Better than random: Quasirandomness for discrete stochastic systems

Jim Propp
(U. Wisconsin)

March 14, 2005

(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
`http://www.math.wisc.edu/~propp/harvard-talk.pdf`

I. Overview

Ordinary Monte Carlo: If some quantity of interest, μ, can be expressed as $E(X)$ for some random variable X, then we can estimate μ by

$$\hat{\mu}_n = (X_1 + X_2 + \ldots + X_n)/n,$$

where $X_1, X_2, \ldots$ are i.i.d. instances of X.

If $\mathrm{Var}(X) < \infty$, the root-mean-square error of our estimate is $O(1/\sqrt{n})$, and if we have a bound on $\mathrm{Var}(X)$, the central limit theorem will give us asymptotic confidence intervals.

What's good about Monte Carlo:

1. It's simple (and often fast).

2. It gives pretty good estimates ($O(1/\sqrt{n})$ isn't bad).

3. It's very general.

What's bad about Monte Carlo:

1. $O(1/\sqrt{n})$ isn't great; we might want something closer to the theoretical ideal of $O(1/n)$.

2. Every now and then, the truth falls outside the confidence interval.

3. Algorithmic random number generators aren't always good approximations to true randomness.

4. Even a good algorithmic random number generator isn't really random, so it's not clear what it means to invoke results like the central limit theorem.

The theory of quasirandomness says that you can sometimes do better by using $(x_1 + \ldots + x_n)/n$ where $x_1, \ldots, x_n$ are properly chosen deterministic samples.

Advantages:

1. We can often achieve typical errors of $O(1/n)$ or $O((\log n)^d/n)$.

2. We can sometimes get "certainty intervals" instead of "confidence intervals".

Disadvantages:

1. extra computational overhead

2. less generality

Key idea behind quasirandomness: Replace randomness by a low-discrepancy property (informally: "evenness" or "fairness").

The simplest quasirandom analogue of a sequence of i.i.d. $U(0,1)$ random variables is a sequence $x_1, x_2, x_3, \ldots$ where x_n is the fractional part of $n\alpha$ for some fixed irrational number α ("irrational rotation").

For any interval I in $[0,1]$, the discrepancy $\#(\{1 \leq k \leq n \ : x_k \in I\})/n - \text{length}(I)$ goes down like $1/n$ (where the analogous discrepancy for an i.i.d. sequence of uniform random numbers in $[0,1]$ would go down like $1/\sqrt{n}$).

Another popular low-discrepancy sequence of real numbers in $[0, 1]$ is the van der Corput sequence

$$\frac{1}{2}, \frac{1}{4}, \frac{3}{4}, \frac{1}{8}, \frac{5}{8}, \frac{3}{8}, \frac{7}{8}, \frac{1}{16}, \ldots .$$

Here

$$x_n = (1/2)a + (1/4)b + (1/8)c + \ldots$$

where

$$n = a + 2b + 4c + \ldots$$

with $a, b, c, \ldots$ all in $\{0, 1\}$.

Continuous quasirandom estimation amounts to approximating the integral

$$\mu = \int f(\mathbf{x})\ d\mathbf{x}$$

by the sum

$$\hat{\mu}_n = \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{x}_i)\ .$$

A typical error bound for quasirandom integration is the Koksma-Hlawka inequality

$$|\hat{\mu}_n - \mu| \leq D_n^*(\mathbf{x}_1, \ldots, \mathbf{x}_n)||f||_{HK}$$

where $D_n^*(\cdot)$ is the "star discrepancy" and $||\cdot||_{HK}$ is "Hardy-Krause total variation".

Discrete quasirandomness:

The simplest quasirandom analogue of a sequence of i.i.d. unbiased bits is the sequence 0,1,0,1,0,1,....

More generally, a quasirandom analogue of a sequence of i.i.d. bits of bias p is the sequence of 0's and 1's whose nth bit is

$$\lfloor np \rfloor - \lfloor (n-1)p \rfloor$$

where $\lfloor x \rfloor$ is the greatest integer $\leq x$.

The discrepancy

$$\frac{\#(\{1 \leq k \leq n \ : x_k = 1\})}{n} - p$$

goes down like $1/n$ (where the analogous discrepancy for an i.i.d. sequence of bits with bias p would go down like $1/\sqrt{n}$).

II. Quasirandom walk

If a particle starts at (0,0) and does unbiased random walk in the infinite square grid, the probability p that it will arrive at (1,1) before it ever returns to (0,0) is $\pi/8$.

If we do n trials, the number of successes divided by the number of trials should be close to $\pi/8$, with an error on the order of $1/\sqrt{n}$.

Equivalently, the number of successes minus $\pi/8$ times the number of trials (write this discrepancy as D_n) should be on the order of $\pm\sqrt{n}$ if we do independent random trials.

For $n = 10^4$, we expect $|D_n| \approx 50$.

Under quasirandom simulation, with "rotor-routers", the n trials aren't independent, or even random — yet D_n seems to be bounded!

See demo at

`http://www.math.wisc.edu/~propp/rotor-router-1.0`

In 10,000 trials, $|D_n| < 0.5$ for $5,070$ of the trials. That is, more than half the time, the number of successes after n trials is equal to the integer closest to $p = \pi/8$ times the number of trials.

Note that the quasirandom simulation is parallelizable, so in a sense this is even faster than Monte Carlo.

We have $|D_n| < 2.05$ for all $n \leq 10^4$.

Does $|D_n|$ stay bounded as $n \to \infty$? Unknown!

For analogous processes in 1 dimension, boundedness of D_n can be proved rigorously using harmonic functions on Markov chains.

Consider an ergodic recurrent discrete Markov chain with state space S, specified by a transition kernel $p(x, y)$. Fix an initial state $s_0 \in S$ and disjoint target sets $S_1, S_2 \subset S$.

Let X be a 0,1-valued random variable that equals 1 when the Markov chain, started from s_0, arrives at S_1 before arriving at S_2, and equals 0 otherwise; and let $\mu = E(X) =$ the probability that the chain hits S_1 before S_2 ("success").

Let $\hat{\mu}_n$ be the proportion of successes achieved in n quasirandom trials using rotor-routers.

Let h be the harmonic function on S defined by $h(s) =$ the probability that the Markov chain started from s hits S_1 before S_2, and define

$$\|\nabla h\| = \sum_x \max\{|h(x) - h(y)| : p(x,y) > 0\}.$$

Theorem (Holroyd and Propp): Suppose

$$\|\nabla h\| < \infty.$$

Then

$$|\hat{\mu}_n - \mu| \leq \|\nabla h\|/n.$$

This gives good results for various kinds of one-dimensional biased and unbiased random walk.

The theorem does not apply to our 2-D example (which has $||\nabla h|| = \infty$).

III. Quasirandom diffusion

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`.

IV. Quasirandom aggregation

Internal Diffusion-Limited Aggregation (IDLA): To add a new bug 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.

Theorem (Lawler, Bramson, and Griffeath, 1992): The n-bug IDLA blob in $\mathbf{Z}^2$ is a disk of radius $\sqrt{n/\pi}$, to within radial fluctuation that are $o(n^{1/2})$.

Theorem (Lawler, 1995): We can replace $o(n^{1/2})$ by $O(n^{1/3})$ in the preceding result.

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

Derandomized IDLA also gives circles.

Theorem (Levine and Peres, 2005): The symmetric difference between the rotor-router blob after n steps and the disk of radius $\sqrt{n/\pi}$ has area $o(n)$.

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 at every time n.

V. Continuous versus discrete quasirandomness

There are numerous links between rotor-based discrete quasirandomness and the kind of continuous quasirandomness used to improve Monte Carlo integration.

We already saw that the rotation process for quasirandom sequences of real numbers in $[0, 1]$ is used in the definition of biased strings of quasirandom bits.

Also, certain rotor-based simulations of discrete systems are essentially Monte Carlo integration of continuous functions in disguise.

One can turn this relationship around, and use rotors to construct quasirandom real numbers.

1. Some 1-dimensional walk and aggregation models built out of unbiased rotors behave in the large like one big biased rotor, whose bias can be irrational.

2. Quasirandom walk on a directed binary tree yields the van der Corput sequence.

One can also do quasirandom simulation of semi-discrete random walk (transitions are discrete, but state space is continuous; this occurs when the allowed steps are incommensurable).

Quasirandom simulation: when leaving a point $x \in \mathbf{R}$ and choosing between steps $s_1, s_2, \ldots$, choose the s_i that was chosen least frequently on earlier visits to $(x - \epsilon, x + \epsilon)$, and move to $x + s_i$.

Such a simulation scheme can exhibit phase transitions with respect to ϵ. When ϵ is too large, the algorithm fails by going into an infinite loop or by exhibiting large bias. A small drop in ϵ can make the bias become drastically smaller or vanish entirely.

V. Conclusion

Note that when we use quasirandom methods to obtain more accurate estimates of $E(X)$, we give up on estimating $\mathrm{Var}(X)$.

This is often a profitable trade-off.

Some quasirandom schemes are by their nature exact (i.e., satisfy an analogue of the law of large numbers). The non-exact schemes are easy to recognize (they fail dramatically for very small examples). The exact schemes are usually at least as good as Monte Carlo.

Sometimes we can prove tight bounds on discrepancy.

Computer random-number generators are non-random anyway!

Why settle for one-size-fits-all pseudo-randomness, if you can have the perfect kind of quasirandomness for your model?

When available, quasirandom methods often:

1. are fast;
2. give small errors; and
3. admit deterministic errors-bounds.

For more information, see

`http://www.math.wisc.edu/~propp/quasirandom.html`