(i+1,j-1)$$

$$b_{\text{coarse}}(i,j)=\tfrac{1}{2}b(i,j)+\tfrac{1}{4}f(i,j)+\tfrac{1}{4}g(i,j)$$
$$+\tfrac{1}{8}b(i+1,j)+\tfrac{1}{4}g(i+1,j)+\tfrac{1}{8}b(i-1,j)+\tfrac{1}{4}f(i-1,j)$$
$$+\tfrac{1}{4}a(i,j+1)+\tfrac{1}{2}b(i,j+1)+\tfrac{1}{8}e(i,j+1)+\tfrac{1}{8}d(i,j+1)$$
$$+\tfrac{1}{4}f(i,j+1)+\tfrac{1}{4}g(i,j+1)+\tfrac{1}{16}a(i+1,j+1)$$
$$+\tfrac{1}{8}b(i+1,j+1)+\tfrac{1}{4}g(i+1,j+1)+\tfrac{1}{8}e(i+1,j+1)$$
$$+\tfrac{1}{16}a(i-1,j+1)+\tfrac{1}{8}b(i-1,j+1)+\tfrac{1}{8}d(i-1,j+1)$$
$$+\tfrac{1}{4}f(i-1,j+1)$$

$$c_{\text{coarse}}(i,j)=\tfrac{1}{2}c(i,j)+\tfrac{1}{4}h(i,j)+\tfrac{1}{4}k(i,j)$$
$$+\tfrac{1}{8}c(i+1,j)+\tfrac{1}{4}k(i+1,j)+\tfrac{1}{8}c(i-1,j)+\tfrac{1}{4}h(i-1,j)$$
$$+\tfrac{1}{4}a(i,j-1)+\tfrac{1}{2}c(i,j-1)+\tfrac{1}{8}e(i,j-1)+\tfrac{1}{8}d(i,j-1)$$
$$+\tfrac{1}{4}h(i,j-1)+\tfrac{1}{4}k(i,j-1)+\tfrac{1}{16}a(i+1,j-1)$$
$$+\tfrac{1}{8}c(i+1,j-1)+\tfrac{1}{8}e(i+1,j-1)+\tfrac{1}{4}k(i+1,j-1)$$
$$+\tfrac{1}{16}a(i-1,j-1)+\tfrac{1}{8}c(i-1,j-1)+\tfrac{1}{8}d(i-1,j-1)$$
$$+\tfrac{1}{4}h(i-1,j-1)$$

$$d_{\text{coarse}}(i,j)=\tfrac{1}{2}d(i,j)+\tfrac{1}{4}f(i,j)+\tfrac{1}{4}h(i,j)+\tfrac{1}{8}d(i,j-1)$$
$$+\tfrac{1}{4}f(i,j-1)+\tfrac{1}{8}d(i,j+1)+\tfrac{1}{4}h(i,j+1)+\tfrac{1}{4}a(i+1,j)$$
$$+\tfrac{1}{2}d(i+1,j)+\tfrac{1}{8}b(i+1,j)+\tfrac{1}{8}c(i+1,j)+\tfrac{1}{4}h(i+1,j)$$
$$+\tfrac{1}{4}f(i+1,j)+\tfrac{1}{16}a(i+1,j-1)+\tfrac{1}{8}d(i+1,j-1)$$
$$+\tfrac{1}{4}f(i+1,j-1)+\tfrac{1}{8}b(i+1,j-1)+\tfrac{1}{16}a(i+1,j+1)$$
$$+\tfrac{1}{8}d(i+1,j+1)+\tfrac{1}{8}c(i+1,j+1)$$
$$+\tfrac{1}{4}h(i+1,j+1)$$

$$
\begin{aligned}
e_{\text{coarse}}(i,j) &= \tfrac{1}{2}e(i,j) + \tfrac{1}{4}g(i,j) + \tfrac{1}{4}k(i,j) \\
&\quad + \tfrac{1}{8}e(i,j+1) + \tfrac{1}{4}k(i,j+1) + \tfrac{1}{8}e(i,j-1) + \tfrac{1}{4}g(i,j-1) \\
&\quad + \tfrac{1}{4}a(i-1,j) + \tfrac{1}{2}e(i-1,j) + \tfrac{1}{8}c(i-1,j) + \tfrac{1}{8}b(i-1,j) \\
&\quad + \tfrac{1}{4}g(i-1,j) + \tfrac{1}{4}k(i-1,j) + \tfrac{1}{16}a(i-1,j+1) \\
&\quad + \tfrac{1}{8}e(i-1,j+1) + \tfrac{1}{4}k(i-1,j+1) + \tfrac{1}{8}c(i-1,j+1) \\
&\quad + \tfrac{1}{16}a(i-1,j-1) + \tfrac{1}{8}e(i-1,j-1) + \tfrac{1}{8}b(i-1,j-1) \\
&\quad + \tfrac{1}{4}g(i-1,j-1) \\
f_{\text{coarse}}(i,j) &= \tfrac{1}{4}f(i,j) + \tfrac{1}{8}b(i+1,j) + \tfrac{1}{4}f(i+1,j) \\
&\quad + \tfrac{1}{8}d(i,j+1) + \tfrac{1}{4}f(i,j+1) + \tfrac{1}{8}b(i+1,j+1) \\
&\quad + \tfrac{1}{8}d(i+1,j+1) + \tfrac{1}{16}a(i+1,j+1) + \tfrac{1}{4}f(i+1,j+1) \\
g_{\text{coarse}}(i,j) &= \tfrac{1}{4}g(i,j) + \tfrac{1}{8}b(i-1,j) + \tfrac{1}{4}g(i-1,j) \\
&\quad + \tfrac{1}{8}e(i,j+1) + \tfrac{1}{4}g(i,j+1) + \tfrac{1}{8}b(i-1,j+1) \\
&\quad + \tfrac{1}{8}e(i-1,j+1) + \tfrac{1}{16}a(i-1,j+1) + \tfrac{1}{4}g(i-1,j+1) \\
h_{\text{coarse}}(i,j) &= \tfrac{1}{4}h(i,j) + \tfrac{1}{8}c(i+1,j) + \tfrac{1}{4}h(i-1,j) \\
&\quad + \tfrac{1}{8}d(i,j-1) + \tfrac{1}{4}h(i,j-1) + \tfrac{1}{8}c(i+1,j-1) \\
&\quad + \tfrac{1}{8}d(i+1,j-1) + \tfrac{1}{16}a(i+1,j-1) + \tfrac{1}{4}h(i+1,j-1) \\
k_{\text{coarse}}(i,j) &= \tfrac{1}{4}k(i,j) + \tfrac{1}{8}c(i-1,j) + \tfrac{1}{4}k(i-1,j) \\
&\quad + \tfrac{1}{8}e(i,j-1) + \tfrac{1}{4}k(i,j-1) + \tfrac{1}{8}c(i-1,j-1) \\
&\quad + \tfrac{1}{8}e(i-1,j-1) + \tfrac{1}{16}a(i-1,j-1) + \tfrac{1}{4}k(i-1,j-1)
\end{aligned}
$$

It is interesting to consider the relationship of $\mathscr{L}_{\text{coarse}}$ to $\mathscr{L}$ when $\mathscr{L}$ has constant coefficients $a, b, c, d, e, f, g, h, k$. Then $\mathscr{L}_{\text{coarse}}$ also has constant coefficients and they are related to the coefficients of $\mathscr{L}$ by the equation,

$$
\begin{aligned}
&[a_{\text{coarse}}, b_{\text{coarse}}, c_{\text{coarse}}, d_{\text{coarse}}, e_{\text{coarse}}, f_{\text{coarse}}, g_{\text{coarse}}, h_{\text{coarse}}, k_{\text{coarse}}]^T \\
&\quad = A[a, b, c, d, e, f, g, h, k]^T
\end{aligned}
$$

where the 9×9 matrix A is given by

$$A = \begin{bmatrix} 9/4 & 3/2 & 3/2 & 3/2 & 3/2 & 1 & 1 & 1 & 1 \\ 3/8 & 3/2 & 0 & 1/4 & 1/4 & 1 & 1 & 0 & 0 \\ 3/8 & 0 & 3/2 & 1/4 & 1/4 & 0 & 0 & 1 & 1 \\ 3/8 & 1/4 & 1/4 & 3/2 & 0 & 1 & 0 & 1 & 0 \\ 3/8 & 1/4 & 1/4 & 0 & 3/2 & 0 & 1 & 0 & 1 \\ 1/16 & 1/4 & 0 & 1/4 & 0 & 1 & 0 & 0 & 0 \\ 1/16 & 1/4 & 0 & 0 & 1/4 & 0 & 1 & 0 & 0 \\ 1/16 & 0 & 1/4 & 1/4 & 0 & 0 & 0 & 1 & 0 \\ 1/16 & 0 & 1/4 & 0 & 1/4 & 0 & 0 & 0 & 1 \end{bmatrix}$$

It is easy to see that the vector $w = [8, -1, -1, -1, -1, -1, -1, -1, -1]^T$ is invariant under A, i.e., $Aw = w$. Thus the nonstandard discretized Laplacian with coefficients $a = 8$, $b = c = d = e = f = g = h = k = -1$, remains invariant under the coarse grid transformation. In contrast the standard discretized Laplacian with coefficients $a = 4$, $b = c = d = e = -1$, $f = g = h = k = 0$ is not invariant under coarse grid transformation. We can see numerically however that the iterates $A^n v$, $n = 1, 2, \ldots$ with $v = [4, -1, -1, -1, -1, 0, 0, 0, 0]^T$ become rapidly parallel to w as n increases.

Numerically, all the eigenvalues of A are powers of 2. The eigenvalues 4 and 1/4 are simple, the eigenvalues 2 and 1/2 both have multiplicity 2 and the eigenvalue 1 has multiplicity 3. We have already exhibited an eigenvector with eigenvalue 1. We can also find exactly the eigenvectors with eigenvalues 4 and 1/4. We have that $[16, 4, 4, 4, 4, 1, 1, 1, 1]$ is the eigenvector with eigenvalue 4 and $[-4, 2, 2, 2, 2, -1, -1, -1, -1]$ is the eigenvector with eigenvalue 1/4. We can find an explanation for the existence of the eigenvalue 4 from the structure of the transformation $\mathscr{L} \to \mathscr{L}_{\text{coarse}}$. Observe first that the operator Q takes the constant function to itself. Hence if $\mathscr{L}$ takes the constant function to zero then $\mathscr{L}_{\text{coarse}}$ also takes the constant function to zero. This is the same as saying that if the vector $v = [a, b, c, d, e, f, g, h, k]^T$ is orthogonal to the constant vector then Av is also. This implies that the constant vector is an eigenvector of the transpose A^T of A. Its eigenvalue turns out to be 4.

We define the multigrid algorithm in the standard way. Thus the fine grid operator is denoted by $\mathscr{L}_M$ (with $M = 6$ in our case). We can then construct from our formulas for a_{coarse}, b_{coarse}, c_{coarse}, d_{coarse}, e_{coarse}, f_{coarse}, g_{coarse}, h_{coarse}, k_{coarse} inductively the coarse grid operators $\mathscr{L}_N$ with N an integer, $0 \leqslant N < M$. Let us suppose we have an initial guess for the solution of the boundary value problem (3.2). If we denote this by v then we iterate

once on the finest grid, corresponding to M, using the Gauss–Seidel algorithm (3.3). Next we compute the residual r_M on the fine grid defined by

$$r_M(i, j) = \mathscr{L} v_M(i, j) = \mathscr{L}_M v_M(i, j), \qquad |i| < L2^M, \quad |j| < 2^M \tag{3.6}$$

Observe that the vector v_M in (3.6) is the vector which replaces the initial guess after the Gauss–Seidel iteration. Next we transfer r_M to the coarse grid corresponding to $M-1$ by using the transfer operator Q^* of (3.5). Next we do one iteration of the Gauss–Seidel algorithm on the equation

$$\mathscr{L}_{M-1} v_{M-1} = 4Q^* r_M$$

where we take v_{M-1} to have zero boundary conditions on $D_{M, L, \text{coarse}}$, and the initial guess for v_{M-1} to be zero. We transfer the residual of this down to the next coarsest grid corresponding to $M-2$ and proceed in an exactly similar way until we get to the lowest level we have chosen which is q for some integer q, $0 \leqslant q < M$. We have now constructed vectors v_N, $N = M \cdots q$. To go up the V cycle of the multigrid algorithm we transfer v_q to the grid corresponding to $q+1$ by means of the operator Q of (3.4). Now, using $v_{q+1} + Qv_q$ as our initial guess, we iterate once using Gauss–Seidel on the $q+1$ grid. We continue this procedure until we reach the finest grid again and this gives us the new value of v_M. We shall refer to this algorithm in the next section as $mgd(M, q)$.

4. NUMERICAL RESULTS

The programs for these numerical simulations were written in *C*. The random number generator random ( ) was used to construct the random environment determined by the variables $b_{n,j}$. From each random number generated by random ( ), the 10th bit was taken and a $b_{n,j}$ constructed from this by subtracting 1. In all the simulations we take $L = 3$ and $\varepsilon = 1/2$. It was discovered that on taking $L = 3$ one obtains the value X_M correct to 3 decimal places. Increasing the value of L to $L = 4$ for example improved accuracy only beyond the third decimal place. Taking $\varepsilon = 1/2$ insures that the effect of the noisy environment is substantial, but still avoids singularities which would occur if we take ε close to 1.

For $M = 4, 5$ results were obtained by Gauss–Seidel iteration. The approximations to the value of X_M were compared every 20th iteration. If these values differed by less than 1/500 then the program terminated. Hence the goal was to do sufficient iterations to obtain X_M correct to 2 decimal places. In the case of $M = 5$ this required up to 1000 iterations.

Table I gives the probability density for X_4 based on 1,000 simulations of the random environment.

Table I. Probability Density for X_4

X_4	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
0.0	0	0	0	0	0	0	0	0	0	0
0.1	0	0	0	0.002	0	0.001	0	0	0.001	0.002
0.2	0.001	0.003	0.002	0.005	0.002	0.006	0.006	0.006	0.004	0.009
0.3	0.016	0.011	0.011	0.012	0.015	0.017	0.018	0.022	0.018	0.026
0.4	0.021	0.023	0.022	0.023	0.025	0.036	0.026	0.026	0.042	0.032
0.5	0.033	0.020	0.032	0.031	0.035	0.025	0.018	0.026	0.038	0.026
0.6	0.026	0.026	0.015	0.020	0.012	0.019	0.018	0.01	0.011	0.011
0.7	0.008	0.011	0.008	0.001	0.009	0.002	0.004	0.002	0.003	0.002
0.8	0.001	0.002	0.001	0	0	0.001	0	0.001	0	0
0.9	0	0	0	0	0	0	0	0	0	0

Thus our computed value for $P(X_4 = 0.34)$ is, from the above table, 0.015. In a separate run of 10,000 simulations roughly the same table of values for the probability density was obtained. Observe that X_4 stays away from both 0 and 1. This is in marked contrast with the $d = 1$ case where we have shown that X_M concentrates at 0 and 1 for large M. Note also that there is a significant spread of X_4 around the value 0.5. This contrasts with the conjectured behavior of X_M in $d \geqslant 3$ dimensions when M is large. There one expects X_M to concentrate at 0.5 for large M.

The qualitative features of the X_4 table persist for X_5 and X_6. The X_5 table, based on 1000 simulations, is given in Table II.

The probability density table for X_6 was obtained by using the multigrid method. It is clear from the formulas for the coarse grid operators given in Section 3 that coarse grid operators will in general not be elliptic everywhere. Hence the spectral radius can be larger than 1. We tested the

Table II. Probability Density for X_5

X_5	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
0.0	0	0	0	0	0	0	0	0	0	0
0.1	0	0	0	0	0	0	0	0.001	0.001	0.001
0.2	0	0.003	0.001	0	0.006	0.001	0.007	0.009	0.005	0.005
0.3	0.012	0.006	0.011	0.019	0.016	0.018	0.018	0.009	0.025	0.025
0.4	0.025	0.027	0.026	0.021	0.037	0.021	0.028	0.031	0.025	0.053
0.5	0.052	0.042	0.031	0.034	0.023	0.029	0.026	0.034	0.024	0.021
0.6	0.017	0.023	0.020	0.018	0.013	0.008	0.012	0.007	0.012	0.006
0.7	0.015	0.006	0.008	0.007	0.001	0.007	0.002	0.004	0.001	0.001
0.8	0.002	0.001	0	0	0	0	0	0	0	0
0.9	0	0	0	0	0	0	0	0	0	0

convergence properties of the coarse grid operators. This was done by iterating on the constant function using the Gauss–Seidel algorithm with a coarse grid operator. One assumes zero boundary conditions on $D_{M, L, \text{coarse}}$ and $M=6$, $L=3$. The value of the vector at the origin is recorded every 10th iteration. Suppose this value is y and the previous recorded value is y_{prev}. The coarse grid operator is determined to give a converging Gauss–Seidel algorithm if $|y - y_{\text{prev}}| < |y_{\text{prev}}|/100$ at some point before 1000 iterations. If this fails to happen we conclude the coarse-grid operator gives a diverging Gauss–Seidel algorithm. The coarse grid operators for $M=6$ can be labeled $\mathscr{L}_m$, $m=5, 4, 3, 2, 1$, with higher values of m corresponding to finer grids. It was discovered that Gauss–Seidel for $\mathscr{L}_5$ converges with probability 0.97, for $\mathscr{L}_4$ with probability 0.87, for $\mathscr{L}_3$ with probability 0.84 and for $\mathscr{L}_2$ with probability 0.92. This was based on 1000 simulations.

In view of the probability of divergence of coarse grid iterations, the multi-grid algorithm needed to be implemented with care. To do this we first computed 20 V cycles of the algorithm $mgd(6, 3)$. This was declared to converge if the standard deviation of the value of X_6 given by the last 4 iterations was less than $0.01/\sqrt{10}$. If the standard deviation was larger than this we computed 40 V cycles of $mgd(6, 4)$. Again this was declared to converge if the standard deviation of the last 4 iterations was less than $0.01/\sqrt{10}$. In 1000 simulations there was just one seed which did not converge according to one of the above criteria. This was found to converge by computing a large number of iterations of $mgd(6, 5)$.

Surprisingly it was discovered that the algorithms $mdg(1)$ and $mgd(2)$ converged for very few seeds. The algorithm $mgd(3)$ converged for about 80% of seeds and is still fast, requiring just 20 V cycles for convergence. In contrast pure Gauss–Seidel iteration on the fine lattice requires 5000 iterations for 2 decimal place accuracy.

Table III. Probability Density for X_6

X_6	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
0.0	0	0	0	0	0	0	0	0	0	0
0.1	0	0	0	0	0.001	0	0	0	0	0.001
0.2	0.001	0.002	0.004	0.002	0.002	0.002	0.004	0.004	0.007	0.009
0.3	0.007	0.013	0.014	0.012	0.019	0.014	0.022	0.021	0.021	0.024
0.4	0.020	0.022	0.032	0.032	0.030	0.039	0.033	0.039	0.039	0.032
0.5	0.029	0.036	0.029	0.031	0.034	0.024	0.036	0.027	0.027	0.030
0.6	0.021	0.025	0.018	0.011	0.012	0.012	0.012	0.005	0.015	0.005
0.7	0.006	0.004	0.007	0.003	0.004	0.004	0.002	0.001	0.002	0.002
0.8	0	0.001	0	0	0	0	0	0	0	0
0.9	0	0	0	0	0	0	0	0	0	0

Using the above implementation of the multi-grid algorithm we obtained Table III for the probability density for X_6. The table was constructed from 1000 simulations.

ACKNOWLEDGMENTS

This research was conducted while the authors were visiting the Center for Industrial and Applied Mathematics of Oxford University. We would like to thank the director Hilary Ockendon and her colleagues for their hospitality. The work of J.G.C. was supported by NSF Grant DMS-9404197.

REFERENCES

[S] Y. Sinai, The limiting behavior of a one dimensional random walk in random medium, *Th. Prob. Appl.* **27**:256–268 (1982).

[BK] J. Bricmont and A. Kupiainen, Random walk in asymmetric random environment, *Commun. in Math. Phys.* **142**:345–420 (1991).

[C] P. Ciarlet, *Introduction to Numerical Linear Algebra and Optimization* (Cambridge University Press, 1991).

[D] R. Durrett, *Probability Theory and Examples* (Wadsworth, 1991).

[B] W. Briggs, *A Multigrid Tutorial* (SIAM, 1994).

[F] D. Fisher, Random walks in random environment, *Physical Review A* **30**:960–964 (1984).

[DL] B. Derrida and J. M. Luck, Diffusion on a random lattice: Weak-disorder expansion in arbitrary dimension, *Phys. Rev. B* **28**:7183–7190 (1983).