Available online at www.sciencedirect.com

SCIENCE DIRECT®

Electronic Notes in Discrete Mathematics 25 (2006) 61–64

Electronic Notes in DISCRETE MATHEMATICS

www.elsevier.com/locate/endm

Quasirandomness in Graphs

Benjamin Doerr

Max-Planck-Institut für Informatik
Stuhlsatzenhausweg 85
66123 Saarbrücken

Tobias Friedrich

Max-Planck-Institut für Informatik
Stuhlsatzenhausweg 85
66123 Saarbrücken

Abstract

Jim Propp's rotor router model is a simple deterministic analogue of a random walk. Instead of distributing chips randomly, it serves the neighbors in a fixed order. We analyze the difference between Propp machine and random walk on the infinite two-dimensional grid. We show that, independent of the starting configuration, at each time, the number of chips on each vertex deviates from the expected number of chips in the random walk model by at most a constant c, which is 7.83 for clockwise rotor sequences and 7.28 otherwise. This is the first paper which demonstrates that the order in which the neighbors are served makes a difference.

Keywords: Random walk, Quasirandomness

1 Introduction

The rotor-router model is a simple deterministic process which has been suggested by Jim Propp as an attempt to derandomize random walks on infinite grids $\mathbb{Z}^d$. There, each vertex $x \in \mathbb{Z}^d$ is associated with a "rotor" and a cyclic permutation of the $2d$ cardinal directions of $\mathbb{Z}^d$. While in a random walk a chip leaves a vertex in a random direction, chips of the Propp machine always go into the direction the current rotor is pointing. After a chip is sent, the rotor is rotated according to the fixed cyclic permutation. This ensures the chips are distributed highly evenly among the neighbors.

1571-0653/$ – see front matter © 2006 Published by Elsevier B.V.
doi:10.1016/j.endm.2006.06.066

This process has attracted considerable attention recently. It has been shown that the Propp machine resembles very closely a random walk in several respects. Cooper and Spencer [1] examined the *single vertex discrepancy.* That is, we start with an arbitrary initial configuration (i.e., number of chips on vertices and rotor directions), run both machines for some time and compare the number of chips of the Propp machine with the expected number of chips of the random walk. Apart from a technicality, which we defer to Section 2, the answer is astonishing: For grids $\mathbb{Z}^d$ the discrepancy can be bounded by a constant c_d, which only depends on the dimension. In particular, c_d is independent of the initial configuration, the runtime, and the cyclic permutation of the cardinal directions. For the graph being the infinite path, Cooper et al. [2] showed that this constant is $c_1 \approx 2.29$.

In this paper, we consider the two-dimensional grid $\mathbb{Z}^2$. This is the highest dimension which can be analyzed rigorously. In comparison to the one-dimensional grid new effects appear, in particular the rotor sequence comes into play. In the one-dimensional case, the arrow must just switch back and forth after each chip sent to minimize the discrepancy between the number of chips sent left and right. This optimally equals out chips sent to the left and to the right. In higher dimension this cannot be achieved for all directions at the same time. This trade-off between balancing out all directions, gives rise to analyze different rotor sequences. We show tight upper and lower bounds for the single vertex discrepancy of $\mathbb{Z}^2$. Strictly speaking, we show $c_2 \approx 7.83$ for clockwise rotor sequences and $c_2 \approx 7.28$ for all other rotor sequences.

2 Preliminaries

On the standard two-dimensional grid $\mathbb{Z}^2$, where each vertex is connected with its up/down/left/right-neighbor, both dimensions do not vary independently in a random walk. Hence, to simplify the calculations, we rotate the grid by 45° and consider instead neighbors in directions $\textsc{dir} := \{ \nearrow, \searrow, \swarrow, \nwarrow \}$. Note that by this, we only allow chips on $\mathbf{x}$ with $x_1 \equiv x_2 (\text{mod } 2)$. Since both models are isomorphic, our results can immediately be translated into the standard two-dimensional grid model with neighbors $\{\uparrow, \rightarrow, \downarrow, \leftarrow\}$.

As already pointed out in the introduction, there is one limitation, without neither the results of [1,2] nor our results hold. For this, note that since $\mathbb{Z}^2$ is bipartite, chips that start on *even* positions never mix with those starting on *odd* positions. It looks like we are playing two games at once. However, this is not true, because chips of different parity may affect each other through the rotors. The number of chips send in each direction at each position is then not balanced within both games. One can cleverly arrange piles of off-parity chips to reorient rotors and steer them away from random walk simulation. We therefore require the starting configuration to have chips only on *one* parity. Without loss of generality, we consider only even starting configurations. Odd starting configurations can be handled in an analogous manner.

3 The Basic Method

A random walk can be described nicely by its probability density. By $H(\mathbf{x},t)$ we denote the probability that a chip from the origin arrives at location $\mathbf{x}$ at time t in a simple random walk. Therefore, $H(\mathbf{x},t) = 4^{-t}\binom{t}{(t+x_1)/2}\binom{t}{(t+x_2)/2}$ for $x_1 \equiv x_2 \equiv t \pmod 2$ and $\|\mathbf{x}\|_\infty \leq t$, and $H(\mathbf{x},t) = 0$ otherwise.

For a fixed starting configuration, $f(\mathbf{x},t)$ denotes the number of chips and $\textsc{arr}(\mathbf{x},t)$ denotes the direction of the arrow at position $\mathbf{x}$ after t steps of the Propp machine. $E(\mathbf{x},t)$ denotes the expected number of chips at location $\mathbf{x}$ after a random walk of t steps.

Let $s_i(\mathbf{y})$ denote the time that $\mathbf{y}$ is occupied by its i-th chip, i.e., $s_i(\mathbf{y}) := \min\{u \geq 0 \mid i < \sum_{t=0}^{u} f(\mathbf{y},t)\}$ for all $i \in \mathbb{N}_0$. With $\textsc{inf}(\mathbf{y},\mathbf{A},t) := H(\mathbf{y}+\mathbf{A},t-1) - H(\mathbf{y},t)$ we denote the influence of position $\mathbf{y}$ with the arrow pointing to $\mathbf{A}$ at time t to the discrepancy. This gives

$$\textsc{inf}(\mathbf{y},\mathbf{A},t) = \left((A_1y_1 \cdot A_2y_2)t^{-2} - (A_1y_1 + A_2y_2)t^{-1}\right) H(\mathbf{y},t).$$

With this, one can show that the single vertex discrepancy between Propp machine and random walk only depends on the influence of the “odd chips”. Strictly speaking, we prove

$$f(\mathbf{0},t) - E(\mathbf{0},t) \;=\; \sum_{\mathbf{y}\in\mathbb{Z}^2}\sum_{i\geq 0} \textsc{inf}(\mathbf{y}, \textsc{arr}(\mathbf{y}, s_i(\mathbf{y})), t - s_i(\mathbf{y})).$$

Some further properties of $\textsc{inf}(\mathbf{y},\mathbf{A},t)$ and sums of $\textsc{inf}(\mathbf{y},\mathbf{A},t)$, help to show the following theorem.

Theorem 3.1

$$f(\mathbf{0},t) - E(\mathbf{0},t) \approx \begin{cases} 7.83 \text{ \textit{for clockwise rotor sequences}} \\ 7.28 \text{ \textit{for other rotor sequences.}} \end{cases}$$

This is an upper bound for the discrepancy for all initial configurations. Given successive settings of the rotors, one can construct an initial configuration, that is, number of chips on vertices and rotor directions, which yields exactly these rotor settings. This can be used to construct a configuration whose discrepancy reaches above upper bound. Interestingly, there are worst-case configurations which send only less than ten odd chips at at most three different times per position.

References

[1] Joshua N. Cooper and Joel Spencer. Simulating a random walk with constant error. *Combinatorics, Probability and Computing.* To appear (see `http://arxiv.org/math/0402323`).

[2] Joshua N. Cooper, Benjamin Doerr, Joel Spencer, and Gábor Tardos. Deterministic random walks on the integers. *European Journal of Combinatorics.* To appear (see `http://arxiv.org/math/0602300`).