The rotor-router mechanism for quasirandom walk and aggregation

Jim Propp
(U. Mass. Lowell)

July 10, 2008

(based on articles in progress with Ander Holroyd and Lionel Levine;

with thanks also to Hal Canary, Matt Cook, Dan Hoey, Michael Kleber, Yuval Peres, and Oded Schramm)

Slides for this talk are on-line at
`jamespropp.org/mcqmc08-1.pdf`

I. Overview

Instead of driving a Markov chain with a single continuous CUD process on [0,1] (as in work of Owen and Tribble), we can drive it with many discrete low-discrepancy processes (typically, many copies of the sequence

$$0, 1, 0, 1, ...$$

or similar periodic sequences with very low period).

For some Markov chains this method of "rotor-router simulation" gives us a way to reduce the error of MC simulation, e.g.:

1. Holroyd and Propp have shown that for estimating escape probabilities for two-dimensional random walk, the use of rotors reduces error from $O(1/\sqrt{N})$ to $O((\log N)/N)$, where N is the number of runs (for sequential simulation) or the number of particles (for parallel simulation).

For some Markov chains this method of "rotor-router simulation" gives us a way to reduce the error of MC simulation, e.g.:

2. Cooper and Spencer have shown that for estimating occupation probabilities for random walk in a d-dimensional lattice (associated with discretized heat-flow), the use of rotors reduces error from $O(1/\sqrt{N})$ to $O(1/N)$.

For some Markov chains this method of "rotor-router simulation" gives us a way to reduce the error of MC simulation, e.g.:

3. Levine and Peres have shown that for estimating the typical shape formed by N steps of Internal Diffusion-Limited Aggregation (IDLA), the use of rotors leads to low error. (For MCMC, the error is known to be $O(N^{1/3})$ and thought to be $O(\log N)$; for rotors, Levine and Peres prove the error is $O(\log N)$ and simulation suggests that the error is much smaller than that.)

Although the rotor-router approach seems superficially different from existing approaches to quasi Monte Carlo, there are connections:

1. Some low-discrepancy sequences in $[0,1]$ can be derived from rotor-router walk on infinite (directed) trees, e.g., the van der Corput sequence.

Although the rotor-router approach seems superficially different from existing approaches to quasi Monte Carlo, there are connections:

2. Many rotor-router algorithms are covertly calculating integrals on probability spaces.

Although the rotor-router approach seems superficially different from existing approaches to quasi Monte Carlo, there are connections:

3. Combinatorial analogues of the Koksma-Hlawka inequality play a role in proofs of the results.

II. Rotor-routers

This quasirandomization scheme works best on a small state space, or on a big state space where one has a good idea about how to group together different decisions.

Simplest case:

MCMC: When there is a 2-way choice, choose randomly.

RR: When there is a 2-way choice, choose whichever option you *didn't* choose the last time.

More generally:

MCMC: When there is an m-way choice, choose randomly.

RR: When there is an m-way choice, choose whichever option you chose *least recently*. That is, *cycle* among the m options.

(It's easy to generalize RR to the case of m unequally likely options.)

Typically, the m choices correspond to the m possible transitions from a given state x in a Markov chain to successor states $y_1, \ldots, y_m$ with $p(x, y_i) = 1/m$ for $1 \leq i \leq m$.

We keep track of past choices by an m-state *rotor* associated with the state x.

Rotor-router rule: To move a particle currently in state x, increment the rotor at x by 1 mod m and move the particle to the ith neighbor of x, where i is the new value of the rotor.

(It's easy to extend this to the case in which $m = m_x$ varies from one state x to another.)

Fundamental theorem of finite almost-but-not-quite Markov chains
(applies to rotor-routers and many other schemes, some random and some deterministic):

Let $p(x, y)$ be the irreducible and aperiodic transition kernel for a Markov chain with finite state-space S and stationary measure $\pi(\cdot)$ on S.
Suppose $x_1, x_2, \ldots$ is an infinite sequence in S such that for all x, y in S, x is followed by y with asymptotic frequency $p(x, y)$.
Then for all x in S, x occurs in $x_1, x_2, \ldots$ with asymptotic frequency $\pi(x)$.

(Note that randomness, correlations, etc. play no role.)

Quantitative versions exist too: if the convergence to $p(x, y)$ in the hypothesis of the theorem occurs with errors $O(1/N^c)$, then so does the convergence to $\pi(x)$ in the conclusion of the theorem.

$O(1/N^{1/2})$ is the case of MCMC.

$O(1/N)$ is the case of RR.

Versions of this theorem exist for infinite-state Markov chains as well.

Whether such theorems are **useful** depends on the specific details of the Markov chain.

III. Derandomization of 2D random walk

Simple random walk on $\mathbf{Z}^2$: For any two vertices $v, w \in \mathbf{Z}^2$, the transition probability $p(v, w)$ (the probability that a particle at v moves to w at the next time step) is $\frac{1}{4}$ if w is one of the four nearest neighbors of v and 0 otherwise.

This random walk is *recurrent*: With probability 1, each vertex in $\mathbf{Z}^2$ gets visited infinitely often.

Fact: If a particle starts at $(0,0)$ and does random walk in $\mathbf{Z}^2$ until it either hits $(1,1)$ or returns to $(0,0)$, the probability that it hits $(1,1)$ before returning to $(0,0)$ ("escape") is exactly $\pi/8$.

Hence, if we modify the walk so that whenever the particle arrives at $(1,1)$ it gets shunted immediately to $(0,0)$, then the number of escapes divided by the number of trials (call this denominator N) converges to $\pi/8$ with probability 1, with error falling like $1/\sqrt{N}$.

Equivalently, the number of escapes minus $\pi/8$ times the number of trials (write this "global" discrepancy as D_N) will be on the order of $\pm\sqrt{N}$ if we do N independent random trials.

For $N = 10^4$, under random simulation, we expect $|D_N| \approx 50$.

Under quasirandom simulation with rotor-routers, D_N is provably $O(\log N)$ (rather than $O(\sqrt{N})$).

See demo at

`jamespropp.org/rotor-router/`

(set Graph/Mode to 2-D Walk).

In 10,000 trials, $|D_N| < 0.5$ for $5,070$ of the trials. That is, more than half the time, the number of escapes during the first N trials is equal to the integer closest to $p = \pi/8$ times the number of trials.

We have $|D_N| < 2.05$ for all $N \leq 10^4$.

Is $|D_N|$ bounded? Unknown.

III. Quasirandom diffusion

It can be shown that rotor-router walk is parallelizable.

Put some particles in $\mathbf{Z}^d$, where the sites are equipped with rotors. (For technical reasons, the particles must all start out on the same index-2 sublattice.)

Let the particles do rotor-router walk in parallel for N steps.

Cooper and Spencer show that the difference between (1) the number of particles at a site after N steps of rotor-router walk, and (2) the expected number of particles at a site after N steps of random walk, is bounded by a constant C that doesn't depend on N, or on what the original distribution of particles was, or which way the rotors were originally pointing. All it depends on is d, the dimension of the lattice.

See "Simulating a random walk with constant error", by Joshua Cooper and Joel Spencer (arXiv:`math.CO/0402323`) as well as more recent articles by Benjamin Doerr and others.

V. Quasirandom aggregation

Why Charlie Geyer is right: in all the models considered so far, the Markov chains are "small", and it's unlikely that the "Fundamental theorem of finite almost-but-not-quite Markov chains" is useful for large Markov chains in general.

But...

Can we use rotors to get speed-up of MCMC using RR in **some** "big" Markov chains, where we cannot expect to visit more than an exponentially tiny fraction of the state space?

Here's an example of that.

Internal Diffusion-Limited Aggregation (IDLA): To add a new site to the (initially empty) blob, put the bug at the origin and let it do random walk until it hits an unoccupied site. Adjoin this site to the blob. Repeat.

The states here are the possible blobs. After N steps, one needs $\geq c^{\sqrt{N}}$ states (with $c > 1$) to get a set with probability $\geq 1/2$. So no simulation, random or deterministic, can visit more than a tiny fraction of the state-space.

Theorem (Lawler, 1995): The N-bug IDLA blob in $\mathbf{Z}^2$ is a disk of area N, to within radial fluctuation that are $o(N^{1/3})$.

It appears empirically that the radial fluctuations are actually $O(\ln N)$.

IDLA can be derandomized using rotor-routers in a fairly obvious way: send the bugs ejected from each site North, East, South, West, North, East, South, West, etc.

See demo at

`jamespropp.org/rotor-router/`

(set Graph/Mode to 2-D Aggregation).

Theorem (Levine and Peres): The N-bug rotor-router IDLA blob in $\mathbf{Z}^d$ is a ball of volume N to within radial fluctuation that are $O(\log N)$.

It appears that the radial fluctuations for derandomized IDLA are even smaller than for true IDLA.

E.g., after a million bugs have been added to the system, the inradius is 563.5 and the outradius is 565.1: these figures differ by 1.6 (about three tenths of one percent).

There may be an absolute bound on the difference between the inner and outer radius of the IDLA blob, valid for every N.

VI. Picking Van der Corput's sequence off a tree

Consider a (directed) binary tree in which each node has an outgoing edge marked 0 and an outgoing edge marked 1.

Put a 2-way rotor at each node, so that the first particle to visit node x goes in the 0 direction, the second goes in the 1 direction, and so on, in alternation.

The first particle follows the path 0000...

The second particle follows the path 1000...

The third particle follows the path 0100...

The fourth particle follows the path 11000...

Etc.

Represent the path $b_1 b_2 b_3 \ldots$ by the binary number

$$. \, b_1 \; b_2 \; b_3 \ldots$$

i.e., $\frac{1}{2}b_1 + \frac{1}{4}b_2 + \frac{1}{8}b_3 + \ldots$.

Then the paths followed by the successive particles are

$$0, 1/2, 1/4, 3/4, 1/8, 5/8, 3/8, 7/8, \ldots$$

(the van der Corput sequence).

VII. Covert integration

Consider asymmetric random walk on $\mathbf{Z}$ where with probability 1/2 one jumps 1 to the right and with probability 1/2 one jumps 2 to the left.

Put targets at 0 and 1. With probability 1, the particle gets absorbed at either 0 or 1.

The probability that the particle gets absorbed at 0 is $1/\phi$ with $\phi = \frac{1+\sqrt{5}}{2}$.

This lends itself to rotor-routing with 2-way rotors. See demo at

`jamespropp.org/rotor-router/`

(set Graph/Mode to 1-D Walk).

With rotor-router simulation, the empirical frequency of absorption at 0 goes to $1/\phi$ with error $O(1/N)$ where N is the number of trials.

One can show that this is mathematically equivalent to derandomized Monte Carlo integration of the indicator function $f : [0,1] \to \{0,1\}$ with $f(x) = 1$ for $x \leq 1/\phi$ and $f(x) = 0$ otherwise, where the sample points in $[0,1]$ are the multiples of ϕ modulo 1.

In this equivalence, settings of the rotors correspond to numbers x in $[0,1]$, $f(x)$ corresponds to 1 or 0 according to whether the particle gets absorbed at 0 or not, and $x + \phi \pmod 1$ corresponds to the new setting of the rotors after the particle has taken its walk.

VIII. A combinatorial Koksma-Hlawka inequality

A *harmonic* function on a Markov chain is a function $h(\cdot)$ on the state space such that for all states x, $\sum_y h(y)p(x,y) = h(x)$.

Example: Let v, w be fixed states in a recurrent Markov chain, and let $h(x)$ be the probability that a particle that starts at x will reach v before w. This function is harmonic at all x except v and w.

Many examples of rotor-routing simulation can be construed as problems of computing the value of some harmonic function on the Markov chain.

Theorem (Holroyd-Propp): If one uses N steps of rotor-routing to estimate the value of a harmonic function $h(\cdot)$ at a particular state s, the error is bounded by $1/N$ times

$$\sum |h(x) - h(y)|$$

where the sum is taken over all x, y with $p(x, y) > 0$.

IX. Wrap-up

For most of the settings where rotor-routers work better than MCMC, there other well-known approaches (e.g., the method of relaxation) that work better than both.

So: To what sort of applications (if any) are rotors best suited?

E.g., could they be applied to solutions of large linear systems in the regime where the stochastic approach (the "particle method") is competitive?

For more information, see

`jamespropp.org/quasirandom.html`