Chip-Firing and Rotor-Routing on Trees

Lionel Levine

University of California, Berkeley

October 8, 2007

Joint work with Itamar Landau and Yuval Peres.
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.

- 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.
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 rotar 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

1. 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

1. 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:
  2. Modify the dynamics: Divisible Sandpile
Three Approaches to Circularity

1. 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:

2. Modify the dynamics: Divisible Sandpile
3. Modify the underlying graph.
   ▶ The tree is easier than the lattice.
Spherical Asymptotics

**Theorem** (L.-Peres) Let $A_n$ be the region of $n$ particles formed by rotor-router aggregation in $\mathbb{Z}^d$. 

$B_r - c \log r \subset A_n \subset B_r(1 + c' r^{-1/d} \log r)$, where $B_\rho$ is the ball of radius $\rho$ centered at the origin. $n = \omega d^d r^d$, where $\omega d$ is the volume of the unit ball in $\mathbb{R}^d$. $c, c'$ depend only on $d$.

**Corollary**: Inradius/Outradius $\to 1$ as $n \to \infty$. 

Lionel Levine

Chip-Firing and Rotor-Routing on Trees
Spherical Asymptotics

**Theorem** (L.-Peres) Let $A_n$ be the region of $n$ particles formed by rotor-router aggregation in $\mathbb{Z}^d$. Then

$$B_{r-c \log r} \subset A_n \subset B_r(1+c'r^{-1/d \log r})$$

where

- $B_\rho$ is the ball of radius $\rho$ centered at the origin.
Spherical Asymptotics

Theorem (L.-Peres) Let $A_n$ be the region of $n$ particles formed by rotor-router aggregation in $\mathbb{Z}^d$. Then

$$B_{r-c \log r} \subset A_n \subset B_r(1+c'r^{-1/d \log r}),$$

where

- $B_\rho$ is the ball of radius $\rho$ centered at the origin.
- $n = \omega_d r^d$, where $\omega_d$ is the volume of the unit ball in $\mathbb{R}^d$. 
Spherical Asymptotics

Theorem (L.-Peres) Let $A_n$ be the region of $n$ particles formed by rotor-router aggregation in $\mathbb{Z}^d$. Then

$$B_{r-c\log r} \subset A_n \subset B_r(1+c'r^{-1/d}\log r),$$

where

- $B_\rho$ is the ball of radius $\rho$ centered at the origin.
- $n = \omega_d r^d$, where $\omega_d$ is the volume of the unit ball in $\mathbb{R}^d$.
- $c, c'$ depend only on $d$.

Corollary: Inradius/Outradius $\to 1$ as $n \to \infty$. 

Lionel Levine
Chip-Firing and Rotor-Routing on Trees
Spherical Asymptotics

- **Theorem** (L.-Peres) Let $A_n$ be the region of $n$ particles formed by rotor-router aggregation in $\mathbb{Z}^d$. Then

$$B_{r-c \log r} \subset A_n \subset B_r(1+c' r^{-1/d} \log r),$$

where
- $B_\rho$ is the ball of radius $\rho$ centered at the origin.
- $n = \omega_d r^d$, where $\omega_d$ is the volume of the unit ball in $\mathbb{R}^d$.
- $c, c'$ depend only on $d$.

- **Corollary**: Inradius/Outradius $\to 1$ as $n \to \infty$. 
Perfect Circularity 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

$$A_{b_n} = B_n$$

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

- Choices of which particles to route in what order don’t affect the final shape generated or the final rotor directions.
The Abelian Property

- Choices of which particles to route in what order don’t affect the final shape generated or the final rotor directions.
- **Chip-firing** or **abelian sandpile** model:
The Abelian Property

- Choices of which particles to route in what order don’t affect the final shape generated or the final rotor directions.

- **Chip-firing** or **abelian sandpile** model:
  - When 4 or more grains of sand accumulate at a site in $\mathbb{Z}^2$, it topples, sending one grain to each neighbor.
The Abelian Property

- Choices of which particles to route in what order don’t affect the final shape generated or the final rotor directions.

- **Chip-firing** or **abelian sandpile** model:
  - When 4 or more grains of sand accumulate at a site in $\mathbb{Z}^2$, it topples, sending one grain to each neighbor.
  - Choices of which sites to topple in what order don’t affect the final sandpile shape.
The Abelian Property

- Choices of which particles to route in what order don’t affect the final shape generated or the final rotor directions.

- **Chip-firing** or **abelian sandpile** model:
  - When 4 or more grains of sand accumulate at a site in \( \mathbb{Z}^2 \), it topples, sending one grain to each neighbor.
  - Choices of which sites to topple in what order don’t affect the final sandpile shape.

- **Equivalent models**:
  - Start with \( n \) particles at the origin.
The Abelian Property

- Choices of which particles to route in what order don’t affect the final shape generated or the final rotor directions.

- **Chip-firing** or **abelian sandpile** model:
  - When 4 or more grains of sand accumulate at a site in $\mathbb{Z}^2$, it **topples**, sending one grain to each neighbor.
  - Choices of which sites to topple in what order don’t affect the final sandpile shape.

- **Equivalent models**:
  - Start with $n$ particles at the origin.
  - If there are $m$ particles at a site, send $\lfloor \frac{m-1}{4} \rfloor$ to each neighbor.
The Abelian Property

- Choices of which particles to route in what order don’t affect the final shape generated or the final rotor directions.

- **Chip-firing** or **abelian sandpile** model:
  - When 4 or more grains of sand accumulate at a site in $\mathbb{Z}^2$, it **topples**, sending one grain to each neighbor.
  - Choices of which sites to topple in what order don’t affect the final sandpile shape.

- **Equivalent models**:
  - Start with $n$ particles at the origin.
  - If there are $m$ particles at a site, send $\lfloor \frac{m-1}{4} \rfloor$ to each neighbor.
  - **Sandpile**: Leave the extra particles where they are.
The Abelian Property

▶ Choices of which particles to route in what order don’t affect the final shape generated or the final rotor directions.

▶ **Chip-firing** or **abelian sandpile** model:
  ▶ When 4 or more grains of sand accumulate at a site in $\mathbb{Z}^2$, it topples, sending one grain to each neighbor.
  ▶ Choices of which sites to topple in what order don’t affect the final sandpile shape.

▶ **Equivalent models**:
  ▶ Start with $n$ particles at the origin.
  ▶ If there are $m$ particles at a site, send $\lfloor \frac{m-1}{4} \rfloor$ to each neighbor.
  ▶ **Sandpile**: Leave the extra particles where they are.
  ▶ **Rotor**: Send extra particles according to the usual rotor rule.
Bounds for the Abelian Sandpile

Theorem (L.-Peres) Let $S_n$ be the set of sites visited by the abelian sandpile in $\mathbb{Z}^d$, starting from $n$ particles at the origin.
Bounds for the Abelian Sandpile

**Theorem** (L.-Peres) Let $S_n$ be the set of sites visited by the abelian sandpile in $\mathbb{Z}^d$, starting from $n$ particles at the origin. Then

$$\left(\text{Ball of volume } \frac{n - o(n)}{2d - 1}\right) \subset S_n \subset \left(\text{Ball of volume } \frac{n + o(n)}{d}\right).$$
Bounds for the Abelian Sandpile

- **Theorem** (L.-Peres) Let $S_n$ be the set of sites visited by the abelian sandpile in $\mathbb{Z}^d$, starting from $n$ particles at the origin. Then

  \[
  \left( \text{Ball of volume } \frac{n - o(n)}{2d - 1} \right) \subset S_n \subset \left( \text{Ball of volume } \frac{n + o(n)}{d} \right).
  \]

- Improves the bounds of Le Borgne and Rossin.
(Disk of area $n/3$) $\subset S_n \subset$ (Disk of area $n/2$)
Chip-Firing on Graphs

- Finite connected graph $G$ with a distinguished vertex $s$ called the sink.
- **Chip configuration**: Each site $v \neq s$ has $\sigma(v) \geq 0$ chips.
Chip-Firing on Graphs

- Finite connected graph $G$ with a distinguished vertex $s$ called the **sink**.
- **Chip configuration**: Each site $v \neq s$ has $\sigma(v) \geq 0$ chips.
- If $\sigma(v) \geq \deg(v)$, the vertex $v$ can **topple**, sending one chip to each neighbor.
Chip-Firing on Graphs

- Finite connected graph $G$ with a distinguished vertex $s$ called the **sink**.
- **Chip configuration**: Each site $v \neq s$ has $\sigma(v) \geq 0$ chips.
- If $\sigma(v) \geq \deg(v)$, the vertex $v$ can **topple**, sending one chip to each neighbor.
- The sink never topples.
- Order of topplings does not affect the final state $\sigma^\circ$. 
The Sandpile Group of a Graph

- A chip configuration $\sigma$ on $G$ is **stable** if
  \[ \sigma(v) \leq \text{deg}(v) - 1 \]
  for all vertices $v$. 

- Stable configurations form a finite commutative monoid $M = M(G)$ under the operation $(\sigma, \tau) \mapsto (\sigma + \tau) \circ$. 

- Babai-Toumpakari: The minimal ideal of $M$ is a finite abelian group $\text{SP}(G)$ called the sandpile group of $G$. 

Lionel Levine
Chip-Firing and Rotor-Routing on Trees
The Sandpile Group of a Graph

▶ A chip configuration $\sigma$ on $G$ is stable if

$$\sigma(v) \leq \deg(v) - 1$$

for all vertices $v$.

▶ Stable configurations form a finite commutative monoid $M = M(G)$ under the operation

$$(\sigma, \tau) \mapsto (\sigma + \tau)^\circ.$$
The Sandpile Group of a Graph

- A chip configuration $\sigma$ on $G$ is **stable** if
  \[ \sigma(v) \leq \deg(v) - 1 \]
  for all vertices $v$.
- Stable configurations form a finite **commutative monoid** $\mathcal{M} = \mathcal{M}(G)$ under the operation
  \[ (\sigma, \tau) \mapsto (\sigma + \tau)^\circ. \]
- Babai-Toumpakari: The minimal ideal of $\mathcal{M}$ is a finite abelian group $SP(G)$ called the **sandpile group** of $G$. 
Two Other Constructions of the Sandpile Group

- \( SP(G) \simeq \mathbb{Z}^{n-1}/\Delta \mathbb{Z}^{n-1} \), where

\[ \Delta = D - A \]

is the \textbf{reduced Laplacian} of \( G \).
Two Other Constructions of the Sandpile Group

- $SP(G) \simeq \mathbb{Z}^{n-1}/\Delta \mathbb{Z}^{n-1}$, where

\[ \Delta = D - A \]

is the reduced Laplacian of $G$.

- **Matrix-tree theorem:**

\[ \#SP(G) = \det \Delta = \#\{\text{spanning trees of } G\}. \]
Two Other Constructions of the Sandpile Group

- \( SP(G) \cong \mathbb{Z}^{n-1}/\Delta \mathbb{Z}^{n-1} \), where
  \[
  \Delta = D - A
  \]
  is the reduced Laplacian of \( G \).

- **Matrix-tree theorem:**
  \[
  \#SP(G) = \det \Delta = \#\{\text{spanning trees of } G\}.
  \]

- \( SP(G) \) is the group of **recurrent chip configurations** on \( G \),
  i.e. configurations \( \sigma \) such that
  \[
  \sigma = (\sigma + \tau)^\circ
  \]
  for some configuration \( \tau \neq 0 \).
The Burning Algorithm

Consider the chip configuration

$$\beta(v) = \# \text{ of edges from } v \text{ to the sink.}$$
The Burning Algorithm

▶ Consider the chip configuration

\[ \beta(v) = \# \text{ of edges from } v \text{ to the sink}. \]

▶ **Lemma** (Dhar): \( \sigma \) is recurrent if and only if

\[ \sigma = (\sigma + \beta)^\circ. \]
The Burning Algorithm

Consider the chip configuration

\[ \beta(v) = \# \text{ of edges from } v \text{ to the sink}. \]

Lemma (Dhar): \( \sigma \) is recurrent if and only if

\[ \sigma = (\sigma + \beta)\circ. \]

Moreover, if \( \sigma \) is recurrent, then every vertex topples exactly once in reducing \( \sigma + \beta \) to \( \sigma \).
The Burning Algorithm

Consider the chip configuration

\[ \beta(v) = \# \text{ of edges from } v \text{ to the sink}. \]

Lemma (Dhar): \( \sigma \) is recurrent if and only if

\[ \sigma = (\sigma + \beta)\. \]

Moreover, if \( \sigma \) is recurrent, then every vertex topples exactly once in reducing \( \sigma + \beta \) to \( \sigma \).

Proof:

\[ \beta = \sum_{v \neq s} \Delta_v. \]
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

\[ x \in T \text{ is critical for a chip configuration } u \text{ if } x \neq s \text{ and } \]

\[ u(x) \leq \# \text{ of critical children of } x. \quad (1) \]
Critical vertices

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

- 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

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

  - 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 $x$. 
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.
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.

- **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

$$SP(T_n) \simeq \mathbb{Z}_{2^{n-1}} \oplus \mathbb{Z}_{2^{n-1}} \oplus \cdots \oplus (\mathbb{Z}_7)^{2^{n-4}} \oplus (\mathbb{Z}_3)^{2^{n-3}}.$$
Structure of the Sandpile Group

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

\[ SP(T_n) \simeq \mathbb{Z}_{2^{n-1}} \oplus \mathbb{Z}_{2^{n-2}} \oplus \cdots \oplus (\mathbb{Z}_7)^{2^{n-4}} \oplus (\mathbb{Z}_3)^{2^{n-3}}. \]

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

Lionel Levine  Chip-Firing and Rotor-Routing on Trees
Subgroup Generated by the Root

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

- What can we say about the subgroup of $\text{SP}(T_n)$ 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 $T_4$:
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 $\hat{r} = \delta_r + e$?

Multiples of the root in $T_4$: 

Lionel Levine
Chip-Firing and Rotor-Routing on Trees
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>0</td>
<td>1</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>
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

$$n - 1 \sum_{k=0}^{\infty} 2^k = 2n - 1.$$
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.
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 $T_1, \ldots, T_k$. 
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

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

where \( r, r_i \) are the roots of \( T, T_i \) respectively.

Proof sketch: Map \( (u_1, \ldots, u_k) \mapsto (u_1, \ldots, u_k) \).

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

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

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

Proof sketch: Map $\begin{pmatrix} a \\ u_1, \ldots, u_k \end{pmatrix} \mapsto (u_1, \ldots, u_k)$. 
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

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

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

Proof sketch: Map $\begin{pmatrix} a \\ u_1, \ldots, u_k \end{pmatrix} \mapsto (u_1, \ldots, u_k)$.

- After modding out by $\hat{r}$, the branches become independent.
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

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

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 $p : SP(T_n) \rightarrow (\hat{r})$. 
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

$$p(u)(x) = 2^{n+1} - |x| \sum_{|y|=|x|} u(y).$$
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 $p : SP(T_n) \to (\hat{r})$.

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

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

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

- $SP(T_4) = \mathbb{Z}_{15} \oplus SP(T_3)^2 / \mathbb{Z}_7 = \mathbb{Z}_{15} \oplus \mathbb{Z}_7 \oplus \mathbb{Z}_2^3$.

- $SP(T_5) = \mathbb{Z}_{31} \oplus SP(T_4)^2 / \mathbb{Z}_{15} = \mathbb{Z}_{31} \oplus \mathbb{Z}_{15} \oplus \mathbb{Z}_2^7 \oplus \mathbb{Z}_4^3$.

- $\ldots$

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

Lionel Levine  Chip-Firing and Rotor-Routing on Trees
Factoring Into Cyclic Subgroups

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

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

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

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

... 

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

\[
\left(\frac{2c3}{3}\right)^{2^n} = (0.858)^{2^n}
\]
Entropy

Number of recurrent states:

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

where

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

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

\[ \approx 1.364. \]
Entropy

- Number of recurrent states:

\[ \log_2 |SP(T_n)| \sim c 2^n \]

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 the root. Then for $r \leq n$

\[
P(R \geq r) \asymp 2^{-r}.
\]
“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 the root. Then for \( r \leq n \) and \( m, t \leq 2^n \)

\[
\mathbb{P}(R \geq r) \asymp 2^{-r}.
\]
\[
\mathbb{P}(M \geq m) \asymp 1/m.
\]
\[
\mathbb{P}(\tau \geq t) \asymp 1/t.
\]
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.
  \[ \implies \text{recurrent states have no oriented cycles}. \]
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: $RR(G) \cong SP(G)$. 

Lionel Levine  

Chip-Firing and Rotor-Routing on Trees
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 $x$. 

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

Lemma: If $H$ is harmonic, and the initial and final rotor configurations are the same, then

$$\sum_{x \in T} H(x) u(x) = \sum_{x \in T} H(x) v(x).$$
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 $v$. 
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 $v$.

**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 Circularity on the Tree

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

\[ H(x) = \mathbb{P}_x(X_\tau = z) \]
Proof of Circularity on the Tree

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 Circularity on the Tree

- 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 Circularity on the Tree

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 Circularity on the Tree

- 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.
  - Get a new aggregation process \( A'_m = A_{f(m)} \).
Proof of Circularity on the Tree

- 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.
  - 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 Circularity on the Tree

- Induct on $n$ to show $A'_{c_n} = B_n$.
Proof of Circularity on the Tree

- 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.
Proof of Circularity on the Tree

- Induct on $n$ to show $A'_c = B_n$.
- 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 Circularity on the Tree

- 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 Circularity on the Tree

- 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 \( a^{(j)} = a_1a_{j+3}a_{j+6} \ldots \).

- **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.
- ...  
- Every subword of \( a^{(j)} \) of length \( 2^k - 1 \) contains at most \( 2^{k-1} \) ones.

Lionel Levine  
Chip-Firing and Rotor-Routing on Trees
Open Problems

Find a bijective proof that

$$\#SP(T_n) = 3^{2n-3} \cdot 7^{2n-4} \cdot \cdots \cdot (2^{n-1} - 1)(2^n - 1).$$
Open Problems

- Find a bijective proof that
  \[ \#SP(T_n) = 3^{2^{n-3}} \cdot 7^{2^{n-4}} \cdots (2^{n-1} - 1)(2^n - 1). \]

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

- Find a bijective proof that
  \[ \#SP(T_n) = 3^{2n-3} 7^{2n-4} \cdots (2^{n-1} - 1)(2^n - 1). \]

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

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

▷ Find a bijective proof that

\[ \#SP(T_n) = 3^{2^{n-3}} 7^{2^{n-4}} \cdots (2^{n-1} - 1)(2^n - 1). \]

▷ 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?
Open Problems

- Find a bijective proof that
  \[ \#SP(T_n) = 3^{2n-3} 7^{2n-4} \cdots (2^{n-1} - 1)(2^n - 1). \]

- 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.
Open Problems: Chip-Firing

- Fix an integer $h \in (-\infty, 2]$.
- Start every site in $\mathbb{Z}^2$ at height $h$.

Question: As $n \to \infty$, is the limiting shape $S_n, h$ a regular $(12 - 4h)$-gon?

Fey and Redig (2007) Case $h = 2$: The limiting shape of $S_n, 2$ is a square.

In all other cases, even the existence of a limiting shape is open.

Even for $h = 2$, the rate of growth of the square is not known.
Open Problems: Chip-Firing

- Fix an integer $h \in (-\infty, 2]$.
- Start every site in $\mathbb{Z}^2$ at height $h$.
- Let $S_{n,h}$ be the set of visited sites for the abelian sandpile started with $n$ particles at the origin.

Question: As $n \to \infty$, is the limiting shape $S_{n,h}$ a regular $(12 - 4h)$-gon?

- Fey and Redig (2007) Case $h = 2$: The limiting shape of $S_{n,2}$ is a square.
- In all other cases, even the existence of a limiting shape is open.
- Even for $h = 2$, the rate of growth of the square is not known.
Open Problems: Chip-Firing

- Fix an integer $h \in (-\infty, 2]$.
- Start every site in $\mathbb{Z}^2$ at height $h$.
- Let $S_{n,h}$ be the set of visited sites for the abelian sandpile started with $n$ particles at the origin.

**Question**: As $n \to \infty$, is the limiting shape $S_{n,h}$ a regular $(12 - 4h)$-gon?

Fey and Redig (2007) Case $h = 2$: The limiting shape of $S_{n,2}$ is a square.

In all other cases, even the existence of a limiting shape is open.

Even for $h = 2$, the rate of growth of the square is not known.
Open Problems: Chip-Firing

- Fix an integer $h \in (-\infty, 2]$.
- Start every site in $\mathbb{Z}^2$ at height $h$.
- Let $S_{n,h}$ be the set of visited sites for the abelian sandpile started with $n$ particles at the origin.

**Question**: As $n \to \infty$, is the limiting shape $S_{n,h}$ a regular $(12 - 4h)$-gon?

- Fey and Redig (2007) Case $h = 2$: The limiting shape of $S_{n,2}$ is a square.
Open Problems: Chip-Firing

Fix an integer \( h \in (-\infty, 2] \).

Start every site in \( \mathbb{Z}^2 \) at height \( h \).

Let \( S_{n,h} \) be the set of visited sites for the abelian sandpile started with \( n \) particles at the origin.

**Question:** As \( n \to \infty \), is the limiting shape \( S_{n,h} \) a regular \((12 - 4h)\)-gon?

Fey and Redig (2007) Case \( h = 2 \): The limiting shape of \( S_{n,2} \) is a square.

In all other cases, even the existence of a limiting shape is open.
Open Problems: Chip-Firing

- Fix an integer \( h \in (-\infty, 2] \).
- Start every site in \( \mathbb{Z}^2 \) at height \( h \).
- Let \( S_{n,h} \) be the set of visited sites for the abelian sandpile started with \( n \) particles at the origin.
- **Question**: As \( n \to \infty \), is the limiting shape \( S_{n,h} \) a regular \((12 - 4h)\)-gon?
- Fey and Redig (2007) Case \( h = 2 \): The limiting shape of \( S_{n,2} \) is a square.
- In all other cases, even the existence of a limiting shape is open.
- Even for \( h = 2 \), the rate of growth of the square is not known.
$h = 2$

$h = 1$

$h = 0$