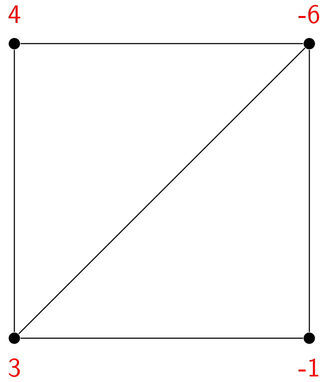
Chip-Firing on Representable Matroids

GOCC

Alex McDonough

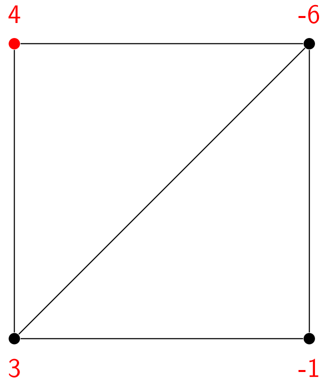
5/6/20

Let's Fire Some Chips!



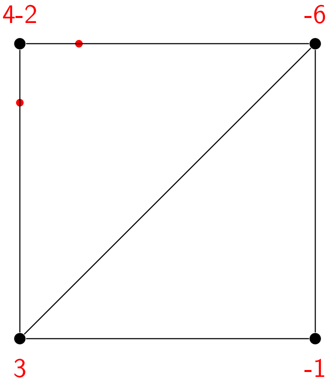
We call an assignment of an integer to each vertex of a graph a *chip configuration.*

Let's Fire Some Chips!



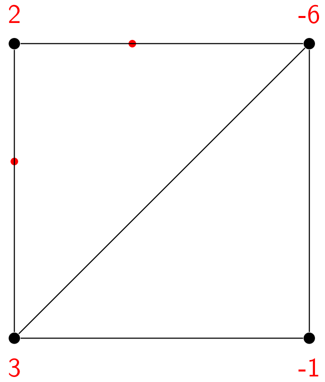
We call an assignment of an integer to each vertex of a graph a *chip configuration.*

Let's Fire Some Chips!



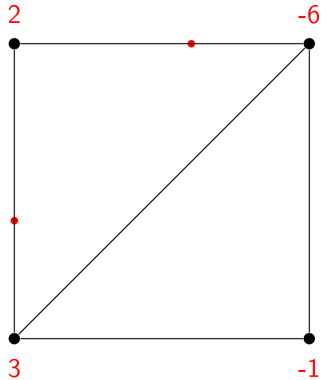
We call an assignment of an integer to each vertex of a graph a *chip configuration.*

Let's Fire Some Chips!



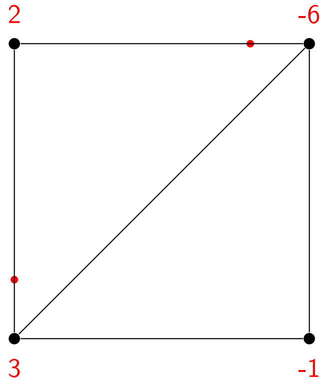
We call an assignment of an integer to each vertex of a graph a *chip configuration.*

Let's Fire Some Chips!

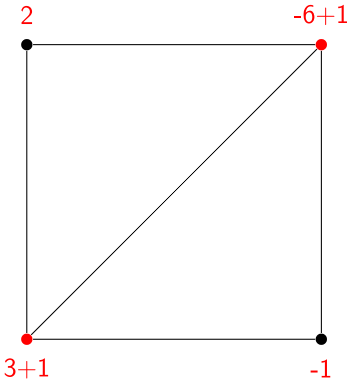


We call an assignment of an integer to each vertex of a graph a *chip configuration.*

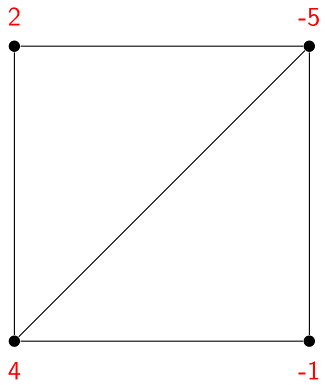
Let's Fire Some Chips!



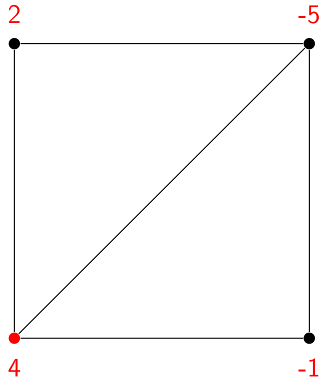
Let's Fire Some Chips!



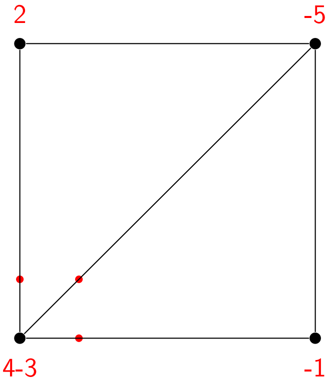
Let's Fire Some Chips!



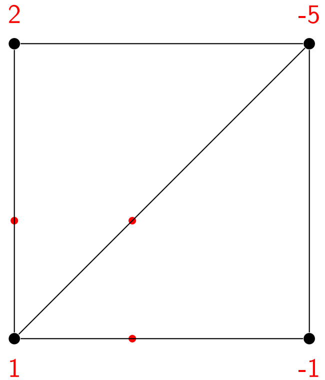
Let's Fire Some Chips!



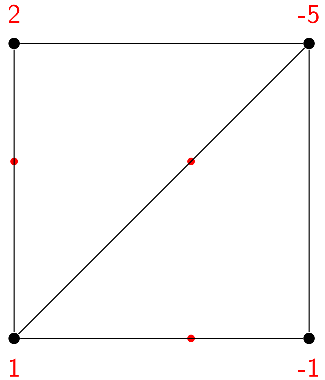
Let's Fire Some Chips!



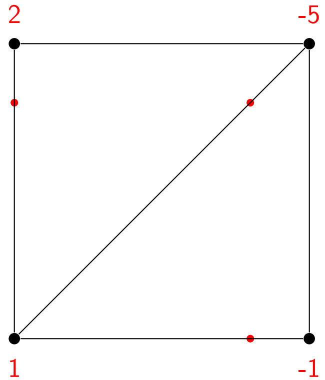
Let's Fire Some Chips!



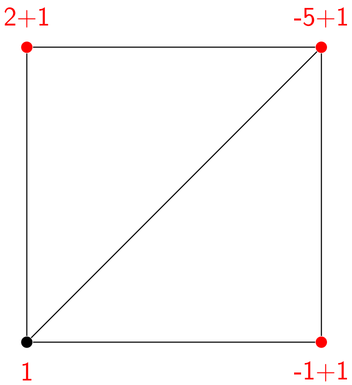
Let's Fire Some Chips!



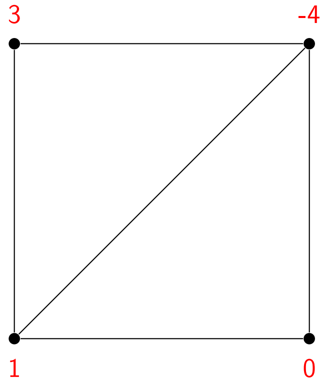
Let's Fire Some Chips!



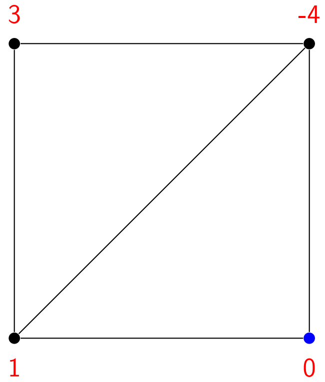
Let's Fire Some Chips!



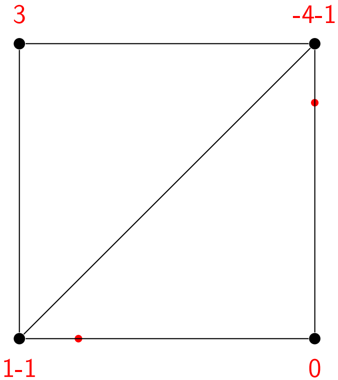
Let's Fire Some Chips!



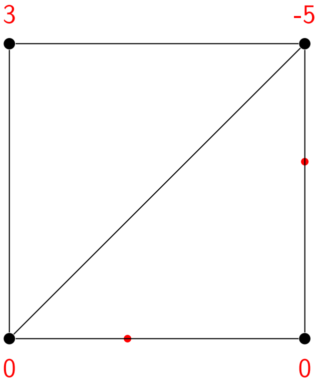
Let's Fire Some Chips!



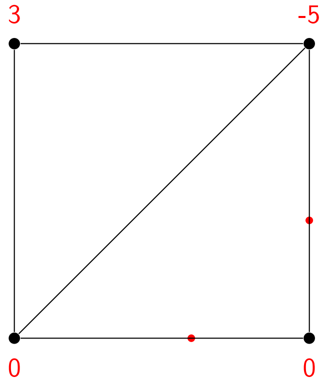
Let's Fire Some Chips!



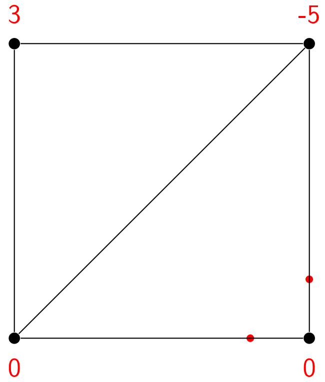
Let's Fire Some Chips!



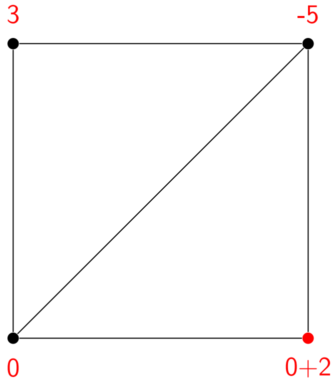
Let's Fire Some Chips!



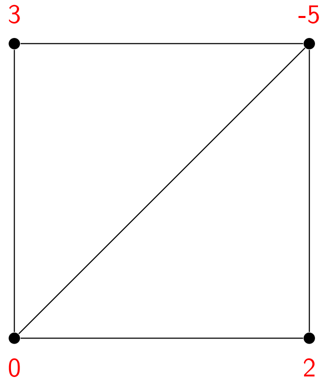
Let's Fire Some Chips!



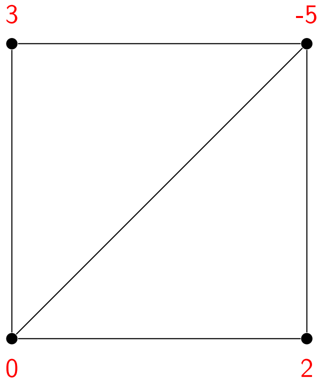
Let's Fire Some Chips!



Let's Fire Some Chips!

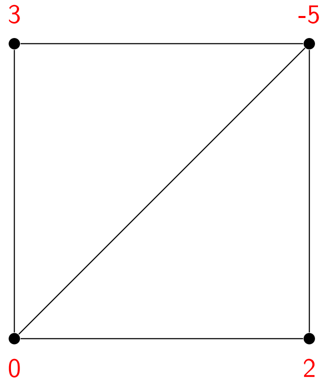


Let's Fire Some Chips!



Does the order of firings matter?

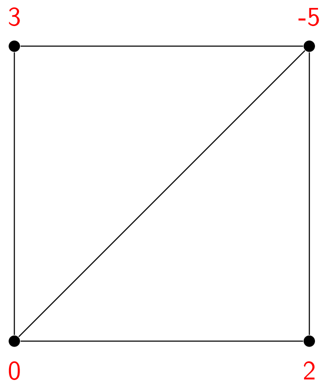
Let's Fire Some Chips!



Does the order of firings matter?

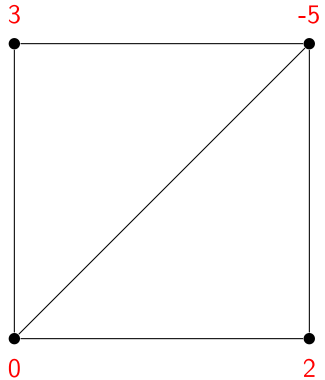
No. For this reason, this is often called the "Abelian Sandpile Model".

Let's Fire Some Chips!



We say chip configurations c_1 and c_2 are *firing equivalent* if we can reach c_2 from c_1 by firing moves.

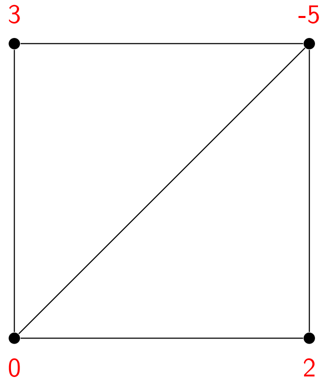
Let's Fire Some Chips!



We say chip configurations c_1 and c_2 are *firing equivalent* if we can reach c_2 from c_1 by firing moves.

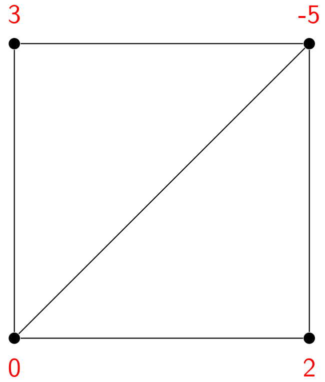
It's not hard to show that firing equivalence is an equivalence relation.

Let's Fire Some Chips!



Are there any chip-configurations that we definitely can't get to?

Let's Fire Some Chips!



Are there any chip-configurations that we definitely can't get to?

We can't change the total number of chips.

The Sandpile Group on Graphs

- Call chip configurations with 0 total chips *cyclic* chip configurations.

The Sandpile Group on Graphs

- Call chip configurations with 0 total chips *cyclic* chip configurations.

Definition (Sandpile Group)

The *sandpile group* of a graph G, denoted $\mathcal{S}(G)$, is the group of cyclic chip configurations under pointwise addition modulo the firing equivalence relation.

The Sandpile Group on Graphs

- Call chip configurations with 0 total chips *cyclic* chip configurations.

Definition (Sandpile Group)

The *sandpile group* of a graph G, denoted $\mathcal{S}(G)$, is the group of cyclic chip configurations under pointwise addition modulo the firing equivalence relation.

- The sandpile group of G is also called the critical group, $\text{Pic}^0(G)$, or $\text{Jac}(G)$.

The Sandpile Group on Graphs

- Call chip configurations with 0 total chips *cyclic* chip configurations.

Definition (Sandpile Group)

The *sandpile group* of a graph G, denoted $\mathcal{S}(G)$, is the group of cyclic chip configurations under pointwise addition modulo the firing equivalence relation.

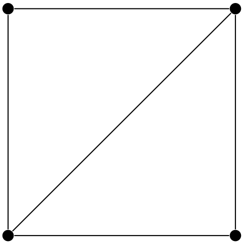
- The sandpile group of G is also called the critical group, $\mathrm{Pic}^0(G)$, or $\mathrm{Jac}(G)$.

Theorem (Sandpile Matrix-Tree Theorem)

The size of $\mathcal{S}(G)$ is the number of spanning trees of G.

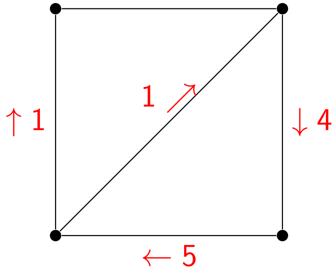
Edge-labeled Chip-Firing

- Instead of working with chips let's imagine some Frito-Lay transportation trucks driving from vertex to vertex.



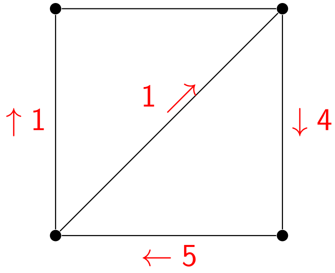
Edge-labeled Chip-Firing

- Instead of working with chips let's imagine some Frito-Lay transportation trucks driving from vertex to vertex.



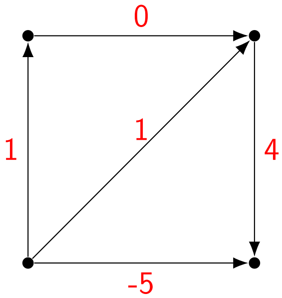
Edge-labeled Chip-Firing

- Instead of working with chips let's imagine some Frito-Lay transportation trucks driving from vertex to vertex.
- If we choose an *orientation* for our graph, we can get rid of arrows on our chip trucks.



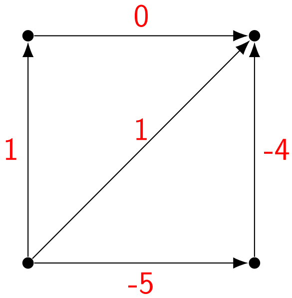
Edge-labeled Chip-Firing

- Instead of working with chips let's imagine some Frito-Lay transportation trucks driving from vertex to vertex.
- If we choose an *orientation* for our graph, we can get rid of arrows on our chip trucks.



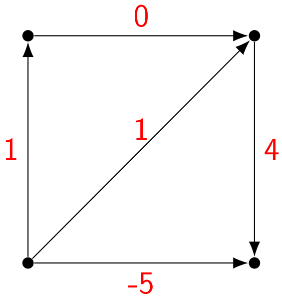
Edge-labeled Chip-Firing

- Instead of working with chips let's imagine some Frito-Lay transportation trucks driving from vertex to vertex.
- If we choose an *orientation* for our graph, we can get rid of arrows on our chip trucks.
- Note that the this is not a directed graph: the orientation is just for bookkeeping.



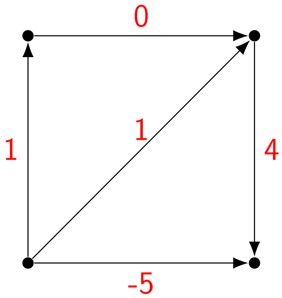
Edge-labeled Chip-Firing

- Instead of working with chips let's imagine some Frito-Lay transportation trucks driving from vertex to vertex.
- If we choose an *orientation* for our graph, we can get rid of arrows on our chip trucks.
- Note that the this is not a directed graph: the orientation is just for bookkeeping.



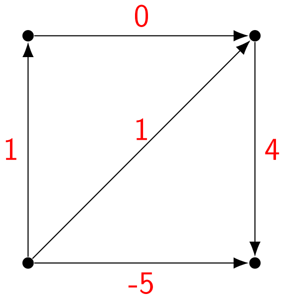
Edge-labeled Chip-Firing

- We call an assignment of an integer to each edge of an oriented graph an *edge configuration*.



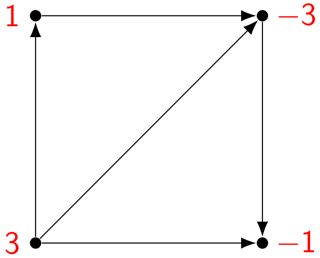
Edge-labeled Chip-Firing

- We call an assignment of an integer to each edge of an oriented graph an *edge configuration*.
- We can get a chip configuration from an edge configuration by having each truck take one chip from its starting vertex and deliver it to the ending vertex.



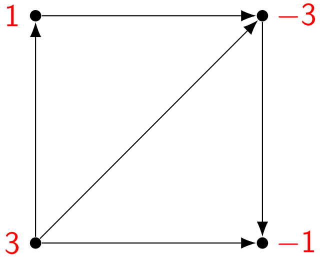
Edge-labeled Chip-Firing

- We call an assignment of an integer to each edge of an oriented graph an *edge configuration*.
- We can get a chip configuration from an edge configuration by having each truck take one chip from its starting vertex and deliver it to the ending vertex.



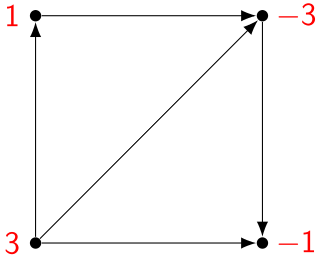
Edge-labeled Chip-Firing

- We call an assignment of an integer to each edge of an oriented graph an *edge configuration*.
- We can get a chip configuration from an edge configuration by having each truck take one chip from its starting vertex and deliver it to the ending vertex.
- We call this map from edge configurations to chip configurations ∂.



Edge-labeled Chip-Firing

- We call an assignment of an integer to each edge of an oriented graph an *edge configuration*.
- We can get a chip configuration from an edge configuration by having each truck take one chip from its starting vertex and deliver it to the ending vertex.
- We call this map from edge configurations to chip configurations ∂.
- Note that ∂ always maps to cyclic chip configurations.



Edge-labeled Chip-Firing

- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $\partial(c_1)$ and $\partial(c_2)$ are firing equivalent.

Edge-labeled Chip-Firing

- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $\partial(c_1)$ and $\partial(c_2)$ are firing equivalent.

Definition (Edge-labeled Sandpile Group)

The *edge-labeled sandpile group* of a graph G, denoted $\mathcal{S}^e(G)$, is the group of edge configurations under pointwise addition modulo the edge firing equivalence relation.

Edge-labeled Chip-Firing

- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $\partial(c_1)$ and $\partial(c_2)$ are firing equivalent.

Definition (Edge-labeled Sandpile Group)

The *edge-labeled sandpile group* of a graph G, denoted $\mathcal{S}^e(G)$, is the group of edge configurations under pointwise addition modulo the edge firing equivalence relation.

Definition (Sandpile Group)

The *sandpile group* of a graph G, denoted $\mathcal{S}(G)$, is the group of cyclic chip configurations under pointwise addition modulo the firing equivalence relation.

Edge-labeled Chip-Firing

- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $\partial(c_1)$ and $\partial(c_2)$ are firing equivalent.

Definition (Edge-labeled Sandpile Group)

The *edge-labeled sandpile group* of a graph G, denoted $\mathcal{S}^e(G)$, is the group of edge configurations under pointwise addition modulo the edge firing equivalence relation.

Definition (Sandpile Group)

The *sandpile group* of a graph G, denoted $\mathcal{S}(G)$, is the group of cyclic chip configurations under pointwise addition modulo the firing equivalence relation.

Topology Time!!!

Topology Time!!!!!

- Chip configurations are also called *0-chains*. The group of 0-chains under pointwise addition is denoted C_0.

Topology Time!!!!!

- Chip configurations are also called *0-chains*. The group of 0-chains under pointwise addition is denoted C_0.
- Edge configurations are also called *1-chains*. The group of 1-chains under pointwise addition is denoted C_1.

Topology Time!!!!!

- Chip configurations are also called *0-chains*. The group of 0-chains under pointwise addition is denoted C_0.
- Edge configurations are also called *1-chains*. The group of 1-chains under pointwise addition is denoted C_1.
- Let ϵ be the map $C_0 \to \mathbb{Z}$ that outputs the total number of chips.

Topology Time!!!!!

- Chip configurations are also called *0-chains*. The group of 0-chains under pointwise addition is denoted C_0.
- Edge configurations are also called *1-chains*. The group of 1-chains under pointwise addition is denoted C_1.
- Let ϵ be the map $C_0 \to \mathbb{Z}$ that outputs the total number of chips.
- Then, $\ker \epsilon$ is the group of cyclic chip configurations.

Topology Time!!!!!

- Chip configurations are also called *0-chains*. The group of 0-chains under pointwise addition is denoted C_0.
- Edge configurations are also called *1-chains*. The group of 1-chains under pointwise addition is denoted C_1.
- Let ϵ be the map $C_0 \to \mathbb{Z}$ that outputs the total number of chips.
- Then, $\ker \epsilon$ is the group of cyclic chip configurations.
- Recall that ∂ is a map from $C_1 \to C_0$.

Topology Time!!!!!

- Chip configurations are also called *0-chains*. The group of 0-chains under pointwise addition is denoted C_0.
- Edge configurations are also called *1-chains*. The group of 1-chains under pointwise addition is denoted C_1.
- Let ϵ be the map $C_0 \to \mathbb{Z}$ that outputs the total number of chips.
- Then, $\ker \epsilon$ is the group of cyclic chip configurations.
- Recall that ∂ is a map from $C_1 \to C_0$.

$$C_1 \xrightarrow{\partial} C_0 \xrightarrow{\epsilon} \mathbb{Z}$$

Topology Time!!!!!

- Chip configurations are also called *0-chains*. The group of 0-chains under pointwise addition is denoted C_0.
- Edge configurations are also called *1-chains*. The group of 1-chains under pointwise addition is denoted C_1.
- Let ϵ be the map $C_0 \to \mathbb{Z}$ that outputs the total number of chips.
- Then, $\ker \epsilon$ is the group of cyclic chip configurations.
- Recall that ∂ is a map from $C_1 \to C_0$.

$$C_1 \xrightarrow{\partial} C_0 \xrightarrow{\epsilon} \mathbb{Z}$$

- We observed that $\operatorname{im} \partial \subseteq \ker \epsilon$.

Topology Time!!!!!

- Chip configurations are also called *0-chains*. The group of 0-chains under pointwise addition is denoted C_0.
- Edge configurations are also called *1-chains*. The group of 1-chains under pointwise addition is denoted C_1.
- Let ϵ be the map $C_0 \to \mathbb{Z}$ that outputs the total number of chips.
- Then, $\ker \epsilon$ is the group of cyclic chip configurations.
- Recall that ∂ is a map from $C_1 \to C_0$.

$$C_1 \xrightarrow{\partial} C_0 \xrightarrow{\epsilon} \mathbb{Z}$$

- We observed that $\operatorname{im} \partial \subseteq \ker \epsilon$.

Definition (Reduced Homology)

The group $\ker \epsilon / \operatorname{im} \partial$ is called the reduced 0^{th} homology group and is denoted $\tilde{H}_0(G)$.

The Two Sandpile Groups

- How are $\mathcal{S}(G)$ and $\mathcal{S}^e(G)$ related?

The Two Sandpile Groups

- How are $\mathcal{S}(G)$ and $\mathcal{S}^e(G)$ related?
- We can think of these groups as $\ker \epsilon$ and $\operatorname{im} \partial$ respectively with the same equivalence relation.

The Two Sandpile Groups

- How are $\mathcal{S}(G)$ and $\mathcal{S}^e(G)$ related?
- We can think of these groups as $\ker \epsilon$ and $\operatorname{im} \partial$ respectively with the same equivalence relation.
- $\mathcal{S}(G) \cong \mathcal{S}^e(G)$ if and only if $\tilde{H}_0(G) \cong 0$.

The Two Sandpile Groups

- How are $\mathcal{S}(G)$ and $\mathcal{S}^e(G)$ related?
- We can think of these groups as $\ker \epsilon$ and $\operatorname{im} \partial$ respectively with the same equivalence relation.
- $\mathcal{S}(G) \cong \mathcal{S}^e(G)$ if and only if $\tilde{H}_0(G) \cong 0$.
- $\tilde{H}_0(G) \cong \mathbb{Z}^{k-1}$ where k is the number of connected components of G.

The Two Sandpile Groups

- How are $\mathcal{S}(G)$ and $\mathcal{S}^e(G)$ related?
- We can think of these groups as $\ker \epsilon$ and $\operatorname{im} \partial$ respectively with the same equivalence relation.
- $\mathcal{S}(G) \cong \mathcal{S}^e(G)$ if and only if $\tilde{H}_0(G) \cong 0$.
- $\tilde{H}_0(G) \cong \mathbb{Z}^{k-1}$ where k is the number of connected components of G.

Proposition

$\mathcal{S}(G) \cong \mathcal{S}^e(G)$ if and only if G is connected.

The Two Sandpile Groups

- How are $\mathcal{S}(G)$ and $\mathcal{S}^e(G)$ related?
- We can think of these groups as $\ker \epsilon$ and $\operatorname{im} \partial$ respectively with the same equivalence relation.
- $\mathcal{S}(G) \cong \mathcal{S}^e(G)$ if and only if $\tilde{H}_0(G) \cong 0$.
- $\tilde{H}_0(G) \cong \mathbb{Z}^{k-1}$ where k is the number of connected components of G.

Proposition

$\mathcal{S}(G) \cong \mathcal{S}^e(G)$ if and only if G is connected. In general, if $G_1, \ldots, G_k$ are the connected components of G, then $\mathcal{S}^e(G) \cong \mathcal{S}(G_1) \oplus \cdots \oplus \mathcal{S}(G_k)$.

The Two Sandpile Groups

- How are $\mathcal{S}(G)$ and $\mathcal{S}^e(G)$ related?
- We can think of these groups as $\ker \epsilon$ and $\operatorname{im} \partial$ respectively with the same equivalence relation.
- $\mathcal{S}(G) \cong \mathcal{S}^e(G)$ if and only if $\tilde{H}_0(G) \cong 0$.
- $\tilde{H}_0(G) \cong \mathbb{Z}^{k-1}$ where k is the number of connected components of G.

Proposition

$\mathcal{S}(G) \cong \mathcal{S}^e(G)$ if and only if G is connected. In general, if $G_1, \ldots, G_k$ are the connected components of G, then $\mathcal{S}^e(G) \cong \mathcal{S}(G_1) \oplus \cdots \oplus \mathcal{S}(G_k)$.

- Lets play around with $\mathcal{S}^e(G)$.

The Edge-Labeled Sandpile Group

- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $\partial(c_1)$ and $\partial(c_2)$ are firing equivalent.

The Edge-Labeled Sandpile Group

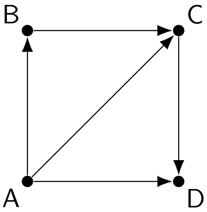
- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $\partial(c_1)$ and $\partial(c_2)$ are firing equivalent.
- Can we make a more useful definition?

The Edge-Labeled Sandpile Group

- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $\partial(c_1)$ and $\partial(c_2)$ are firing equivalent.
- Can we make a more useful definition?
- Let's start by taking a closer look at ∂.

The Edge-Labeled Sandpile Group

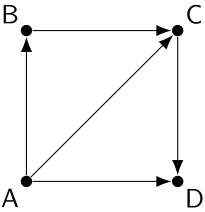
- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $\partial(c_1)$ and $\partial(c_2)$ are firing equivalent.
- Can we make a more useful definition?
- Let's start by taking a closer look at ∂.



$$\begin{array}{c c} & \begin{array}{ccccc} AB & AC & AD & BC & CD \end{array} \\ \begin{array}{c} A \\ B \\ C \\ D \end{array} & \begin{pmatrix} -1 & -1 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 & -1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}$$

The Edge-Labeled Sandpile Group

- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $\partial(c_1)$ and $\partial(c_2)$ are firing equivalent.
- Can we make a more useful definition?
- Let's start by taking a closer look at ∂.

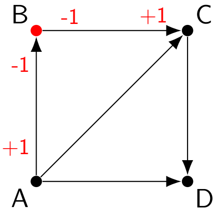


$$\begin{array}{c} \\ A \\ B \\ C \\ D \end{array} \begin{array}{c} \begin{array}{ccccc} AB & AC & AD & BC & CD \end{array} \\ \begin{pmatrix} -1 & -1 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 & -1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}$$

- What happens when we fire a vertex?

The Edge-Labeled Sandpile Group

- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $\partial(c_1)$ and $\partial(c_2)$ are firing equivalent.
- Can we make a more useful definition?
- Let's start by taking a closer look at ∂.

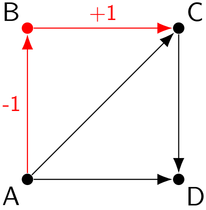


$$\begin{array}{c c} & \begin{array}{ccccc} AB & AC & AD & BC & CD \end{array} \\ \begin{array}{c} A \\ B \\ C \\ D \end{array} & \begin{pmatrix} -1 & -1 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 & -1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}$$

- What happens when we fire a vertex?

The Edge-Labeled Sandpile Group

- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $\partial(c_1)$ and $\partial(c_2)$ are firing equivalent.
- Can we make a more useful definition?
- Let's start by taking a closer look at ∂.

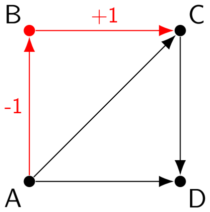


$$\begin{array}{c} \\ A \\ B \\ C \\ D \end{array} \begin{array}{c} \begin{array}{ccccc} AB & AC & AD & BC & CD \end{array} \\ \begin{pmatrix} -1 & -1 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 & -1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}$$

- What happens when we fire a vertex?

The Edge-Labeled Sandpile Group

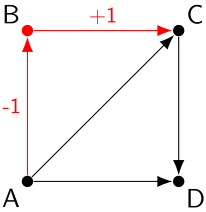
- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $\partial(c_1)$ and $\partial(c_2)$ are firing equivalent.
- Can we make a more useful definition?
- Let's start by taking a closer look at ∂.



$$\begin{array}{c} \\ A \\ B \\ C \\ D \end{array} \begin{array}{c} \begin{array}{ccccc} AB & AC & AD & BC & CD \end{array} \\ \begin{pmatrix} -1 & -1 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 & -1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}$$

- What happens when we fire a vertex?

- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $\partial(c_1)$ and $\partial(c_2)$ are firing equivalent.
- Can we make a more useful definition?
- Let's start by taking a closer look at ∂.

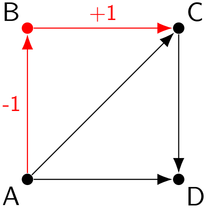


$$\begin{array}{c c} & \begin{array}{ccccc} AB & AC & AD & BC & CD \end{array} \\ \begin{array}{c} A \\ B \\ C \\ D \end{array} & \begin{pmatrix} -1 & -1 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 & -1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}$$

- What happens when we fire a vertex? We subtract a row of ∂.

The Edge-Labeled Sandpile Group

- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $\partial(c_1)$ and $\partial(c_2)$ are firing equivalent.
- Can we make a more useful definition?
- Let's start by taking a closer look at ∂.

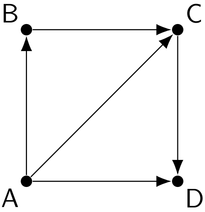


$$\begin{array}{c} \\ A \\ B \\ C \\ D \end{array} \begin{array}{c} \begin{array}{ccccc} AB & AC & AD & BC & CD \end{array} \\ \begin{pmatrix} -1 & -1 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 & -1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}$$

- What happens when we fire a vertex? We subtract a row of ∂. A collection of firings adds an element of $\text{im}_{\mathbb{Z}}(\partial^T)$.

The Edge-Labeled Sandpile Group

- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $\partial(c_1)$ and $\partial(c_2)$ are firing equivalent.
- Can we make a more useful definition?
- Let's start by taking a closer look at ∂.

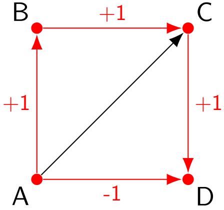


$$\begin{array}{cc} & \begin{matrix} AB & AC & AD & BC & CD \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \end{matrix} & \begin{pmatrix} -1 & -1 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 & -1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}$$

- What happens when we fire a vertex? We subtract a row of ∂. A collection of firings adds an element of $\mathrm{im}_{\mathbb{Z}}(\partial^T)$.
- We can also add a 1-chain with trivial boundary.

The Edge-Labeled Sandpile Group

- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $\partial(c_1)$ and $\partial(c_2)$ are firing equivalent.
- Can we make a more useful definition?
- Let's start by taking a closer look at ∂.

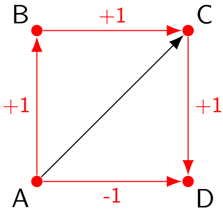


$$\begin{array}{c} \\ A \\ B \\ C \\ D \end{array} \begin{array}{c} \begin{array}{ccccc} AB & AC & AD & BC & CD \end{array} \\ \begin{pmatrix} -1 & -1 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 & -1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}$$

- What happens when we fire a vertex? We subtract a row of ∂. A collection of firings adds an element of $\text{im}_{\mathbb{Z}}(\partial^T)$.
- We can also add a 1-chain with trivial boundary.

The Edge-Labeled Sandpile Group

- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $\partial(c_1)$ and $\partial(c_2)$ are firing equivalent.
- Can we make a more useful definition?
- Let's start by taking a closer look at ∂.

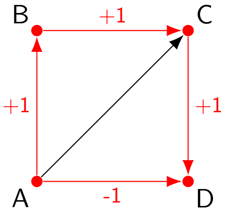


$$\begin{array}{c} \\ A \\ B \\ C \\ D \end{array} \begin{array}{c} \begin{array}{ccccc} AB & AC & AD & BC & CD \end{array} \\ \begin{pmatrix} -1 & -1 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 & -1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}$$

- What happens when we fire a vertex? We subtract a row of ∂. A collection of firings adds an element of $\text{im}_{\mathbb{Z}}(\partial^T)$.
- We can also add a 1-chain with trivial boundary.

The Edge-Labeled Sandpile Group

- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $\partial(c_1)$ and $\partial(c_2)$ are firing equivalent.
- Can we make a more useful definition?
- Let's start by taking a closer look at ∂.

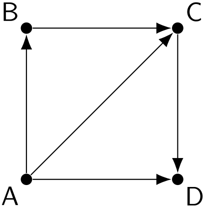


$$\begin{array}{c c} & \begin{array}{ccccc} AB & AC & AD & BC & CD \end{array} \\ \begin{array}{c} A \\ B \\ C \\ D \end{array} & \begin{pmatrix} -1 & -1 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 & -1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}$$

- What happens when we fire a vertex? We subtract a row of ∂. A collection of firings adds an element of $\mathrm{im}_{\mathbb{Z}}(\partial^T)$.
- We can also add a 1-chain with trivial boundary. This new type of firing adds an element of $\ker_{\mathbb{Z}}(\partial)$.

The Edge-Labeled Sandpile Group

- We say that 2 edge configurations c_1 and c_2 are *edge firing equivalent* if $c_1 - c_2 \in \left(\mathrm{im}_{\mathbb{Z}}(\partial^T) \oplus \ker_{\mathbb{Z}}(\partial)\right)$.
- Can we make a more useful definition?
- Let's start by taking a closer look at ∂.



$$\begin{array}{c} \\ A \\ B \\ C \\ D \end{array} \begin{array}{c} \begin{array}{ccccc} AB & AC & AD & BC & CD \end{array} \\ \begin{pmatrix} -1 & -1 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 & -1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}$$

- What happens when we fire a vertex? We subtract a row of ∂. A collection of firings adds an element of $\mathrm{im}_{\mathbb{Z}}(\partial^T)$.
- We can also add a 1-chain with trivial boundary. This new type of firing adds an element of $\ker_{\mathbb{Z}}(\partial)$.

Let's Talk About Matroids

- In general, a *matroid* is a pair $(E, \mathcal{B})$ where $\mathcal{B} \subseteq \mathcal{P}(E)$ satisfying some conditions. E is called the *ground set* while $\mathcal{B}$ is called the set of *bases*. .

Let's Talk About Matroids

- In general, a *matroid* is a pair $(E, \mathcal{B})$ where $\mathcal{B} \subseteq \mathcal{P}(E)$ satisfying some conditions. E is called the *ground set* while $\mathcal{B}$ is called the set of *bases*.
- For this talk, we will only work with *representable matroids*.

Let's Talk About Matroids

- In general, a *matroid* is a pair $(E, \mathcal{B})$ where $\mathcal{B} \subseteq \mathcal{P}(E)$ satisfying some conditions. E is called the *ground set* while $\mathcal{B}$ is called the set of *bases*.
- For this talk, we will only work with *representable matroids*.
- Start with a matrix. The columns form the *ground set* and the maximal linearly independent collections of column vectors are the *bases*.

Let's Talk About Matroids

- In general, a *matroid* is a pair $(E, \mathcal{B})$ where $\mathcal{B} \subseteq \mathcal{P}(E)$ satisfying some conditions. E is called the *ground set* while $\mathcal{B}$ is called the set of *bases*.
- For this talk, we will only work with *representable matroids*.
- Start with a matrix. The columns form the *ground set* and the maximal linearly independent collections of column vectors are the *bases*.

$$\begin{array}{c} \begin{array}{cccccc} A & B & C & D & E & F \end{array} \\ \begin{pmatrix} 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

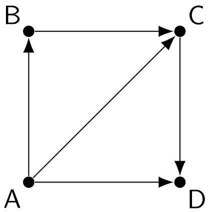
- The matroid arising from this matrix has $E = \{A, B, C, D, E, F\}$ and $\mathcal{B} = \{\{A, B, D\}, \{A, C, D\}, \{A, D, F\}, \{B, C, D\}, \{B, D, F\}\}$.

Graphical Matroid

- What do we get when we find the matroid represented by the boundary matrix of a graph?

Graphical Matroid

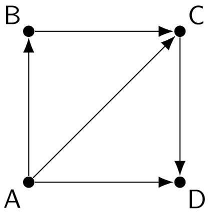
- What do we get when we find the matroid represented by the boundary matrix of a graph?



$$\begin{array}{c|c} & \begin{array}{ccccc} AB & AC & AD & BC & CD \end{array} \\ \begin{array}{c} A \\ B \\ C \\ D \end{array} & \begin{pmatrix} -1 & -1 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 & -1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}$$

Graphical Matroid

- What do we get when we find the matroid represented by the boundary matrix of a graph?

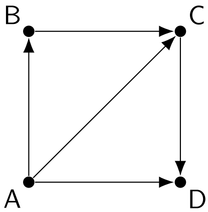


$$\begin{array}{c} \\ A \\ B \\ C \\ D \end{array} \begin{array}{c} \begin{array}{ccccc} AB & AC & AD & BC & CD \end{array} \\ \begin{pmatrix} -1 & -1 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 & -1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}$$

- The ground set corresponds to the edges of G

Graphical Matroid

- What do we get when we find the matroid represented by the boundary matrix of a graph?

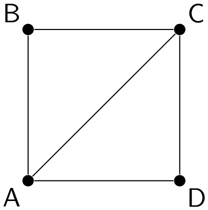


$$
\begin{array}{c}
\\ A \\ B \\ C \\ D
\end{array}
\begin{array}{c}
\begin{array}{ccccc} AB & AC & AD & BC & CD \end{array} \\
\begin{pmatrix}
-1 & -1 & -1 & 0 & 0 \\
1 & 0 & 0 & -1 & 0 \\
0 & 1 & 0 & 1 & -1 \\
0 & 0 & 1 & 0 & 1
\end{pmatrix}
\end{array}
$$

- The ground set corresponds to the edges of G and bases correspond to spanning trees of G.

Another Graph Representation

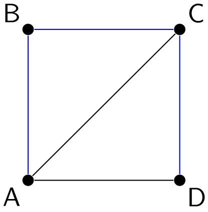
- We give a useful alternative method for producing a matrix D that represents the same matroid as ∂.



$$D = \begin{pmatrix} \quad & \quad & \quad \end{pmatrix}$$

Another Graph Representation

- We give a useful alternative method for producing a matrix D that represents the same matroid as ∂.

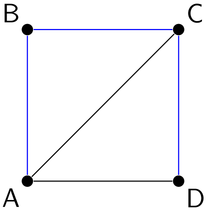


$$D = \begin{pmatrix} & & & \\ & & & \\ & & & \end{pmatrix}$$

1. Choose any spanning tree T.

Another Graph Representation

- We give a useful alternative method for producing a matrix D that represents the same matroid as ∂.

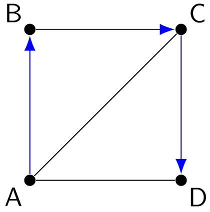


$$D = \begin{array}{c} \begin{array}{ccc} AB & BC & CD \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & \\ 0 & 1 & 0 & \\ 0 & 0 & 1 & \end{pmatrix} \end{array}$$

1. Choose any spanning tree T.
2. Write the identity matrix in the first $|T|$ columns of M. Associate these columns with the edges of T.

Another Graph Representation

- We give a useful alternative method for producing a matrix D that represents the same matroid as ∂.

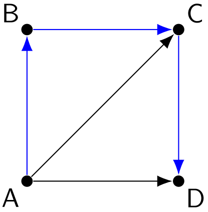


$$D = \begin{array}{ccc} AB & BC & CD \\ \end{array}$$
$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

1. Choose any spanning tree T.
2. Write the identity matrix in the first $|T|$ columns of M. Associate these columns with the edges of T.
3. Choose a leaf of T to act as a *root* and orient the tree away from this point.

Another Graph Representation

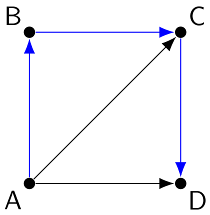
- We give a useful alternative method for producing a matrix D that represents the same matroid as ∂.



$$D = \begin{array}{c} \\ \\ \\ \end{array}\begin{array}{c} \begin{array}{ccccc} AB & BC & CD & AC & AD \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}$$

1. Choose any spanning tree T.
2. Write the identity matrix in the first $|T|$ columns of M. Associate these columns with the edges of T.
3. Choose a leaf of T to act as a *root* and orient the tree away from this point.
4. Orient the remaining edges and fill in the columns to give the desired dependencies. (This step is easier to describe verbally).

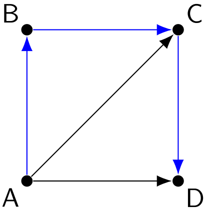
Back to $\mathcal{S}^e(G)$



$$D = \begin{array}{c} \begin{array}{ccccc} AB & BC & CD & AC & AD \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}$$

- We saw before that c_1 and c_2 are edge firing equivalent if and only if they are equivalent modulo $\mathrm{im}_{\mathbb{Z}}(\partial^T) \oplus \ker_{\mathbb{Z}}(\partial)$.

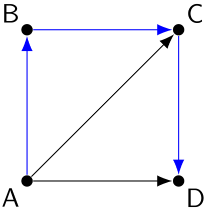
Back to $\mathcal{S}^e(G)$



$$D = \begin{array}{c} \\ \\ \end{array}\begin{array}{ccccc} AB & BC & CD & AC & AD \end{array}$$
$$D = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$$

- We saw before that c_1 and c_2 are edge firing equivalent if and only if they are equivalent modulo $\mathrm{im}_{\mathbb{Z}}(\partial^T) \oplus \ker_{\mathbb{Z}}(\partial)$.
- In fact, c_1 and c_2 are also edge firing equivalent if and only if they are equivalent modulo $\mathrm{im}_{\mathbb{Z}}(D^T) \oplus \ker_{\mathbb{Z}}(D)$.

Back to $\mathcal{S}^e(G)$

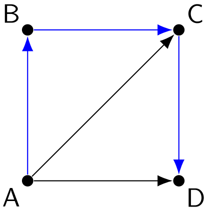


$$D = \begin{array}{c} \\ \end{array}\begin{array}{ccccc} AB & BC & CD & AC & AD \end{array}$$

$$D = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$$

- We saw before that c_1 and c_2 are edge firing equivalent if and only if they are equivalent modulo $\text{im}_{\mathbb{Z}}(\partial^T) \oplus \ker_{\mathbb{Z}}(\partial)$.
- In fact, c_1 and c_2 are also edge firing equivalent if and only if they are equivalent modulo $\text{im}_{\mathbb{Z}}(D^T) \oplus \ker_{\mathbb{Z}}(D)$.
- Let's add a few rows to D to get $\mathcal{D}$. Call the last 2 rows $\hat{D}$.

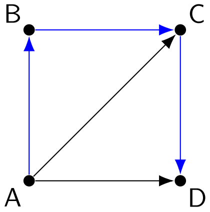
Back to $\mathcal{S}^e(G)$



$$\mathcal{D} = \begin{array}{c} \begin{array}{ccccc} AB & BC & CD & AC & AD \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ -1 & -1 & 0 & 1 & 0 \\ -1 & -1 & -1 & 0 & 1 \end{pmatrix} \end{array}$$

- We saw before that c_1 and c_2 are edge firing equivalent if and only if they are equivalent modulo $\text{im}_{\mathbb{Z}}(\partial^T) \oplus \ker_{\mathbb{Z}}(\partial)$.
- In fact, c_1 and c_2 are also edge firing equivalent if and only if they are equivalent modulo $\text{im}_{\mathbb{Z}}(D^T) \oplus \ker_{\mathbb{Z}}(D)$.
- Let's add a few rows to D to get $\mathcal{D}$. Call the last 2 rows $\hat{D}$.

Back to $\mathcal{S}^e(G)$



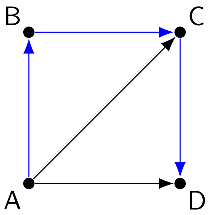
$$\mathcal{D} = \begin{array}{c} \begin{array}{ccccc} AB & BC & CD & AC & AD \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ -1 & -1 & 0 & 1 & 0 \\ -1 & -1 & -1 & 0 & 1 \end{pmatrix} \end{array}$$

- We saw before that c_1 and c_2 are edge firing equivalent if and only if they are equivalent modulo $\mathrm{im}_{\mathbb{Z}}(\partial^T) \oplus \ker_{\mathbb{Z}}(\partial)$.
- In fact, c_1 and c_2 are also edge firing equivalent if and only if they are equivalent modulo $\mathrm{im}_{\mathbb{Z}}(D^T) \oplus \ker_{\mathbb{Z}}(D)$.
- Let's add a few rows to D to get $\mathcal{D}$. Call the last 2 rows $\hat{D}$.

Proposition

$\mathrm{im}_{\mathbb{Z}}(\hat{D}^T) = \ker_{\mathbb{Z}}(D)$.

Back to $\mathcal{S}^e(G)$



$$\mathcal{D} = \begin{array}{c} \begin{array}{ccccc} AB & BC & CD & AC & AD \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ -1 & -1 & 0 & 1 & 0 \\ -1 & -1 & -1 & 0 & 1 \end{pmatrix} \end{array}$$

- We saw before that c_1 and c_2 are edge firing equivalent if and only if they are equivalent modulo $\text{im}_{\mathbb{Z}}(\partial^T) \oplus \text{ker}_{\mathbb{Z}}(\partial)$.
- In fact, c_1 and c_2 are also edge firing equivalent if and only if they are equivalent modulo $\text{im}_{\mathbb{Z}}(D^T) \oplus \text{ker}_{\mathbb{Z}}(D)$.
- Let's add a few rows to D to get $\mathcal{D}$. Call the last 2 rows $\hat{D}$.

Proposition

$\text{im}_{\mathbb{Z}}(\hat{D}^T) = \text{ker}_{\mathbb{Z}}(D)$. *This implies that* $\text{im}_{\mathbb{Z}}(D^T) \oplus \text{ker}_{\mathbb{Z}}(D) = \text{im}_{\mathbb{Z}}(\mathcal{D}^T)$

The Matrix Sandpile Group

- Let D be an $n \times (n+m)$ matrix of the form: $\left(\begin{array}{c|c} I_n & M \end{array} \right)$ where M is any $n \times m$ integer matrix.

The Matrix Sandpile Group

- Let D be an $n \times (n+m)$ matrix of the form: $\left(\begin{array}{c|c} I_n & M \end{array} \right)$ where M is any $n \times m$ integer matrix.

Definition (Matrix Sandpile Group)

The *sandpile lattice* of D, denoted $\mathcal{S}(D)$, is $\mathrm{im}_{\mathbb{Z}}(D^T) \oplus \ker_{\mathbb{Z}}(D)$. The *sandpile group* of D is $\mathbb{Z}^{n+m} / \left(\mathrm{im}_{\mathbb{Z}}(D^T) \oplus \ker_{\mathbb{Z}}(D)\right)$.

The Matrix Sandpile Group

- Let D be an $n \times (n+m)$ matrix of the form: $\left(\begin{array}{c|c} I_n & M \end{array} \right)$ where M is any $n \times m$ integer matrix.

Definition (Matrix Sandpile Group)

The *sandpile lattice* of D, denoted $\mathcal{S}(D)$, is $\mathrm{im}_{\mathbb{Z}}(D^T) \oplus \ker_{\mathbb{Z}}(D)$. The *sandpile group* of D is $\mathbb{Z}^{n+m}/\left(\mathrm{im}_{\mathbb{Z}}(D^T) \oplus \ker_{\mathbb{Z}}(D)\right)$.

- Let $\hat{D}$ be the $m \times (n+m)$ matrix $\left(\begin{array}{c|c} -M^T & I_m \end{array} \right)$.

The Matrix Sandpile Group

- Let D be an $n \times (n+m)$ matrix of the form: $\left(\begin{array}{c|c} I_n & M \end{array} \right)$ where M is any $n \times m$ integer matrix.

Definition (Matrix Sandpile Group)

The *sandpile lattice* of D, denoted $\mathcal{S}(D)$, is $\text{im}_{\mathbb{Z}}(D^T) \oplus \text{ker}_{\mathbb{Z}}(D)$. The *sandpile group* of D is $\mathbb{Z}^{n+m} / \left(\text{im}_{\mathbb{Z}}(D^T) \oplus \text{ker}_{\mathbb{Z}}(D)\right)$.

- Let $\hat{D}$ be the $m \times (n+m)$ matrix $\left(\begin{array}{c|c} -M^T & I_m \end{array} \right)$.
- Let $\mathcal{D}$ be the $(n+m) \times (n+m)$ matrix $\left(\frac{D}{\hat{D}} \right) = \left(\begin{array}{c|c} I_n & M \\ \hline -M^T & I_m \end{array} \right)$.

The Matrix Sandpile Group

- Let D be an $n \times (n+m)$ matrix of the form: $\left(\begin{array}{c|c} I_n & M \end{array} \right)$ where M is any $n \times m$ integer matrix.

Definition (Matrix Sandpile Group)

The *sandpile lattice* of D, denoted $\mathcal{S}(D)$, is $\mathrm{im}_{\mathbb{Z}}(D^T) \oplus \ker_{\mathbb{Z}}(D)$. The *sandpile group* of D is $\mathbb{Z}^{n+m} / \left(\mathrm{im}_{\mathbb{Z}}(D^T) \oplus \ker_{\mathbb{Z}}(D)\right)$.

- Let $\hat{D}$ be the $m \times (n+m)$ matrix $\left(\begin{array}{c|c} -M^T & I_m \end{array} \right)$.
- Let $\mathcal{D}$ be the $(n+m) \times (n+m)$ matrix $\left(\begin{array}{c} D \\ \hline \hat{D} \end{array} \right) = \left(\begin{array}{c|c} I_n & M \\ \hline -M^T & I_m \end{array} \right)$.

Theorem

$\mathrm{im}_{\mathbb{Z}}(D^T) \oplus \ker_{\mathbb{Z}}(D) = \mathrm{im}_{\mathbb{Z}}(\mathcal{D}^T)$.

The Matrix Sandpile Group

- Let D be an $n \times (n+m)$ matrix of the form: $\left(\begin{array}{c|c} I_n & M \end{array} \right)$ where M is any $n \times m$ integer matrix.

Definition (Matrix Sandpile Group)

The *sandpile lattice* of D, denoted $\mathcal{S}(D)$, is $\mathrm{im}_{\mathbb{Z}}(D^T) \oplus \ker_{\mathbb{Z}}(D)$. The *sandpile group* of D is $\mathbb{Z}^{n+m} / \left(\mathrm{im}_{\mathbb{Z}}(D^T) \oplus \ker_{\mathbb{Z}}(D) \right)$.

- Let $\hat{D}$ be the $m \times (n+m)$ matrix $\left(\begin{array}{c|c} -M^T & I_m \end{array} \right)$.
- Let $\mathcal{D}$ be the $(n+m) \times (n+m)$ matrix $\left(\begin{array}{c} D \\ \hline \hat{D} \end{array} \right) = \left(\begin{array}{c|c} I_n & M \\ \hline -M^T & I_m \end{array} \right)$.

Theorem

$\mathrm{im}_{\mathbb{Z}}(D^T) \oplus \ker_{\mathbb{Z}}(D) = \mathrm{im}_{\mathbb{Z}}(\mathcal{D}^T)$. *Furthermore,* $|\mathcal{S}(D)| = |\mathcal{D}|$.

Finding $|\mathcal{D}|$

Lemma

$|\mathcal{D}| = |D \cdot D^T|$

Finding $|\mathcal{D}|$

Lemma

$|\mathcal{D}| = |D \cdot D^T|$

Proof.

$$\mathcal{D} = \left(\begin{array}{c|c} I_n & M \\ \hline -M^T & I_m \end{array}\right) \to \left(\begin{array}{c|c} M \cdot M^T + I_n & 0 \\ \hline -M^T & I_m \end{array}\right) = \left(\begin{array}{c|c} D \cdot D^T & 0 \\ \hline -M^T & I_m \end{array}\right) \to ☺$$

$\square$

Finding $|\mathcal{D}|$

Lemma

$|\mathcal{D}| = |D \cdot D^T|$

Proof.

$$\mathcal{D} = \left(\begin{array}{c|c} I_n & M \\ \hline -M^T & I_m \end{array}\right) \to \left(\begin{array}{c|c} M \cdot M^T + I_n & 0 \\ \hline -M^T & I_m \end{array}\right) = \left(\begin{array}{c|c} D \cdot D^T & 0 \\ \hline -M^T & I_m \end{array}\right) \to ☺$$

□

- For $s \subset [n+m]$, let D_s be D restricted to columns of s.

Finding $|\mathcal{D}|$

Lemma

$|\mathcal{D}| = |D \cdot D^T|$

Proof.

$$\mathcal{D} = \left(\begin{array}{c|c} I_n & M \\ \hline -M^T & I_m \end{array}\right) \to \left(\begin{array}{c|c} M \cdot M^T + I_n & 0 \\ \hline -M^T & I_m \end{array}\right) = \left(\begin{array}{c|c} D \cdot D^T & 0 \\ \hline -M^T & I_m \end{array}\right) \to ☺$$

$\square$

- For $s \subset [n+m]$, let D_s be D restricted to columns of s.

Theorem (Consequence of Cauchy-Biney Formula)

$$|D \cdot D^T| = \sum_{s \subseteq [n+m], |s| = n} |D_s|^2$$

Cellular Matrix-Tree Theorem

Theorem (Duval-Klivans-Martin, 2009)

$$|\mathcal{S}(D)| = \sum_{s \subseteq [n+m], |s|=n} |D_s|^2$$

Cellular Matrix-Tree Theorem

Theorem (Duval-Klivans-Martin, 2009)

$$|\mathcal{S}(D)| = \sum_{s \subseteq [n+m], |s| = n} |D_s|^2$$

$$D = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 1 & 3 \end{pmatrix} \qquad \mathcal{D} = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 1 & 3 \\ -2 & -3 & 1 \end{pmatrix}$$

Cellular Matrix-Tree Theorem

Theorem (Duval-Klivans-Martin, 2009)

$$|\mathcal{S}(D)| = \sum_{s \subseteq [n+m], |s|=n} |D_s|^2$$

$$D = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 1 & 3 \end{pmatrix} \qquad \mathcal{D} = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 1 & 3 \\ -2 & -3 & 1 \end{pmatrix}$$

$$|\mathcal{S}(D)| = |\mathcal{D}| = \begin{vmatrix} 1 & 0 \\ 0 & 1 \end{vmatrix}^2 + \begin{vmatrix} 1 & 2 \\ 0 & 3 \end{vmatrix}^2 + \begin{vmatrix} 0 & 2 \\ 1 & 3 \end{vmatrix}^2 = 1^2 + 3^2 + (-2)^2 = 14$$

Cellular Matrix-Tree Theorem

Theorem (Duval-Klivans-Martin, 2009)

$$|\mathcal{S}(D)| = \sum_{s\subseteq[n+m],|s|=n} |D_s|^2$$

$$D = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 1 & 3 \end{pmatrix} \qquad \mathcal{D} = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 1 & 3 \\ -2 & -3 & 1 \end{pmatrix}$$

$$|\mathcal{S}(D)| = |\mathcal{D}| = \begin{vmatrix} 1 & 0 \\ 0 & 1 \end{vmatrix}^2 + \begin{vmatrix} 1 & 2 \\ 0 & 3 \end{vmatrix}^2 + \begin{vmatrix} 0 & 2 \\ 1 & 3 \end{vmatrix}^2 = 1^2 + 3^2 + (-2)^2 = 14$$

- For each $s \subseteq [n+m]$ of size n, $|D_s| \neq 0$ iff s corresponds to a basis of D.

Cellular Matrix-Tree Theorem

Theorem (Duval-Klivans-Martin, 2009)

$$|\mathcal{S}(D)| = \sum_{s \in \mathcal{B}(D)} |D_s|^2$$

$$D = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 1 & 3 \end{pmatrix} \qquad \mathcal{D} = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 1 & 3 \\ -2 & -3 & 1 \end{pmatrix}$$

$$|\mathcal{S}(D)| = |\mathcal{D}| = \begin{vmatrix} 1 & 0 \\ 0 & 1 \end{vmatrix}^2 + \begin{vmatrix} 1 & 2 \\ 0 & 3 \end{vmatrix}^2 + \begin{vmatrix} 0 & 2 \\ 1 & 3 \end{vmatrix}^2 = 1^2 + 3^2 + (-2)^2 = 14$$

- For each $s \subseteq [n+m]$ of size n, $|D_s| \neq 0$ iff s corresponds to a basis of D.

Regular Matroids

Definition

A matrix whose square submatrices all have determinant 0, 1, or -1 is called *totally unimodular*.

Regular Matroids

Definition

A matrix whose square submatrices all have determinant 0, 1, or -1 is called *totally unimodular*. A matroid represented by a totally unimodular matrix is called *regular*.

Regular Matroids

Definition

A matrix whose square submatrices all have determinant 0, 1, or -1 is called *totally unimodular*. A matroid represented by a totally unimodular matrix is called *regular*.

Proposition

If D is totally unimodular, $|\mathcal{S}(D)| = |\mathcal{B}(D)|$.

Regular Matroids

Definition

A matrix whose square submatrices all have determinant 0, 1, or -1 is called *totally unimodular*. A matroid represented by a totally unimodular matrix is called *regular*.

Proposition

If D is totally unimodular, $|\mathcal{S}(D)| = |\mathcal{B}(D)|$.

Proposition (Tutte, 1958)

Graphical matroids are always regular.

Regular Matroids

Definition

A matrix whose square submatrices all have determinant 0, 1, or -1 is called *totally unimodular*. A matroid represented by a totally unimodular matrix is called *regular*.

Proposition

If D is totally unimodular, $|\mathcal{S}(D)| = |\mathcal{B}(D)|$.

Proposition (Tutte, 1958)

Graphical matroids are always regular.

- This proves the Sandpile Matrix-Tree Theorem for graphs.

Theorem (Sandpile Matrix-Tree Theorem)

The size of $\mathcal{S}(G)$ is the number of spanning trees of G.

Cell Complex Chip-Firing

- A *simplicial complex* is a higher dimensional analogue of a graph.

Cell Complex Chip-Firing

- A *simplicial complex* is a higher dimensional analogue of a graph. We can construct a matroid on an n-dimensional simplicial complex from the boundary map from n-chains to $(n-1)$-chains.

Cell Complex Chip-Firing

- A *simplicial complex* is a higher dimensional analogue of a graph. We can construct a matroid on an n-dimensional simplicial complex from the boundary map from n-chains to $(n-1)$-chains.

Theorem (Cordovil-Las Vergnas, Lindström 1979)

The complete 2-dimensional simplicial complex on 6 vertices is not regular.

Cell Complex Chip-Firing

- A *simplicial complex* is a higher dimensional analogue of a graph. We can construct a matroid on an n-dimensional simplicial complex from the boundary map from n-chains to $(n-1)$-chains.

Theorem (Cordovil-Las Vergnas, Lindström 1979)

The complete 2-dimensional simplicial complex on 6 vertices is not regular.

- We can also work with *cell complexes* (aka finite CW complexes) whose boundary maps can be any collection of integers.

Cell Complex Chip-Firing

- A *simplicial complex* is a higher dimensional analogue of a graph. We can construct a matroid on an n-dimensional simplicial complex from the boundary map from n-chains to $(n-1)$-chains.

Theorem (Cordovil-Las Vergnas, Lindström 1979)

The complete 2-dimensional simplicial complex on 6 vertices is not regular.

- We can also work with *cell complexes* (aka finite CW complexes) whose boundary maps can be any collection of integers.
- In general, given a cell complex Σ, we want to convert the top boundary map to the form $\left(\begin{array}{c|c} I_n & M \end{array} \right)$ and then we get the Cellular Matrix-Tree Theorem.

Cell Complex Chip-Firing

- A *simplicial complex* is a higher dimensional analogue of a graph. We can construct a matroid on an n-dimensional simplicial complex from the boundary map from n-chains to $(n-1)$-chains.

Theorem (Cordovil-Las Vergnas, Lindström 1979)

The complete 2-dimensional simplicial complex on 6 vertices is not regular.

- We can also work with *cell complexes* (aka finite CW complexes) whose boundary maps can be any collection of integers.
- In general, given a cell complex Σ, we want to convert the top boundary map to the form $\left(\begin{array}{c|c} I_n & M \end{array} \right)$ and then we get the Cellular Matrix-Tree Theorem.
- This works as long as Σ contains at least one basis with determinant 1 or -1.

Cell Complex Chip-Firing

- A *simplicial complex* is a higher dimensional analogue of a graph. We can construct a matroid on an n-dimensional simplicial complex from the boundary map from n-chains to $(n-1)$-chains.

Theorem (Cordovil-Las Vergnas, Lindström 1979)

The complete 2-dimensional simplicial complex on 6 vertices is not regular.

- We can also work with *cell complexes* (aka finite CW complexes) whose boundary maps can be any collection of integers.
- In general, given a cell complex Σ, we want to convert the top boundary map to the form $\left(\begin{array}{c|c} I_n & M \end{array} \right)$ and then we get the Cellular Matrix-Tree Theorem.
- This works as long as Σ contains at least one basis with determinant 1 or -1. In particular, this works when Σ is the complete simplicial complex in any dimension and with any number of vertices.

Sandpile Torsors

Theorem (Duval-Klivans-Martin, 2009)

$$|\mathcal{S}(D)| = \sum_{s \in \mathcal{B}(D)} |D_s|^2$$

- Can we find a combinatorially interesting map f from $\mathcal{S}(D) \to \mathcal{B}(D)$ such that for each $s \in B(D)$, we have $f^{-1}(s) = |D_s|^2$?

Sandpile Torsors

Theorem (Duval-Klivans-Martin, 2009)

$$|\mathcal{S}(D)| = \sum_{s \in \mathcal{B}(D)} |D_s|^2$$

- Can we find a combinatorially interesting map f from $\mathcal{S}(D) \to \mathcal{B}(D)$ such that for each $s \in B(D)$, we have $f^{-1}(s) = |D_s|^2$?
- When D is from a graphical matroid, these are bijections called *sandpile torsors*.

Sandpile Torsors

Theorem (Duval-Klivans-Martin, 2009)

$$|\mathcal{S}(D)| = \sum_{s \in \mathcal{B}(D)} |D_s|^2$$

- Can we find a combinatorially interesting map f from $\mathcal{S}(D) \to \mathcal{B}(D)$ such that for each $s \in B(D)$, we have $f^{-1}(s) = |D_s|^2$?
- When D is from a graphical matroid, these are bijections called *sandpile torsors*.
- Some sandpile torsors are from the rotor-routing process (Holroyd et al, (2008)) and the Bernardi Process (Baker-Wang, (2017)).

Sandpile Torsors

Theorem (Duval-Klivans-Martin, 2009)

$$|\mathcal{S}(D)| = \sum_{s \in \mathcal{B}(D)} |D_s|^2$$

- Can we find a combinatorially interesting map f from $\mathcal{S}(D) \to \mathcal{B}(D)$ such that for each $s \in B(D)$, we have $f^{-1}(s) = |D_s|^2$?
- When D is from a graphical matroid, these are bijections called *sandpile torsors*.
- Some sandpile torsors are from the rotor-routing process (Holroyd et al, (2008)) and the Bernardi Process (Baker-Wang, (2017)).
- When D is from a regular matroid, a bijection is given by Backman, Baker, and Yuen (2019).

Sandpile Torsors

Theorem (Duval-Klivans-Martin, 2009)

$$|\mathcal{S}(D)| = \sum_{s \in \mathcal{B}(D)} |D_s|^2$$

- Can we find a combinatorially interesting map f from $\mathcal{S}(D) \to \mathcal{B}(D)$ such that for each $s \in B(D)$, we have $f^{-1}(s) = |D_s|^2$?
- When D is from a graphical matroid, these are bijections called *sandpile torsors*.
- Some sandpile torsors are from the rotor-routing process (Holroyd et al, (2008)) and the Bernardi Process (Baker-Wang, (2017)).
- When D is from a regular matroid, a bijection is given by Backman, Baker, and Yuen (2019).
- The general result for any $D = \left(\begin{array}{c|c} I_n & M \end{array} \right)$ will appear in my upcoming paper.

Thanks For Listening!!!