The Sandpile Group of a Tree

Lionel Levine

University of California, Berkeley

September 11, 2007

Joint work with Itamar Landau and Yuval Peres.
The Rotor-Router Model

- Deterministic analogue of random walk.
  - Priezzhev-Dhar-Dhar-Krishnamurthy ("Eulerian walkers")
The Rotor-Router Model

- Deterministic analogue of random walk.
  - Priezzhev-Dhar-Dhar-Krishnamurthy ("Eulerian walkers")
- Each site $x \in \mathbb{Z}^2$ has a rotor pointing North, South, East or West.
  (Start all rotors pointing North, say.)
The Rotor-Router Model

- Deterministic analogue of random walk.
  - Priezzhev-Dhar-Dhar-Krishnamurthy ("Eulerian walkers")
- Each site \( x \in \mathbb{Z}^2 \) has a rotor pointing North, South, East or West. 
  (Start all rotors pointing North, say.)
- A particle starts at the origin. At each site it comes to, it
  1. Turns the rotor clockwise by 90 degrees;
  2. Takes a step in direction of the rotor.
The Rotor-Router Model

- Deterministic analogue of random walk.
  - Priezzhev-Dhar-Dhar-Krishnamurthy ("Eulerian walkers")
- Each site \( x \in \mathbb{Z}^2 \) has a rotor pointing North, South, East or West.
  (Start all rotors pointing North, say.)
- A particle starts at the origin. At each site it comes to, it
  1. Turns the rotor clockwise by 90 degrees;
  2. Takes a step in direction of the rotor.
- For a general directed graph, fix a cyclic ordering of the outgoing neighbors.
Rotor-Router Aggregation

- Sequence of lattice regions

\[
A_1 = \{o\}
\]

\[
A_n = A_{n-1} \cup \{x_n\}
\]
Rotor-Router Aggregation

Sequence of lattice regions

\[ A_1 = \{ o \} \]

\[ A_n = A_{n-1} \cup \{ x_n \} \]

where \( x_n \in \mathbb{Z}^d \) is the site at which rotor walk first leaves the region \( A_{n-1} \).
How close to circular?

How fast does

\[ R(n) = \max_{k \leq n} (\text{outrad}(A_k) - \text{inrad}(A_k)) \]

really grow?
How close to circular?

How fast does

\[ R(n) = \max_{k \leq n} (\text{outrad}(A_k) - \text{inrad}(A_k)) \]

really grow?

<table>
<thead>
<tr>
<th>n</th>
<th>R(n)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.822</td>
</tr>
<tr>
<td>10^2</td>
<td>1.588</td>
</tr>
<tr>
<td>10^3</td>
<td>1.637</td>
</tr>
<tr>
<td>10^4</td>
<td>1.683</td>
</tr>
<tr>
<td>10^5</td>
<td>1.724</td>
</tr>
<tr>
<td>10^6</td>
<td>1.741</td>
</tr>
</tbody>
</table>
Three Approaches to Circularity

- Try to bound

\[ R(n) = \max_{k \leq n} (\text{outrad}(A_k) - \text{inrad}(A_k)) \]

for rotor-router aggregation on \( \mathbb{Z}^d \).
Three Approaches to Circularity

- Try to bound

\[ R(n) = \max_{k \leq n} (\text{outrad}(A_k) - \text{inrad}(A_k)) \]

for rotor-router aggregation on \( \mathbb{Z}^d \).

- Two ways to get sharper results:
  - Modify the dynamics: Divisible Sandpile
Three Approaches to Circularity

▶ Try to bound

\[ R(n) = \max_{k \leq n} (\text{outrad}(A_k) - \text{inrad}(A_k)) \]

for rotor-router aggregation on \( \mathbb{Z}^d \).

▶ Two ways to get sharper results:
  ▶ Modify the dynamics: Divisible Sandpile
  ▶ Modify the underlying graph.
    ▶ The tree is easier than the lattice.
Divisible Sandpile

- Start with mass $m$ at the origin.
Divisible Sandpile

- Start with mass $m$ at the origin.
- Each site keeps mass 1, divides excess mass equally among its neighbors.
Divisible Sandpile

- Start with mass $m$ at the origin.
- Each site keeps mass 1, divides excess mass equally among its neighbors.
- As $t \to \infty$, get a limiting region $A_m$ of mass 1, fractional mass on $\partial A_m$, and zero outside.
Divisible Sandpile

- Start with mass $m$ at the origin.
- Each site keeps mass 1, divides excess mass equally among its neighbors.
- As $t \to \infty$, get a limiting region $A_m$ of mass 1, fractional mass on $\partial A_m$, and zero outside.
- **Theorem** (L.-Peres): There are constants $c$ and $c'$ depending only on $d$, such that

$$B_{r-c} \subset A_m \subset B_{r+c'}$$

where $m = \omega_d r^d$. 
Odometer Function

- $u(x) = \text{total mass emitted from } x$. 
Odometer Function

- $u(x) = \text{total mass emitted from } x$.
- Discrete Laplacian:

$$\Delta u(x) = \frac{1}{2d} \sum_{y \sim x} u(y) - u(x)$$
Odometer Function

- $u(x)$ = total mass emitted from $x$.
- Discrete Laplacian:

$$
\Delta u(x) = \frac{1}{2d} \sum_{y \sim x} u(y) - u(x)
$$

= mass received – mass emitted
Odometer Function

- \( u(x) \) = total mass emitted from \( x \).
- Discrete Laplacian:

\[
\Delta u(x) = \frac{1}{2d} \sum_{y \sim x} u(y) - u(x)
\]

= mass received − mass emitted
= \( 1 - m\delta_{ox} \).
Circularity for the Divisible Sandpile

- Dirichlet problem for the odometer function

\[ \Delta u = 1 \quad \text{on} \quad A_m - \{o\} \]
Circularity for the Divisible Sandpile

- Dirichlet problem for the odometer function

\[ \Delta u = 1 \quad \text{on } A_m - \{o\} \]
\[ \Delta u(o) = 1 - m \]

- Additional constraints:
  
  - \( u \geq 0 \) everywhere.
  
  - \( 0 \leq \Delta u < 1 \) on \( \partial A_m \).

These conditions characterize \( A_m \) uniquely!
Circularity for the Divisible Sandpile

- Dirichlet problem for the odometer function

\[ \Delta u = 1 \quad \text{on } A_m - \{o\} \]
\[ \Delta u(o) = 1 - m \]
\[ u = 0 \quad \text{off } A_m. \]
Circularity for the Divisible Sandpile

- Dirichlet problem for the odometer function

\[
\begin{align*}
\Delta u &= 1 & \text{on } A_m - \{o\} \\
\Delta u(o) &= 1 - m \\
u &= 0 & \text{off } A_m.
\end{align*}
\]

- Additional constraints:
  - \( u \geq 0 \) everywhere.
  - \( 0 \leq \Delta u < 1 \) on \( \partial A_m \).

Lionel Levine
The Sandpile Group of a Tree
Circularity for the Divisible Sandpile

- Dirichlet problem for the odometer function
  \[ \Delta u = 1 \quad \text{on } A_m - \{o\} \]
  \[ \Delta u(o) = 1 - m \]
  \[ u = 0 \quad \text{off } A_m. \]

- Additional constraints:
  - \( u \geq 0 \) everywhere.
  - \( 0 \leq \Delta u < 1 \) on \( \partial A_m \).

- These conditions characterize \( A_m \) uniquely!
Continuum Solution

Solution in $\mathbb{R}^d$:

$$u(x) = |x|^2 + mg(x) + C(m)$$

In $\mathbb{R}^d$, the domain $A_m$ is a perfect ball.

In $\mathbb{Z}^d$, since the discrete solution is close to the continuous solution, we get $B_{r-c} \subset A_m \subset B_{r+c'}$. 
Continuum Solution

Solution in $\mathbb{R}^d$:

$$u(x) = |x|^2 + mg(x) + C(m)$$

where

$$g(x) = \begin{cases} 
-a_2 \log |x|, & d = 2 \\
 a_d |x|^{2-d}, & d \geq 3.
\end{cases}$$
Continuum Solution

Solution in $\mathbb{R}^d$:

\[ u(x) = |x|^2 + mg(x) + C(m) \]

where

\[ g(x) = \begin{cases} 
    -a_2 \log |x|, & d = 2 \\
    a_d |x|^{2-d}, & d \geq 3.
\end{cases} \]

In $\mathbb{R}^d$, the domain $A_m$ is a perfect ball.
Continuum Solution

Solution in $\mathbb{R}^d$:

$$u(x) = |x|^2 + m g(x) + C(m)$$

where

$$g(x) = \begin{cases} 
-a_2 \log |x|, & d = 2 \\
 a_d |x|^{2-d}, & d \geq 3.
\end{cases}$$

In $\mathbb{R}^d$, the domain $A_m$ is a perfect ball.

In $\mathbb{Z}^d$, since the discrete solution is close to the continuous solution, we get

$$B_{r-c} \subset A_m \subset B_{r+c'}.$$
Adapting the Proof for Rotors

- Rotor-router odometer:

  \[ u(x) = \text{total number of particles emitted from } x. \]
Adapting the Proof for Rotors

- Rotor-router odometer:
  \[ u(x) = \text{total number of particles emitted from } x. \]

- Instead of \( \Delta u = 1 \), we only know \( -2 \leq \Delta u \leq 4 \).
Adapting the Proof for Rotors

- Rotor-router odometer:
  \[ u(x) = \text{total number of particles emitted from } x. \]

- Instead of \( \Delta u = 1 \), we only know \( -2 \leq \Delta u \leq 4 \).

- Repeating the argument only gives
  \[ B_{cr} \subset A_n \subset B_{c'r}. \]
Adapting the Proof for Multiple Sources

- Start with $n$ particles at each of two sources in $\mathbb{Z}^2$ separated by distance $c\sqrt{n}$. 
Adapting the Proof for Multiple Sources

Start with \( n \) particles at each of two sources in \( \mathbb{Z}^2 \) separated by distance \( c\sqrt{n} \).
Adapting the Proof for Multiple Sources

- Start with $n$ particles at each of two sources in $\mathbb{Z}^2$ separated by distance $c\sqrt{n}$.

- Not so easy to write down an explicit solution to the free boundary problem in $\mathbb{R}^2$. 
The Sandpile Group of a Tree

- Finite rooted tree $T$.  
- Collapse the leaves to a single sink vertex.  
- Add an edge from the root to the sink.
The Sandpile Group of a Tree

- Finite rooted tree \( T \).
- Collapse the leaves to a single sink vertex.
- Add an edge from the root to the sink.

- What are the recurrent configurations?
The Sandpile Group of a Tree

- Finite rooted tree $T$.
- Collapse the leaves to a single sink vertex.
- Add an edge from the root to the sink.

- What are the recurrent configurations?
- What is the structure of the sandpile group?
Critical vertices

$\quad x \in T$ is critical for a chip configuration $u$ if $x \neq s$ and

$$u(x) \leq \# \text{ of critical children of } x.$$  \hspace{1cm} (1)
Critical vertices

- $x \in T$ is critical for a chip configuration $u$ if $x \neq s$ and
  \[ u(x) \leq \# \text{ of critical children of } x. \quad (1) \]

- This is an inductive definition, beginning with the leaves.
Critical vertices

- $x \in T$ is *critical* for a chip configuration $u$ if $x \neq s$ and

$$u(x) \leq \# \text{ of critical children of } x.$$  \hspace{1cm} (1)

- This is an inductive definition, beginning with the leaves.

- **Claim**: A configuration $u$ is recurrent if and only if equality holds in (1) for every critical vertex $x$. 

Lionel Levine
The Sandpile Group of a Tree
Critical vertices

- $x \in T$ is *critical* for a chip configuration $u$ if $x \neq s$ and
  \[ u(x) \leq \# \text{ of critical children of } x. \]  
  \hspace{1cm} (1)

- This is an inductive definition, beginning with the leaves.

**Claim:** A configuration $u$ is recurrent if and only if equality holds in (1) for every critical vertex $x$.

**Proof:** Use the burning algorithm.

- A critical vertex cannot burn before its parent.
Critical vertices

$x \in T$ is critical for a chip configuration $u$ if $x \neq s$ and

$$u(x) \leq \# \text{ of critical children of } x.$$  

This is an inductive definition, beginning with the leaves.

Claim: A configuration $u$ is recurrent if and only if equality holds in (1) for every critical vertex $x$.

Proof: Use the burning algorithm.

- A critical vertex cannot burn before its parent.
- If strict inequality holds at $x$, then $x$ will never be burned.
A Recurrent Configuration on the Regular Ternary Tree

Critical vertices are circled.
Structure of the Sandpile Group

Theorem (L.) Let $T_n$ be a branch of the regular ternary tree of height $n$. Then

$$SP(T_n) \cong \mathbb{Z}_{2^n-1} \oplus \mathbb{Z}_{2^n-1} \oplus \ldots \oplus (\mathbb{Z}_7)^{2^{n-4}} \oplus (\mathbb{Z}_3)^{2^{n-3}}.$$
Theorem (L.) Let $T_n$ be a branch of the regular ternary tree of height $n$. Then

$$SP(T_n) \cong \mathbb{Z}_{2^n-1} \oplus \mathbb{Z}_{2^{n-1}-1} \oplus \ldots \oplus (\mathbb{Z}_7)^{2^{n-4}} \oplus (\mathbb{Z}_3)^{2^{n-3}}.$$ 

Similar decomposition for the $d$-regular tree for any $d$. 

Structure of the Sandpile Group
Subgroup Generated by the Root

Regular ternary tree $T_n$ of height $n$. 

What can we say about the subgroup of $SP(T_n)$ generated by $\hat{r} = \delta_r + e$?

Its elements are constant on levels of the tree.

What about the converse?

Note that if $u$ is recurrent, then $u + \hat{r} = u + (e + \delta_r) = (u + e) + \delta_r = u + \delta_r$.

Multiples of the root in $T_4$: 

Lionel Levine 
The Sandpile Group of a Tree
Subgroup Generated by the Root

- Regular ternary tree $T_n$ of height $n$.
- What can we say about the subgroup of $SP(T_n)$ generated by $\hat{r} = \delta_r + e$?

- Its elements are constant on levels of the tree.
- What about the converse?

Note that if $u$ is recurrent, then $u + \hat{r} = u + (e + \delta_r) = (u + e) + \delta_r = u + \delta_r$.

Multiples of the root in $T_4$: [Lionel Levine] The Sandpile Group of a Tree
Subgroup Generated by the Root

- Regular ternary tree $T_n$ of height $n$.
- What can we say about the subgroup of $SP(T_n)$ generated by $\hat{r} = \delta_r + e$?
- Its elements are constant on levels of the tree.
Subgroup Generated by the Root

- Regular ternary tree $T_n$ of height $n$.
- What can we say about the subgroup of $SP(T_n)$ generated by $\hat{r} = \delta_r + e$?
- Its elements are constant on levels of the tree.
- What about the converse?
Subgroup Generated by the Root

- Regular ternary tree $T_n$ of height $n$.
- What can we say about the subgroup of $SP(T_n)$ generated by $\hat{r} = \delta_r + e$?
- Its elements are constant on levels of the tree.
- What about the converse?
- Note that if $u$ is recurrent, then

$$u + \hat{r} = u + (e + \delta_r)$$

$$= (u + e) + \delta_r$$

$$= u + \delta_r.$$
Subgroup Generated by the Root

- Regular ternary tree $T_n$ of height $n$.
- What can we say about the subgroup of $SP(T_n)$ generated by $\hat{r} = \delta_r + e$?
- Its elements are constant on levels of the tree.
- What about the converse?
- Note that if $u$ is recurrent, then

$$u + \hat{r} = u + (e + \delta_r) = (u + e) + \delta_r = u + \delta_r.$$ 

- Multiples of the root in $T_4$:

<table>
<thead>
<tr>
<th>$\hat{r}$</th>
<th>$2\hat{r}$</th>
<th>$3\hat{r}$</th>
<th>$4\hat{r}$</th>
<th>$5\hat{r}$</th>
<th>$6\hat{r}$</th>
<th>$7\hat{r}$</th>
<th>$8\hat{r}$</th>
<th>$9\hat{r}$</th>
<th>$10\hat{r}$</th>
<th>$11\hat{r}$</th>
<th>$12\hat{r}$</th>
<th>$13\hat{r}$</th>
<th>$14\hat{r}$</th>
<th>$15\hat{r} = e$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Lionel Levine The Sandpile Group of a Tree
The Order of $\hat{r}$

- A recurrent configuration constant on levels has the form

$$u = (2, \ldots, 2, 0, a_1, \ldots, a_k)$$

with $a_i \in \{1, 2\}$. 

Lemma: $\hat{r}$ consists of all recurrent configurations that are constant on levels of the tree.

In particular, $\hat{r}$ has order $n - 1 \sum_{k=0}^{\infty} 2^k = 2^n - 1$. 

Lionel Levine
The Sandpile Group of a Tree
A recurrent configuration constant on levels has the form

\[ u = (2, \ldots, 2, 0, a_1, \ldots, a_k) \]

with \( a_i \in \{1, 2\} \).

**Lemma:** \( \hat{r} \) consists of all recurrent configurations that are constant on levels of the tree.
The Order of $\hat{r}$

- A recurrent configuration constant on levels has the form
  \[ u = (2, \ldots, 2, 0, a_1, \ldots, a_k) \]
  with $a_i \in \{1, 2\}$.

- **Lemma:** $\hat{r}$ consists of all recurrent configurations that are constant on levels of the tree.

- In particular, $\hat{r}$ has order
  \[ \sum_{k=0}^{n-1} 2^k = 2^n - 1. \]
The Sandpile Group of a Tree, In Terms of its Branches

- **Lemma:** Let $T$ be any finite tree, with principal branches $T_1, \ldots, T_k$. 

- **Proof sketch:** Map $(u_1, \ldots, u_k) \mapsto (u_1, \ldots, u_k)$. After modding out by $\hat{r}$, the branches become independent. Since $(k+1)\hat{r} \mapsto (\hat{r}_1, \ldots, \hat{r}_k)$ we have to mod out by this on the right.
The Sandpile Group of a Tree, In Terms of its Branches

Lemma: Let $T$ be any finite tree, with principal branches $T_1, \ldots, T_k$. Then

$$SP(T)/\langle \hat{r} \rangle \cong \bigoplus_{i=1}^{k} SP(T_i)/((\hat{r}_1, \ldots, \hat{r}_k))$$

where $r, r_i$ are the roots of $T, T_i$ respectively.
The Sandpile Group of a Tree, In Terms of its Branches

Lemma: Let $T$ be any finite tree, with principal branches $T_1, \ldots, T_k$. Then

$$SP(T)/(\hat{r}) \cong \bigoplus_{i=1}^{k} SP(T_i)/((\hat{r}_1, \ldots, \hat{r}_k))$$

where $r, r_i$ are the roots of $T, T_i$ respectively.

Proof sketch: Map $\left( \begin{array}{c} a \\ u_1, \ldots, u_k \end{array} \right) \mapsto (u_1, \ldots, u_k)$. 

Lionel Levine
The Sandpile Group of a Tree
The Sandpile Group of a Tree, In Terms of its Branches

Lemma: Let $T$ be any finite tree, with principal branches $T_1, \ldots, T_k$. Then

$$SP(T)/\langle \hat{r} \rangle \simeq \bigoplus_{i=1}^{k} SP(T_i)/((\hat{r}_1, \ldots, \hat{r}_k))$$

where $r, r_i$ are the roots of $T, T_i$ respectively.

Proof sketch: Map $\left( \begin{array}{c} a \\ u_1, \ldots, u_k \end{array} \right) \mapsto (u_1, \ldots, u_k)$.

- After modding out by $\hat{r}$, the branches become independent.
Lemma: Let $T$ be any finite tree, with principal branches $T_1, \ldots, T_k$. Then

$$SP(T)/(\hat{r}) \simeq \bigoplus_{i=1}^{k} SP(T_i)/(((\hat{r}_1, \ldots, \hat{r}_k))$$

where $r, r_i$ are the roots of $T, T_i$ respectively.

Proof sketch: Map $\left( \begin{array}{c} a \\ u_1, \ldots, u_k \end{array} \right) \mapsto (u_1, \ldots, u_k)$.

- After modding out by $\hat{r}$, the branches become independent.
- Since $(k + 1)\hat{r} \mapsto (\hat{r}_1, \ldots, \hat{r}_k)$ we have to mod out by this on the right.
Lemma: Let \( T_n \) be the regular ternary tree of height \( n \). Then

\[
SP(T_n) = \mathbb{Z}_{2^n-1} \oplus SP(T_{n-1})^2 / \mathbb{Z}_{2^{n-1}-1}.
\]
**Strengthening to a Direct Sum**

- **Lemma:** Let $T_n$ be the regular ternary tree of height $n$. Then

$$SP(T_n) = \mathbb{Z}_{2^n - 1} \oplus SP(T_{n-1})^2 / \mathbb{Z}_{2^{n-1} - 1}.$$  

- **Proof:** Need a projection map $p : SP(T_n) \to (\hat{r})$.  

Lionel Levine  
**The Sandpile Group of a Tree**
Strengthening to a Direct Sum

Lemma: Let $T_n$ be the regular ternary tree of height $n$. Then

$$SP(T_n) = \mathbb{Z}_2^{2n-1} \oplus SP(T_{n-1})^2 / \mathbb{Z}_{2^n-1}. $$

Proof: Need a projection map $p : SP(T_n) \rightarrow (\hat{r})$.

Use the symmetrization map

$$p(u)(x) = 2^{n+1-|x|} \sum_{|y|=|x|} u(y).$$
Strengthening to a Direct Sum

Lemma: Let $T_n$ be the regular ternary tree of height $n$. Then

$$SP(T_n) = \mathbb{Z}_{2^n-1} \oplus SP(T_{n-1})^2 / \mathbb{Z}_{2^{n-1}-1}.$$ 

Proof: Need a projection map $p : SP(T_n) \rightarrow (\hat{r})$.

Use the symmetrization map

$$p(u)(x) = 2^{n+1-|x|} \sum_{|y|=|x|} u(y).$$

Note if $u$ is already constant on levels, then

$$p(u) = 2^n u = u$$

since $u = k\hat{r}$ and $\hat{r}$ has order $2^n - 1$. □
Factoring Into Cyclic Subgroups

- $SP(T_2) = \mathbb{Z}_3$. 

- $SP(T_3) = \mathbb{Z}_7 \oplus SP(T_2)^2/\mathbb{Z}_3 = \mathbb{Z}_7 \oplus \mathbb{Z}_3$.

- $SP(T_4) = \mathbb{Z}_{15} \oplus SP(T_3)^2/\mathbb{Z}_7 = \mathbb{Z}_{15} \oplus \mathbb{Z}_7 \oplus \mathbb{Z}_3^3$.

- $SP(T_5) = \mathbb{Z}_{31} \oplus SP(T_4)^2/\mathbb{Z}_{15} = \mathbb{Z}_{31} \oplus \mathbb{Z}_{15} \oplus \mathbb{Z}_7 \oplus \mathbb{Z}_4^3$.

- $SP(T_n) = \mathbb{Z}_{2^{n-1}} \oplus \mathbb{Z}_{2^{n-1}-1} \oplus \ldots \oplus (\mathbb{Z}_7)^{2^{n-4}} \oplus (\mathbb{Z}_3)^{2^{n-3}}$. 

Lionel Levine  
The Sandpile Group of a Tree
Factoring Into Cyclic Subgroups

- $SP(T_2) = \mathbb{Z}_3$.
- $SP(T_3) = \mathbb{Z}_7 \oplus SP(T_2)^2 / \mathbb{Z}_3 = \mathbb{Z}_7 \oplus \mathbb{Z}_3$. 

Lionel Levine
Factoring Into Cyclic Subgroups

- $SP(T_2) = \mathbb{Z}_3$.
- $SP(T_3) = \mathbb{Z}_7 \oplus SP(T_2)^2 / \mathbb{Z}_3 = \mathbb{Z}_7 \oplus \mathbb{Z}_3$.
- $SP(T_4) = \mathbb{Z}_{15} \oplus SP(T_3)^2 / \mathbb{Z}_7 = \mathbb{Z}_{15} \oplus \mathbb{Z}_7 \oplus \mathbb{Z}_3^2$. 

Lionel Levine  The Sandpile Group of a Tree
Factoring Into Cyclic Subgroups

- $SP(T_2) = \mathbb{Z}_3$.
- $SP(T_3) = \mathbb{Z}_7 \oplus SP(T_2)^2 / \mathbb{Z}_3 = \mathbb{Z}_7 \oplus \mathbb{Z}_3$.
- $SP(T_4) = \mathbb{Z}_{15} \oplus SP(T_3)^2 / \mathbb{Z}_7 = \mathbb{Z}_{15} \oplus \mathbb{Z}_7 \oplus \mathbb{Z}_3^2$.
- $SP(T_5) = \mathbb{Z}_{31} \oplus SP(T_4)^2 / \mathbb{Z}_{15} = \mathbb{Z}_{31} \oplus \mathbb{Z}_{15} \oplus \mathbb{Z}_7^2 \oplus \mathbb{Z}_3^4$. 

Lionel Levine  The Sandpile Group of a Tree
Factoring Into Cyclic Subgroups

- $SP(T_2) = \mathbb{Z}_3$.
- $SP(T_3) = \mathbb{Z}_7 \oplus SP(T_2)^2 / \mathbb{Z}_3 = \mathbb{Z}_7 \oplus \mathbb{Z}_3$.
- $SP(T_4) = \mathbb{Z}_{15} \oplus SP(T_3)^2 / \mathbb{Z}_7 = \mathbb{Z}_{15} \oplus \mathbb{Z}_7 \oplus \mathbb{Z}_3^2$.
- $SP(T_5) = \mathbb{Z}_{31} \oplus SP(T_4)^2 / \mathbb{Z}_{15} = \mathbb{Z}_{31} \oplus \mathbb{Z}_{15} \oplus \mathbb{Z}_7^2 \oplus \mathbb{Z}_3^4$.

... 

- $SP(T_n) = \mathbb{Z}_{2^n-1} \oplus \mathbb{Z}_{2^{n-1}-1} \oplus \ldots \oplus (\mathbb{Z}_7)^{2^{n-4}} \oplus (\mathbb{Z}_3)^{2^{n-3}}$. 

Lionel Levine 

The Sandpile Group of a Tree
Entropy

- Number of recurrent states:

\[ \log_2 |SP(T_n)| \sim c2^n \]
Entropy

- Number of recurrent states:

\[
\log_2 |SP(T_n)| \sim c2^n
\]

where

\[
c = \frac{\log_2 3}{4} + \frac{\log_2 7}{8} + \ldots + \frac{\log_2 (2^k - 1)}{2^k} + \ldots
\]

So the probability that a stable state is recurrent is about

\[
\left(\frac{2}{c^3}\right)^{2^n} = \left(0.858\right)^{2^n}
\]
Number of recurrent states:

\[ \log_2 |SP(T_n)| \sim c2^n \]

where

\[
c = \frac{\log_2 3}{4} + \frac{\log_2 7}{8} + \ldots + \frac{\log_2 (2^k - 1)}{2^k} + \ldots
\]

\[
= 2 - \frac{1}{\log 2} \sum_{n=2}^{\infty} \frac{1}{(n-1)(2^n - 1)}
\]

\[
\approx 1.364.
\]
Entrophy

- Number of recurrent states:

\[ \log_2 |SP(T_n)| \sim c2^n \]

where

\[ c = \frac{\log_2 3}{4} + \frac{\log_2 7}{8} + \ldots + \frac{\log_2(2^k - 1)}{2^k} + \ldots \]

\[ = 2 - \frac{1}{\log 2} \sum_{n=2}^{\infty} \frac{1}{(n-1)(2^n - 1)} \]

\[ \approx 1.364. \]

- So the probability that a stable state is recurrent is about

\[ \left( \frac{2^c}{3} \right)^{2^n} = (0.858)^{2^n}. \]
“Physical” Consequences

- Three ways to measure the size of an avalanche:
  - $R =$ diameter of the set of sites that topple.
  - $M =$ number of sites that topple.
  - $\tau =$ total number of topplings.
“Physical” Consequences

Three ways to measure the size of an avalanche:

- $R =$ diameter of the set of sites that topple.
- $M =$ number of sites that topple.
- $\tau =$ total number of topplings.

Starting from a uniform recurrent state in $T_n$, add a single grain at at the root.
“Physical” Consequences

- Three ways to measure the size of an avalanche:
  - $R =$ diameter of the set of sites that topple.
  - $M =$ number of sites that topple.
  - $\tau =$ total number of topplings.

- Starting from a uniform recurrent state in $T_n$, add a single grain at at the root. Then for $r \leq n$

$$\mathbb{P}(R \geq r) \sim 2^{-r}.$$
“Physical” Consequences

Three ways to measure the size of an avalanche:

- $R =$ diameter of the set of sites that topple.
- $M =$ number of sites that topple.
- $\tau =$ total number of topplings.

Starting from a uniform recurrent state in $T_n$, add a single grain at at the root. Then for $r \leq n$ and $m, t \leq 2^n$

\[
\mathbb{P}(R \geq r) \asymp 2^{-r}.
\]
\[
\mathbb{P}(M \geq m) \asymp 1/m.
\]
\[
\mathbb{P}(\tau \geq t) \asymp 1/t.
\]
Rotor-Router Aggregation on the Tree

Let $A_m$ be the region formed by rotor-router aggregation on the infinite $d$-regular tree, starting from $m$ chips at the origin.
Let $A_m$ be the region formed by rotor-router aggregation on the infinite $d$-regular tree, starting from $m$ chips at the origin.

**Theorem** (Landau-L.) If the initial rotor configuration is acyclic, then

$$A_{b_n} = B_n$$
Rotor-Router Aggregation on the Tree

Let $A_m$ be the region formed by rotor-router aggregation on the infinite $d$-regular tree, starting from $m$ chips at the origin.

**Theorem** (Landau-L.) If the initial rotor configuration is acyclic, then

$$A_{b_n} = B_n$$

where $B_n$ is the ball of radius $n$ centered at the origin, and $b_n = \#B_n$. 

Lionel Levine
The Sandpile Group of a Tree
Rotor-Router Aggregation on the Tree

Let $A_m$ be the region formed by rotor-router aggregation on the infinite $d$-regular tree, starting from $m$ chips at the origin.

**Theorem** (Landau-L.) If the initial rotor configuration is acyclic, then

$$A_{b_n} = B_n$$

where $B_n$ is the ball of radius $n$ centered at the origin, and $b_n = \#B_n$.

In particular, if $b_n < m < b_{n+1}$, then

$$B_n \subset A_m \subset B_{n+1}.$$
The Rotor-Router Group

- Finite directed graph $G$.
- Rotors point in direction of last exit.
The Rotor-Router Group

- Finite directed graph $G$.
- Rotors point in direction of last exit.
  $\implies$ recurrent states have no oriented cycles.

Fact: $\text{RR}(G) \cong \text{SP}(G)$. 

Lionel Levine
The Rotor-Router Group

- Finite directed graph $G$.
- Rotors point in direction of last exit.  
  $\implies$ recurrent states have no oriented cycles.
- In fact, a state is recurrent if and only if the rotors form an oriented spanning tree.
The Rotor-Router Group

- Finite directed graph $G$.
- Rotors point in direction of last exit. $\implies$ recurrent states have no oriented cycles.
- In fact, a state is recurrent if and only if the rotors form an oriented spanning tree.
- For each vertex $x$, get a permutation $e_x$ on spanning trees given by adding a chip at $x$ and routing it to the sink.
The Rotor-Router Group

- Finite directed graph $G$.
- Rotors point in direction of last exit. $\implies$ recurrent states have no oriented cycles.
- In fact, a state is recurrent if and only if the rotors form an oriented spanning tree.
- For each vertex $x$, get a permutation $e_x$ on spanning trees given by adding a chip at $x$ and routing it to the sink.
- $RR(G) = \text{subgroup of the permutation group of spanning trees}$ generated by $\{e_x\}_{x \in V(G)}$. 

Fact: $RR(G) \simeq SP(G)$. 

Lionel Levine
The Rotor-Router Group

- Finite directed graph $G$.
- Rotors point in direction of last exit. $\implies$ recurrent states have no oriented cycles.
- In fact, a state is recurrent if and only if the rotors form an oriented spanning tree.
- For each vertex $x$, get a permutation $e_x$ on spanning trees given by adding a chip at $x$ and routing it to the sink.
- $RR(G) =$ subgroup of the permutation group of spanning trees generated by $\{e_x\}_{x \in V(G)}$.
- **Fact:** $RR(G) \simeq SP(G)$. 
Holroyd-Propp Invariant

A function $H$ on the vertices of $T$ is harmonic if

$$H(x) = \frac{1}{\deg(x)} \sum_{y \sim x} H(y)$$

for all $x$. 
Holroyd-Propp Invariant

- A function $H$ on the vertices of $T$ is harmonic if

$$H(x) = \frac{1}{\deg(x)} \sum_{y \sim x} H(y)$$

for all $x$.

- Starting chip configuration $u$, ending configuration $v$. 

Lionel Levine  The Sandpile Group of a Tree
A function $H$ on the vertices of $T$ is harmonic if
\[
H(x) = \frac{1}{\deg(x)} \sum_{y \sim x} H(y)
\]
for all $x$.

Starting chip configuration $u$, ending configuration $v$.

**Lemma:** If $H$ is harmonic, and the initial and final rotor configurations are the same, then
\[
\sum_{x \in T} H(x)u(x) = \sum_{x \in T} H(x)v(x).
\]
Proof of the Theorem

Fix a leaf $z \in T_n$, and let

$$H(x) = \mathbb{P}_x(X_\tau = z)$$
Proof of the Theorem

Fix a leaf $z \in T_n$, and let

$$H(x) = \mathbb{P}_x(X_\tau = z)$$

where $\tau$ is the first time for SRW to hit either a leaf or the sink.
Proof of the Theorem

Fix a leaf $z \in T_n$, and let

$$H(x) = \mathbb{P}_x(X_\tau = z)$$

where $\tau$ is the first time for SRW to hit either a leaf or the sink.

Easy calculation: $H(r) = \frac{1}{2^n-1}$.
Proof of the Theorem

Fix a leaf $z \in T_n$, and let

$$H(x) = \mathbb{P}_x(X_\tau = z)$$

where $\tau$ is the first time for SRW to hit either a leaf or the sink.

Easy calculation: $H(r) = \frac{1}{2^n-1}$.

Time change:

Stop each particle either when it reaches an unoccupied site, or returns to the root.
Proof of the Theorem

Fix a leaf $z \in T_n$, and let

$$H(x) = P_x(X_\tau = z)$$

where $\tau$ is the first time for SRW to hit either a leaf or the sink.

Easy calculation: $H(r) = \frac{1}{2^n - 1}$.

Time change:

- Stop each particle either when it reaches an unoccupied site, or returns to the root.
- Get a new aggregation process $A'_m = A_{f(m)}$. 

Proof of the Theorem

▶ Fix a leaf $z \in T_n$, and let

$$H(x) = P_x(X_\tau = z)$$

where $\tau$ is the first time for SRW to hit either a leaf or the sink.

▶ Easy calculation: $H(r) = \frac{1}{2^n - 1}$.

▶ Time change:
  ▶ Stop each particle either when it reaches an unoccupied site, or returns to the root.
  ▶ Get a new aggregation process $A'_m = A_{f(m)}$.
▶ Enough to show $A'_{c_n} = B_n$ for some sequence $c_n$. 
Proof of the Theorem

▶ Induct on $n$ to show $A'_{c_n} = B_n$. 
Proof of the Theorem

- Induct on $n$ to show $A'_{c_{n}} = B_{n}$.
- With $B_{n-1}$ occupied, start with $3(2^n - 1)$ chips at the root.

Since $\hat{r}$ has order $2^{n-1}$, initial and final rotors are the same.

By the Lemma, final weight = initial weight = 1, so exactly one chip ends up at each leaf.

Thus $A'_{c_{n}} = B_{n}$, where $c_{n} = c_{n-1} + 3(2^n - 1)$. 
Proof of the Theorem

- Induct on $n$ to show $A'_{c_n} = B_n$.
- With $B_{n-1}$ occupied, start with $3(2^n - 1)$ chips at the root.
- Since $\hat{r}$ has order $2^n - 1$, initial and final rotors are the same.
Proof of the Theorem

▶ Induct on $n$ to show $A'_{c_n} = B_n$.
▶ With $B_{n-1}$ occupied, start with $3(2^n - 1)$ chips at the root.
▶ Since $\hat{r}$ has order $2^n - 1$, initial and final rotors are the same.
▶ By the Lemma, final weight = initial weight = 1, so exactly one chip ends up at each leaf.
Proof of the Theorem

- Induct on $n$ to show $A'_{c_n} = B_n$.
- With $B_{n-1}$ occupied, start with $3(2^n - 1)$ chips at the root.
- Since $\hat{r}$ has order $2^n - 1$, initial and final rotors are the same.
- By the Lemma, final weight = initial weight = 1, so exactly one chip ends up at each leaf.
- Thus $A'_{c_n} = B_n$, where

$$c_n = c_{n-1} + 3(2^n - 1).$$
Escape Sequences

- Escape sequence

\[ a_j = \begin{cases} 
0, & \text{if the } j^{th} \text{ chip returns to the origin;} \\
1, & \text{if the } j^{th} \text{ chip escapes to infinity.} 
\end{cases} \]
Escape Sequences

- Escape sequence

\[ a_j = \begin{cases} 
0, & \text{if the } j^{th} \text{ chip returns to the origin;} \\
1, & \text{if the } j^{th} \text{ chip escapes to infinity.}
\end{cases} \]

- For \( j \in \{1, 2, 3\} \) write \( a^{(j)} = a_j a_{j+3} a_{j+6} \ldots \)
Escape Sequences

- Escape sequence
  \[ a_j = \begin{cases} 
  0, & \text{if the } j^{th} \text{ chip returns to the origin;} \\
  1, & \text{if the } j^{th} \text{ chip escapes to infinity.} 
\end{cases} \]

- For \( j \in \{1, 2, 3\} \) write \( a^{(j)} = a_j a_{j+3} a_{j+6} \ldots \).

- **Theorem** (Landau-L.) A binary word \( a_1 \ldots a_n \) is an escape sequence for some rotor configuration on the infinite ternary tree if and only if for each \( j \)
  - Every subword of \( a^{(j)} \) of length 3 contains at most 2 ones.
Escape Sequences

- Escape sequence

\[ a_j = \begin{cases} 
0, & \text{if the } j^{th} \text{ chip returns to the origin;} \\
1, & \text{if the } j^{th} \text{ chip escapes to infinity.} 
\end{cases} \]

- For \( j \in \{1, 2, 3\} \) write \( a^{(j)} = a_j a_{j+3} a_{j+6} \ldots \).

- **Theorem** (Landau-L.) A binary word \( a_1 \ldots a_n \) is an escape sequence for some rotor configuration on the infinite ternary tree if and only if for each \( j \)
  - Every subword of \( a^{(j)} \) of length 3 contains at most 2 ones.
  - Every subword of \( a^{(j)} \) of length 7 contains at most 4 ones.
Escape Sequences

- Escape sequence

\[ a_j = \begin{cases} 
0, & \text{if the } j^{th} \text{ chip returns to the origin}; \\
1, & \text{if the } j^{th} \text{ chip escapes to infinity}. 
\end{cases} \]

- For \( j \in \{1, 2, 3\} \) write \( a^{(j)} = a_j a_{j+3} a_{j+6} \ldots \).

- **Theorem** (Landau-L.) A binary word \( a_1 \ldots a_n \) is an escape sequence for some rotor configuration on the infinite ternary tree if and only if for each \( j \)
  - Every subword of \( a^{(j)} \) of length 3 contains at most 2 ones.
  - Every subword of \( a^{(j)} \) of length 7 contains at most 4 ones.
  - \( \ldots \)
  - Every subword of \( a^{(j)} \) of length \( 2^k - 1 \) contains at most \( 2^{k-1} \) ones.
Further Directions and Open Problems

- Aggregation on general trees
  - What takes the place of a ball?
Further Directions and Open Problems

- Aggregation on general trees
  - What takes the place of a ball?
  - On a transient tree, level sets of the function

\[ g(x) = \mathbb{P}_o(T_x < \infty). \]
Further Directions and Open Problems

- Aggregation on general trees
  - What takes the place of a ball?
  - On a transient tree, level sets of the function
    \[ g(x) = \Pr_o(T_x < \infty). \]

- Does there exist a rotor configuration on \( \mathbb{Z}^3 \) which causes every chip to return to the origin in finitely many steps?

Known to exist for \( \mathbb{Z}^2 \) (Jim Propp) and for the \( d \)-regular tree.

Does there exist a rotor configuration on \( \mathbb{Z}^2 \) (or any recurrent graph) which causes infinitely many chips to escape to infinity?

Can show that the proportion that escape is zero.
Further Directions and Open Problems

- Aggregation on general trees
  - What takes the place of a ball?
  - On a transient tree, level sets of the function
    \[ g(x) = \mathbb{P}_o(T_x < \infty). \]

- Does there exist a rotor configuration on \( \mathbb{Z}^3 \) which causes every chip to return to the origin in finitely many steps?
  - Known to exist for \( \mathbb{Z}^2 \) (Jim Propp) and for the \( d \)-regular tree.
Further Directions and Open Problems

- Aggregation on general trees
  - What takes the place of a ball?
  - On a transient tree, level sets of the function 
    \[ g(x) = \mathbb{P}_o(T_x < \infty). \]

- Does there exist a rotor configuration on \( \mathbb{Z}^3 \) which causes every chip to return to the origin in finitely many steps?
  - Known to exist for \( \mathbb{Z}^2 \) (Jim Propp) and for the \( d \)-regular tree.

- Does there exist a rotor configuration on \( \mathbb{Z}^2 \) (or any recurrent graph) which causes infinitely many chips to escape to infinity?
  - Can show that the proportion that escape is zero.

Lionel Levine
The Sandpile Group of a Tree
Further Directions and Open Problems

- Aggregation on general trees
  - What takes the place of a ball?
  - On a transient tree, level sets of the function $g(x) = \mathbb{P}(T_x < \infty)$.

- Does there exist a rotor configuration on $\mathbb{Z}^3$ which causes every chip to return to the origin in finitely many steps?
  - Known to exist for $\mathbb{Z}^2$ (Jim Propp) and for the $d$-regular tree.

- Does there exist a rotor configuration on $\mathbb{Z}^2$ (or any recurrent graph) which causes infinitely many chips to escape to infinity?
  - Can show that the proportion that escape is zero.

Lionel Levine
The Sandpile Group of a Tree