Local-to-Global Phenomena for Rotor-Routing (soon to be http://jamespropp.org/glpz.pdf)

by Jim Propp (UMass Lowell)

(describing joint work with
Giuliano Giacaglia (MIT),
Lionel Levine (MIT/Microsoft), and
Linda Zayas-Palmer (MIT),
supported by MIT UROP and NSF)

MIT Combinatorics Seminar
April 7, 2010
(slides at http://jamespropp.org/mitcomb10a.pdf)

Part I. Background and statement of results

Let G be a finite directed graph.
Assume that associated with each vertex v is an infinite list, or **stack**, of neighbors of v:

$$v^{(1)}, v^{(2)}, v^{(3)}, \ldots$$

Assume we also have a walker/particle/chip located at a particular vertex.
To execute a *stack-walk move* when the particle is at v:

- **move** the particle to the neighbor of v pointed to by the top entry in the stack at v, and
- **pop** that entry from the stack at v.

That is, the particle travels from v to $v^{(i)}$ after its ith departure from v.

Special case: Wilson stacks

Suppose the stack at each vertex v is an IID sequence, where the probability of the event $v^{(i)} = w$ is $p(v, w)$ (where $\sum_w p(v, w) = 1$).

Suppose furthermore that all the stacks are independent of one another.

Then the itinerary of the particle is just a (possibly biased) **random walk** on G.

This is David Wilson's stacks model for the Markov chain with transition probabilities $p(v, w)$; see `http://dbwilson.com/ja/tau.ps`

Special case: Rotor-routing

Suppose the stack at each vertex v is a periodic sequence.

Then the itinerary of the particle is a **rotor-router walk** (or "rotor walk") on G.

See "Chip-Firing and Rotor-Routing on Directed Graphs" (Alexander Holroyd, Lionel Levine, Karola Mészáros, Yuval Peres, James Propp, and David Wilson), arXiv:0801.3306 .

This rotor-walk reflects many of the properties of the random walk with transition probabilities $p(v, w)$, where $p(v, w)$ is the density of $\{i : v^{(i)} = w\}$ in $\mathbf{N}$; "low local discrepancy implies low global discrepancy". See "Rotor Walks and Markov Chains" (Alexander Holroyd and James Propp), arXiv:0904.4507 .

Sources, targets, and hitting sequences

Assume we have a designed **source** vertex s and one or more **target** vertices $t_i \neq s$, where the stack at a target vertex t_i is

$$s, s, s, \ldots$$

(so that when the particle leaves a target vertex it goes immediately back to the source vertex).

The **hitting sequence** is the sequence of targets that the particle visits.

Example of a hitting sequence

Vertex set: $\{0, 1, 2, 3, 4\}$ (source is 1, targets are 0 and 4)

Initial stack at 1: $0, 2, 0, 2, 0, 2, \ldots$ (period 2)
Initial stack at 2: $1, 3, 1, 3, 1, 3, \ldots$ (period 2)
Initial stack at 3: $4, 2, 4, 2, 4, 2, \ldots$ (period 2)

Stack-walk: 1, **0**, 1, 2, 1, **0**, 1, 2, 3, **4**,
1, **0**, 1, 2, 1, **0**, 1, 2, 3, 2, 1, **0**, 1, 2, 3, **4**, ...
(eventually periodic with period 16)

Hitting sequence: $\mathbf{0}, \mathbf{0}, \mathbf{4}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{4}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{4}, \ldots$ (period 4)

Global periodicity from local periodicity

Theorem 1 (Periodicity Theorem): If all stacks are periodic, so is the hitting sequence.

Remark: It is trivial that if all stacks are periodic, the absorption sequence is *eventually* periodic. Theorem 1 is a stronger claim.

Remark: It is *false* that if all stacks are periodic, then the *walk* periodic (see the preceding example).

Palindromicity

Call a periodic sequence **palindromic** if its fundamental period is the same forwards and backwards.

That is, if we define the **reversal** of a periodic sequence

$$a_1, a_2, \ldots, a_{p-1}, a_p, a_1, a_2, \ldots, a_{p-1}, a_p, \ldots$$

as

$$a_p, a_{p-1}, \ldots, a_2, a_1, a_p, a_{p-1}, \ldots, a_2, a_1, \ldots$$

then a periodic sequence is palindromic iff it is its own reversal.

Examples and non-examples of palindromicity

Not palindromic: 1, 2, 1, 2, . . . (period 2)
(reversal is 2, 1, 2, 1, . . .)

Palindromic: 1, 2, 1, 1, 2, 1, . . . (period 3)

Palindromic: 1, 2, 2, 1, 1, 2, 2, 1, . . . (period 4)
(this is how service alternates in a tennis tiebreaker)

(Aside: Thanks to Peter Winkler, who launched this whole investigation by suggesting that “tennis rotors” would be worth studying.)

Global palindromicity from local palindromicity

Theorem 3 (Periodicity Theorem): If all stacks are palindromic, so is the hitting sequence.

Global reversal from local reversal

Theorem 3 is a consequence of a more general theorem:

Theorem 2 (Reversal Theorem): Reversing all the (periodic) stacks results in reversal of the (periodic) hitting sequence.

Example of Reversal Theorem

Reverse the earlier example of a hitting sequence (slide 6):

Vertex set: $\{0, 1, 2, 3, 4\}$ (source is 1, targets are 0 and 4)

Initial stack at 1: $2, 0, 2, 0, 2, 0, \ldots$ (period 2)
Initial stack at 2: $3, 1, 3, 1, 3, 1, \ldots$ (period 2)
Initial stack at 3: $2, 4, 2, 4, 2, 4, \ldots$ (period 2)

Stack-walk: 1, 2, 3, 2, 1, **0**, 1, 2, 3, **4**, 1, **0**, 1, 2, 1, **0**, ...

Hitting sequence: $\mathbf{0}, \mathbf{4}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{4}, \mathbf{0}, \mathbf{0}, \ldots$
(reversal of $\mathbf{0}, \mathbf{0}, \mathbf{4}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{4}, \mathbf{0}, \ldots$)

Part II. Sketch of proof of Theorem 1 (Periodicity Theorem)

Main tools for Theorem 1:

- Abelian property for rotor-routing (Holroyd-Levine-Mészáros-Peres-Propp-Wilson)
- Notion of equivalence of rotor-configurations (GLPZ)
- Combinatorial characterization of equivalence (GLPZ)

By making multiple copies of arcs (directed edges) when necessary, we may assume that the number of arcs emanating from each vertex v in G equals the period of the stack at v.

Instead of a one-sided infinite stack of neighbors of v, we'll associate with v a two-sided infinite stack of arcs emanating from v.

Doubly-infinite stacks of arcs

Before the particle leaves a vertex v, the stack at v is

$$\ldots, c, d | e, f, g, \ldots$$

After the particle leaves v, the stack at v is

$$\ldots, c, d, e | f, g, \ldots$$

and the particle leaves v along arc e.

What appears before the divider is the **Past**;
what appears after the divider is the **Future**.

It's helpful to imagine that the divider is a moving pointer, and that the entries in the stack don't move or change.

Retrospective rotors

The last entry of **Past** shows which neighbor the particle went to the **last** time it visited v (assuming the particle has been to v before). We call this the **(retrospective) rotor** at v.

When we move the particle from v, we advance the retrospective rotor at v by "one click" and move the particle to the neighbor of v that the *new* setting of the rotor at v points to.

Note that when we advance the rotor at v, we may obtain an arc that points to the same neighbor of v as before.

Example of a hitting sequence, revisited

Initial stack at 1: $\ldots, 0, 2, 0, 2 | 0, 2, 0, 2, \ldots$
Initial stack at 2: $\ldots, 1, 3, 1, 3 | 1, 3, 1, 3, \ldots$
Initial stack at 3: $\ldots, 4, 2, 4, 2 | 4, 2, 4, 2, \ldots$

Initial rotor at 1: 2
Initial rotor at 2: 3
Initial rotor at 3: 2

The crux of the problem

After the particle has passed from the source at 1 to the respective targets 0, 0, 4, 0 (before returning to 1 each time), the setting of the rotors is

Rotor at 1: 0
Rotor at 2: 3
Rotor at 3: 4

This is not the same as the rotor configuration we saw at the start of the process, so why should we expect the hitting sequence to continue periodically thereafter?

Particle addition operators

For any vertex v and any rotor-configuration ρ, we can add a chip at v and let it do rotor-walk until it hits a target vertex, leaving in its wake a new rotor-configuration ρ'.

The **particle addition operator** E_v is the map that sends ρ to ρ'.

It can be shown that the particle addition operators E_v commute with one another (the **abelian property**): $E_v E_w = E_w E_v$.

That is, $E_v E_w \rho = E_w E_v \rho$ for all vertices v, w and all rotor-configurations ρ.

Equivalence of rotor-router configurations

If C is any chip-configuration (i.e., a mapping from $V(G)$ to the non-negative integers, i.e., a multiset of vertices $\{v_1, v_2, \ldots, v_k\}$), we define $E_C = E_{v_1} E_{v_2} \cdots E_{v_k}$, a self-map from the set of rotor-configurations to itself.

We say rotor-configurations ρ and ρ' are **equivalent** if any of the following equivalent conditions holds:

- There exists *some* chip-configuration C such that $E_C\,\rho = E_C\,\rho'$.
- If C is *the* chip-configuration with $\deg(v) - 1$ chips at v, $E_C\,\rho = E_C\,\rho'$.
- If C is *any* chip-configuration with $\geq \deg(v) - 1$ chips at v, $E_C\,\rho = E_C\,\rho'$.

Example of equivalence

Consider once again the example of a hitting sequence from slide 6.

Let ρ_0 be the initial rotor-setting

1: 2
2: 3
3: 2

and let ρ_4 be the rotor-setting

1: 0
2: 3
3: 4

after 4 particles have gone to the targets.

If we let C be the chip-configuration with 4 chips at vertex 1 (and no other chips), then one can check that $E_C\, \rho_0 = \rho_4 = E_C\, \rho_4$.

The relevance of equivalence to Theorem 1

If you take a snapshot of the rotor-settings at those moments when a target is reached, the sequence of snapshots

$$\rho_0, \rho_1, \rho_2, \rho_3, \rho_4, \ldots$$

(with $\rho_{i+1} = E_s \, \rho_i$ for all $i \geq 0$) is not periodic.

(Fact A:) But if you replace each snapshot by its *equivalence* class, the resulting sequence of equivalence classes

$$[\rho_0], [\rho_1], [\rho_2], [\rho_3], [\rho_4], \ldots$$

is periodic.

(Fact B:) Moreover, if two rotor-configurations ρ, ρ' are equivalent, and we apply E_s to each, the target that the chip arrives at is the same for both rotor-configurations.

Outline of proof of Theorem 1

Define a simple local operation $\rho \mapsto \rho'$ on rotor-configurations (**reverse cycle-popping**) such that equivalence of rotor-configurations is the reflexive-symmetric-transitive closure of the relation $\rho \equiv \rho'$.

Use reverse cycle-popping to prove Fact A.

Use reverse cycle-popping to prove Fact B.

Conclude that the hitting sequence is periodic.

Reverse cycle-popping

If the rotors in ρ contain a cycle passing through vertices $v_1, v_2, \ldots, v_r$, we may **reverse-pop** this cycle by regressing the rotor at each vertex v_i that participates in the cycle.

Example:

$$\begin{array}{ll} 1: & 2 \\ 2: & 3 \\ 3: & 2 \end{array} \rightarrow \begin{array}{ll} 1: & 2 \\ 2: & 1 \\ 3: & 4 \end{array} \rightarrow \begin{array}{ll} 1: & 0 \\ 2: & 3 \\ 3: & 4 \end{array}$$

By reverse-popping the cycle (23) and then reverse-popping the cycle (12), we can turn ρ_0 into ρ_4.

Part III. Sketch of proof of Theorem 2 (Reversal Theorem)

Main tools for Theorem 2:

- Abelian property for rotor-routing (Holroyd-Levine-Mésáros-Peres-Propp-Wilson)
- Notion of equivalence of rotor-configurations (GLPZ)
- Combinatorial characterization of equivalence (GLPZ)
- Inclusion of antiparticles/holes in rotor-router dynamics (as in Friedrich and Levine)

The relevance of equivalence to Theorem 3

Suppose the Future half of the stack at each vertex is initially palindromic.

If you take a snapshot of the rotor-system at those moments when a target is reached, the sequence of snapshots

$$\rho_0, \rho_1, \rho_1, \rho_2, \rho_3, \rho_4, \ldots$$

is not palindromic.

But if you replace each snapshot by its *equivalence* class, the resulting sequence of equivalence classes

$$[\rho_0], [\rho_1], [\rho_2], [\rho_3], [\rho_4], \ldots$$

is palindromic.

The relevance of equivalence to Theorem 2

More generally (i.e., without assumption of palindromicity):

Reversing all the stacks does not reverse the stacks in the sequence of snapshots, but it does reverse them modulo equivalence.

Particles and antiparticles

To move a particle from v: advance the rotor at v (from arc d to arc e, say) and then move the particle along arc e.

To move an antiparticle from v: move the particle along arc e and then *regress* the rotor at v (from arc e back to arc d).

The relevance of antiparticles

Reversing the stacks and then seeing which targets a particle released from s will hit is the same as seeing which targets an antiparticle released from s will hit (without reversing the stacks); the antiparticle will see the same targets, but in the reverse order.

We define operators E_v^- that modify a rotor-configuration by putting an antiparticle at v and letting it do rotor-walk until it hits a target.

Just as E_s sends $[\rho_i]$ to $[\rho_{i+1}]$ for all $i \geq 0$, E_s^- sends $[\rho_{i+1}]$ to $[\rho_i]$ for all $i \geq 0$ (though it generally does not send ρ_{i+1} to ρ_i).

Working with antiparticles

The operators E_u and E_v^- do not commute, but they do commute modulo equivalence.

In particular, the composition $E_s^- E_s$ acts as the identity on equivalence classes.

Furthermore, starting from rotor-configuration ρ, if we add a particle at s and let it walk until it hits a target, and then add an antiparticle at s and let it walk until it hits a target, the antiparticle will traverse the **loop erasure** (cf. Lawler's work on Loop-Erased Random Walk) of the path taken by the particle, ending up at the *same target*.

The final configuration ρ' of the rotors (once the particle and antiparticle have both reached the target) will be equivalent to ρ.

Part IV. The vorticity phenomenon

Consider $\mathbf{Z}^2$ with its usual degree-4 graph structure, with the subset $S = \{(i,j) : \; i \in \mathbf{Z}, \; j \in \mathbf{Z}, \; i^2 + j^2 \leq r^2\}$ (the set of lattice points lying in the disk of radius r centered on $(0,0)$).

Let T be the set of lattice points of $\mathbf{Z}^2$ that are not in S but are adjacent to an element of S.

Let G be the induced subgraph of $\mathbf{Z}^2$ with vertex set $S \cup T$.

Turn this into a source-and-targets problem with source $s = (0,0)$ and target set T.

Angle between successively-hit targets

Assume the stack at each vertex $(i, j) \in S$ starts as $(i+1, j), (i, j+1), (i-1, j), (i, j-1), \ldots$ (circulating counterclockwise with period 4).

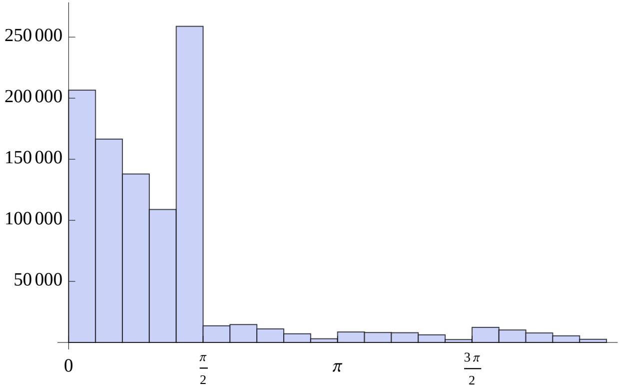
The stack at each vertex in T is just $s, s, s, \ldots$ (as always for a target vertex).

Look at the directed angle between the elements of T that occur as the nth and $n+1$st terms of the target sequence.

Observation: This directed angle (a priori somewhere between 0 and 2π) is nearly always between 0 and $\pi/2$, and is shockingly often equal to $\pi/2$ (exactly).

Histogram

Example: For $r = 10$, if we plot the first million angles, we get



The "unlikely" angles (the ones greater than $\pi/2$) are not equally unlikely; there appear to be four successive "waves" in the histogram, running from $(k-1)\pi/2$ to $k\pi/2$ for $k = 1, 2, 3, 4$.

Global vorticity from local vorticity

It seems that the *local* 90-degree relation between the successive emissions of particles from each vertex v somehow manifests itself in a *global* 90-degree relation between the successive targets that are hit (the spike at 90 degrees, along with the predominance of angles between 0 and 90 degrees).

No theorems yet, but some nice animations for $r = 25$:
http://faculty.uml.edu/jpropp/vortex-short.gif (steps 800 – 999)
http://faculty.uml.edu/jpropp/vortex-long.gif (steps 0 – 999)

Transition: rotor-router aggregation

The preceding observation is analogous to observations made independently by Levine and Cook, pertaining to a rotor-router aggregation model, in which the target boundary is not fixed but moves steadily outward.

Suppose our vertex set is $\mathbf{Z}^2$, our source is $(0,0)$, and the stack at each vertex (i,j) starts as

$$(0,0),(i,j+1),(i+1,j),(i,j-1),(i-1,j),$$
$$(i,j+1),(i+1,j),(i,j-1),(i-1,j),\ldots$$

(continuing with period 4). There is no target set.

The path taken by the particle can be viewed as a process of aggregation, where the growing "blob" consists of all the sites that have been visited up to a particular time. See http://www.cs.uml.edu/∼jpropp/rotor-router-model/#01 (maximize the window and set Graph/Mode to 2-D Aggregation).

Part V: Mysterious global structures for aggregation models (Cook, Friedrich, Hoey, Levine)

The figure http://jamespropp.org/million.gif (image courtesy of Kleber) shows the rotor-router blob when a million chips have been added to the origin.

Hoey was the first to notice that the flower-pattern seen in the middle turns into a repeating pattern in the plane if we view the picture as living in the unit disk of the complex plane and apply the analytic function $z \mapsto 1/z^2$; see Levine's picture http://math.mit.edu/∼levine/gallery/inverted.html .

Why settle for a million?

The web-page http://rotor-router.mpi-inf.mpg.de/ (images developed by Friedrich and Levine) shows the rotor-router blob when *ten billion* chips have been added to the origin.

Zooming in (with Google Maps-style controls) reveals several levels of interesting global structure with no obvious source in the local dynamics.

The number of rotor-router steps involved in the process that gives rise to this picture is about $N^2/2\pi$, where $N =$ ten billion.

That's about 1.6 times 10^{19} elementary operations.

How could such a computation have been completed on contemporary computers?!?

Fast simulation with particles and antiparticles

See http://math.mit.edu/∼levine/fast-simulation.pdf: Friedrich and Levine have a way of quickly *guessing* the approximate number of times each site gets visited during the entire process.

They can detect inconsistencies in their guess, and can fix the inconsistencies by sending *particles and antiparticles* through the system, converging on the truth.

Once they know how many times each site gets visited, they know which sites have been visited and which have not.

Also, once they know how many times each site gets visited, they know how many times each site gets visited mod 4, which determines the final rotor setting at each site.

Friedrich and Levine can prove that their method works, and it works quickly in practice, but they can't prove that it works quickly.

Deviations from circularity

Levine and Peres (arXiv:0704.0688) showed that the aggregation blobs are asymptotically round, although there is much room for improvement of their results; the deviations from roundness seen empirically are much smaller than the bounds of Levine and Peres.

Recently Cook has noted that the deviations from circularity are not erratic, but exhibit beautiful patterns of their own: see http://www.paradise.caltech.edu/cook/Warehouse/ForPropp/LittleWindmill.png

See Cook's Rotor Router Page for other interesting investigations.

Other rotation patterns

Friedrich's website also shows what you get if you use different sorts of rotors.

I find the picture he calls "Irdu" (the third on the page) especially intriguing.

What are we seeing?

And what *aren't* (here and elsewhere on the page) we seeing because we've chosen the wrong color-scheme, or because we're using the wrong display-device, or because we've got the wrong sort of eyes?