The halting problem for chip-firing on finite directed graphs

Cornell University

Mathematics Department Senior Thesis

Matthew Farrell[*]   Lionel Levine[†]

December 2, 2014

We consider a chip-firing game on finite directed graphs and give an answer to a question posed by Bjorner, Lovasz, and Shor in 1991: given an initial configuration of chips, when does it stabilize? The approach they took to address this halting problem involves computing a period vector p with the property that toppling the vertices according to p results in the original configuration, and then checking if it is possible to topple according to p legally (without any of the vertices ever having negative chips). This approach is problematic because the entries of p can grow exponentially in the number of vertices. We make precise a measure of "Eulerianness" and show that, in addition to graphs with a high degree of Eulerianness, for relatively "anti-Eulerian" graphs you can do much better. In addition, we take steps toward a potential proof that the problem is NP-Hard by reducing an NP-Hard problem in the context of another chip-firing game, called the dollar game, to our problem (where the number of edges making up the graphs could be a very fast-growing function of the number of vertices). The ideas developed in the course of answering this stabilization question give rise to a natural generalization of the BEST theorem to general directed graphs.

1   Introduction

Chip-firing on directed graphs belongs to an interesting class of network automaton known as *abelian networks*. In an abelian network we define an action on the vertices of a graph such that changing the order in which we apply the action does not affect the end state. Despite this step down from the full complexity of automata, the dynamics of chip-firing are still rich and many open questions remain that probe at fundamental aspects of system order and behavior. This paper gives an answer to one such question.

Before we can pose the question, we need to describe the chip-firing model. Given a finite, directed, strongly connected multigraph $G = (V, E)$ with $|V| = n$ and a collection of indistinguishible chips, we distribute the chips among the vertices. Labeling vertices as $V = \{v_1, v_2, ..., v_n\}$, to keep track of these chips we define a **full chip configuration** σ, or **full configuration** for short, to be a vector in $\mathbb{Z}^n$, where the value of $\sigma(i)$ is meant to represents the number of chips on the vertex v_i. For additional flexibility, given a vertex v with $v = v_i$ we abuse notation and write $\sigma(v)$ to mean $\sigma(i)$.

We let d_i denote the outdegree of vertex v_i, and we say v_i is **stable** with respect to σ if $\sigma(i) < d_i$ and **active** otherwise. We say that σ is **stable** if every vertex is stable with respect to σ. Given vertices v_i and v_j, we let d_{ij} denote the number of edges from v_i to v_j. We commit an abuse of

---

[*]Arts and Sciences, Class of (January) 2015
[†]Senior Thesis Advisor

notation similar to before, writing d_v to mean the outdegree of vertex v, d_{vw} to denote the number of edges from v to w, and sometimes we mix notation, letting d_{iv} denote the number of edges from v_i to v. When we write $\mathbb{N}$, we include 0. Given a vector $\mathbf{x}$, we let $|\mathbf{x}| = \mathbf{1} \cdot \mathbf{x}$ and call $|\mathbf{x}|$ the degree of $\mathbf{x}$.

The action on the vertices in this model is known as **toppling**, wherin a vertex sends one chip along each of its outgoing edges to its neighbors. To more succinctly describe this toppling action, we define the **Laplacian** of G to be the matrix $\Delta = D - A^T$ where D is the diagonal matrix whose (i, i) entry is d_i and $A = (d_{ij})$ is the adjacency matrix of G. That is,

$$\Delta_{ij} = \begin{cases} -d_{ji} & \text{for } i \neq j \\ d_i - d_{ii} & \text{for } i = j \end{cases}.$$

Now given a configuration σ with active vertex v_i, let σ' be the result of toppling v_i. We have that

$$\sigma'(j) = \begin{cases} \sigma(j) + d_{ij} & \text{for } i \neq j \\ \sigma(i) - d_i & \text{for } i = j \end{cases},$$

or more succinctly $\sigma' = \sigma - \Delta e_i$. We can specify a sequence of topplings with a **toppling vector** $\mathbf{x} \in \mathbb{N}^n$, with $\mathbf{x}(i)$ being the number of times we topple v_i. Since our toppling action is abelian ([9]), if we start with a configuration σ and write σ' as the result of toppling according to $\mathbf{x}$, we have that σ' is uniquely determined by $\mathbf{x}$; that is, $\sigma' = \sigma - \Delta \mathbf{x}$. In other words, the order in which we topple vertices does not matter, only the number of times we topple each vertex. We say that toppling v is **legal** if v is active, and **illegal** if v is not active. We say that σ is **stabilizable** if there is a legal sequence of topplings resulting in a configuration σ' such that σ' is **stable**.

1.1 Sandpiles and the halting problem

The goal of this paper is to address a question asked by Björner, Lovász, and Shor ([10, 3]): given a finite, directed multigraph G and a full configuration σ with $\sigma \geq 0$, when does σ stabilize? We call the problem of computing whether or not σ stabilizes the halting problem for chip-firing on finite directed multigraphs, or simply halting problem if there is no confusion.

In this paper we restrict our consideration to **strongly connected** multigraphs, that is, graphs for which there is a directed path from each vertex v to every other vertex w. As Björner and Lovász demonstrate in [3], G can be broken into strongly connected components in such a way that the halting problem on G reduces to considering the problem for each of the components, so we lose little generality in requiring that G be strongly connected. We hence pose our problem as follows.

Problem. The **halting problem for chip-firing on finite, strongly connected multigraphs**.

Input: A finite, strongly connected multigraph G and a full chip configuration σ with $\sigma \geq 0$.

Output: A decision on whether or not σ stabilizes.

The following result frees us from considering only legal toppling sequences in looking for an answer to the halting problem on finite directed graphs.

Lemma 1 ([9])**.** (Least Action Principle) *A full configuration σ is stabilizable if and only if there exists an $\mathbf{x} \in \mathbb{N}^n$ such that $\sigma - \Delta \mathbf{x}$ is stable.*

A sizeable portion of the ground soon to be covered is motivated by the following principle: in looking for a stabilizing toppling sequence, instead of toppling willy-nilly we can establish some structure by choosing a special vertex s that we "topple last" (in a sense to be made more explicit),

and we will call this vertex the **sink**. The process is as follows: we topple active, nonsink vertices until each of the nonsink vertices are stable. At this point we then topple the sink (once), repeat the non-sink stabilization, topple the sink, and repeat, paying attention to what shows up as we go along.

Specifying a sink gives rise to a familiar object in the context of chip-firing games, the *abelian sandpile*. We begin by defining the sandpile analogue of the Laplacian, the **reduced Laplacian** Δ_s which is gotten by deleting the row and column of Δ corresponding with the sink.

Definition 2. Let $G = (V, E)$ be a finite directed strongly connected multigraph. Specify a global sink s. The **sandpile group** of G is the group quotient

$$S(G, s) = \mathbb{Z}^{n-1}/\Delta_s\mathbb{Z}^{n-1}$$

where $\Delta_s\mathbb{Z}^{n-1}$ is the integer column-span of Δ_s.

If the choice of sink is clear, we sometimes write $S(G, s)$. We call a chip configuration $\eta \in \mathbb{Z}^{n-1}$ indexed by the nonsink nodes of G a **sandpile chip configuration**, or **sandpile** for short (as opposed to *full* chip configurations belonging to $\mathbb{Z}^n$). The definitions "stable" and" toppling vector" have direct analogues in the context of sandpiles: a sandpile η is stable if $\eta(i) < d_i$ for all $v_i \neq s$, and we have toppling vectors $\mathbf{x} \in \mathbb{Z}^{n-1}$. On strongly connected graphs, every sandpile is stabilizable [1], and we denote the stabilization of η as η°. The sandpile group treats as equivalent sandpiles such that one can be gotten from the other by toppling nonsink vertices.

To help us go back and forth between sandpiles and full configurations, we make the following definitions: given $\sigma \in \mathbb{Z}^n$, we let $\tilde{\sigma}$ denote the restriction of σ to the nonsink vertices. Given a $\eta \in \mathbb{Z}^{n-1}$, we let η_k denote the extension of η to $\mathbb{Z}^n$ such that $|\eta_k| = k$. Note that η and k together uniquely determine η_k, since $|\eta| + \eta_k(s) = k$.

We begin with some useful properties of sandpiles.

Lemma 3 ([5])**.** *The order of $S(G, s)$ is the determinant of Δ_s.*

There is another interesting and useful connection with what are called the oriented spanning trees of G.

Definition 4. An **oriented spanning tree** of a digraph $G = (V, E)$ rooted at $r \in V$ is a spanning subgraph $T = (V, A)$ such that

1. Every vertex $v \neq r$ has outdegree 1.

2. r has outdegree 0.

3. T has no oriented cycles.

Hence an oriented spanning tree has as its limbs edges that point toward the root. Let $\mathcal{T}_G(v)$ denote the number of oriented spanning trees in G rooted at v.

Theorem 5 ([8])**.** (Kirchhoff's matrix tree theorem) *Let G be a directed graph. Then $T_G(s)$ is the determinant of the reduced laplacian for $S(G, s)$.*

To summarize, we have that $|S(G, s)| = \det \Delta_s = T_G(s)$. In particular, if G is strongly connected then Δ_s is invertible and Δ has rank $n - 1$.

It turns out that there is a natural representative for each equivalence class of $S(G, s)$. To describe these, we say that a sandpile η is **accessible** if from any other sandpile it is possible to obtain η by a combination of adding chips and selectively firing active vertices. A sandpile that is both stable and accessible is called **recurrent**.

We then have the following:

Theorem 6 ([5]). *The set of all recurrent chip configurations on G is an abelian group under the operation $(\sigma, \eta) \mapsto (\sigma + \eta)^\circ$, and it is isomorphic via the inclusion map to the sandpile group $S(G, s)$.*

Despite this isomorphism, it will be convenient to consider the set of recurrent chip configurations on G as an individual entity, so we give it the notation $S_r(G, s)$. We denote the group operation on recurrent sandpiles by $\oplus$ (we will write the operation additively).

Let $\sigma_{\max}$ denote the maximal (sometimes called "saturated") stable full configuration, so $\sigma_{\max}(i) = d_i - 1$ for all i, and let $m_G = |\sigma_{\max}|$. Clearly if the number of chips on G exceeds m_G, then it is possible to topple active vertices indefinitely. We consider a natural question that arises: what conditions on G will ensure that toppling ceases if the number of chips on G is exactly m_G? Graphs with this property will turn out to be in some senses opposite to Eulerian graphs. With this in mind, we call a graph **co-Eulerian** if it has the property that $V_0/\Delta\mathbb{Z}^n$ is trivial, where $V_0 = \{\mathbf{x} \in \mathbb{Z}^n : |\mathbf{x}| = 0\}$. Our main result shows that co-Eulerian graphs have the property we seek (the reasoning behind the name "co-Eulerian" will become clearer as we develop more theory). Let β_s be the sandpile such that $\beta_s(i) = d_{si}$, and let e_s denote the identity of $S_r(G, s)$.

Theorem 7. *Let G be a strongly connected, directed multigraph. Then the following are equivalent.*

1. Every full configuration σ with $|\sigma| = m_G$ stabilizes.

2. For some choice of sink s, $S(G, s)$ is cyclic with generator $\overline{\beta}$. Equivalently, $S_r(G, s)$ is cyclic with generator $\gamma_s = (\beta_s + e_s)^\circ$.

3. The greatest common divisor M of the numbers $\{|S(G, v)| v \in V\}$ is 1.

4. G is co-Eulerian.

2 Cyclic Subgroups of the Sandpile Group

In accordance with our principle of controlled sink toppling, given a recurrent sandpile σ we are interested in
$$C_\sigma = \{(\sigma + k\beta_s)^\circ : k \in \mathbb{N}\},$$
the set of sandpiles that appear starting from σ when we topple the sink, stabilize, and repeat. It is a basic property of recurrent configurations that adding chips to the nonsink vertices and stabilizing results in another recurrent configuration ([1]), so that C_σ is a set of recurrent configurations. We want to consider recurrent configurations as equivalent if they are in the same C_σ for some recurrent σ (that is, if one can be reached from the other by repeatedly toppling the sink and then stabilizing). This is an equivalence relation, so $S_r(G, s)$ is partitioned by the C_σ. Note that we can write
$$(\sigma + \beta_s)^\circ = (\sigma + e_s + \beta_s)^\circ = \sigma \oplus \gamma_s$$
where $\gamma_s = (e_s + \beta_s)^\circ$ is recurrent. It follows that $C_\sigma = \sigma \oplus \langle\gamma_s\rangle$ where $\langle\gamma_s\rangle$ denotes the cyclic subgroup of $S_r(G, s)$ generated by γ_s. Then $|C_\sigma| = \text{ord}(\gamma_s)$ where $\text{ord}(\gamma_s)$ denotes the order of γ_s as an element of $S_r(G, s)$.

To investigate these cosets of the sandpile group, we will first need a concept introduced by ([3]), that of a period vector.

Definition 8. Given a graph G with full laplacian Δ, a vector $\mathbf{p} \in \mathbb{Z}^n$ is called a **period vector** if it is non-negative and $\Delta\mathbf{p} = 0$. A period vector is **primitive** if its entries have no non-trivial common divisor.

4

In other words, a period vector p is a toppling vector such that starting with a full configuration σ and toppling each vertex $v \in V$ a total of $\mathbf{p}(v)$ times get us back to σ. The following Lemma sums up some useful properties of period vectors.

Lemma 9 ([3]). *We have the following:*

1. Every strongly connected digraph G has a unique primitive period vector π_G. It is strictly positive, and all period vectors are of the form $t\pi_G$, $t = 1, 2, \dots$.

2. If G is connected eulerian, then $\pi_G = \mathbf{1}$.

We now introduce a very special period vector. Recall that $\mathcal{T}_G(v)$ denotes the number of oriented spanning trees in G rooted at v.

Lemma 10 ([6, 7]). *We have that* $(\mathcal{T}_G(v_1), \mathcal{T}_G(v_2), ..., \mathcal{T}_G(v_n)) \in \ker \Delta$.

Let M be the greatest common divisor of the numbers in $\{\mathcal{T}_G(v)|v \in G\}$. By Lemma 9, the vector $\pi = \frac{1}{M}\left(\mathcal{T}_G(v_1), \mathcal{T}_G(v_2), ..., \mathcal{T}_G(v_n)\right)$ is the unique primitive period vector of G. Recalling the definition $\gamma_s = (e_s + \beta_s)^\circ$, we have that $\operatorname{ord}(\gamma_s) = \pi(s)$ since $\pi(s)$ is the least positive integer m such that $(e_s + m\beta_s)^\circ = e_s$; that is, the least positive number of times we need to topple the sink, starting from e_s, before we can return to e_s. More explicitly, we have that $e_s + \pi(s)\beta_s \equiv e_s$ mod $\Delta_s \mathbb{Z}^{n-1}$ since, starting from the sandpile $e_s + \pi(s)\beta_s$, we can topple each nonsink vertex v_i a total of $\pi(i)$ times and return to the configuration e_s. Since e_s is recurrent it follows that $(e_s + \pi(s)\beta_s)^\circ = e_s$. Upon observing that $\pi(s) = \mathcal{T}_G(s)/M$, from this discussion we conclude the following.

Lemma 11. *We have that*

$$\operatorname{ord}(\gamma_s) = \mathcal{T}_G(s)/M = |S(G, s)|/M.$$

so that M is the number of distinct cosets of $\langle \gamma_s \rangle$ in $S(G, s)$; that is,

$$|S(G, s)/\langle \gamma_s \rangle| = M.$$

The value M was originally studied in ([2]) as a way to count unicycles in the rotor-router model. We will demonstrate that it also serves as a measure of the "Eulerianness" of a graph.

2.1 An Isomorphism of Groups

We now investigate the structure of $S_r(G, s)/\langle \gamma_s \rangle$ by using the isomorphism $S_r(G, s)/\langle \gamma_s \rangle \cong S(G, s)/\langle \overline{\beta} \rangle$, where for a sandpile σ we write $\overline{\sigma} \equiv \sigma \mod \Delta_s \mathbb{Z}^{n-1}$. Recall that β_s is the sandpile such that $\beta_s(i) = d_{s,i}$ where s denotes the sink, and that V_0 is the group of vectors in $\mathbb{Z}^n$ with degree 0.

Theorem 12. *We have that*

$$S_r(G, s)/\langle \gamma_s \rangle \cong S(G, s)/\langle \overline{\beta_s} \rangle \cong V_0/\Delta \mathbb{Z}^n.$$

The meat of the proof for this Theorem is packaged in the following workhorse lemma.

Lemma 13. *The following are equivalent.*

1. $\sigma \equiv \eta \mod \Delta \mathbb{Z}^n$

2. $\tilde{\sigma} \equiv \tilde{\eta} \mod \Delta_s \mathbb{Z}^{n-1} + \mathbb{Z}\beta_s$

3. $(\tilde{\sigma} + e_s)^{\circ} \in (\tilde{\eta} + e_s)^{\circ} \oplus \langle \gamma_s \rangle$

Proof. (1 $\iff$ 2) Let $m = |\sigma| = |\eta|$. Recall that σ_k denotes the extension of σ to $\mathbb{Z}^n$ such that $|\sigma_k| = k$. We observe that $\tilde{\sigma} \in \tilde{\eta} \oplus \langle \gamma_s \rangle$ if and only if there is an $\tilde{\mathbf{x}} \in \mathbb{Z}^{n-1}$ and a $k \in \mathbb{Z}$ such that $\tilde{\sigma} = \tilde{\eta} - k\beta_s - \Delta_s \tilde{\mathbf{x}}$. If $\sigma = \eta - \Delta \mathbf{x}$, then

$$\sigma = \eta - \Delta \mathbf{x} = \eta - \Delta \begin{bmatrix} \tilde{\mathbf{x}} \\ 0 \end{bmatrix} - \Delta \begin{bmatrix} 0 \\ \mathbf{x}(s) \end{bmatrix} = \eta - \begin{bmatrix} \Delta_s \tilde{\mathbf{x}} \\ a \end{bmatrix} - \mathbf{x}(s)\mathbf{c}_s$$

where $\mathbf{c}_s$ denotes the column of Δ corresponding with the sink and a is the dot product of the nth row of Δ with $(\tilde{\mathbf{x}}, 0)$. Since $\beta_s(i) = -\mathbf{c}_s(i)$ for each $i \neq s$, it follows that $\tilde{\sigma} = \tilde{\eta} - \Delta_s \tilde{\mathbf{x}} + \mathbf{x}(s)\beta_s$. Note that if $\tilde{\sigma}$ and $\tilde{\eta}$ are recurrent, then we can drop the $\Delta_s \tilde{\mathbf{x}}$ and so write $\tilde{\sigma} \in \tilde{\eta} \oplus \langle \gamma_s \rangle$. Going the other way, we assume that $\tilde{\sigma} = \tilde{\eta} - k\beta_s - \Delta_s \tilde{\mathbf{x}}$ for some $k \in \mathbb{N}$ and $\tilde{\mathbf{x}} \in \mathbb{N}^{n-1}$. Note that if $\tilde{\sigma}$ and $\tilde{\eta}$ are recurrent, then this assumption is equivalent to the assumption that $\tilde{\sigma} \in \tilde{\eta} \oplus \langle \gamma_s \rangle$. Let σ' be the full configuration

$$\sigma' = \eta - \Delta \begin{bmatrix} \tilde{\mathbf{x}} \\ k \end{bmatrix}.$$

Then $\sigma'(i) = \tilde{\sigma}(i)$ for all $i \neq s$ and $|\sigma'| = |\eta|$. Since $\sigma(s)$ is determined by $|\sigma'|$, we have that $\sigma' = \sigma$.

(2 $\iff$ 3) As can easily be checked, $(e_s)_0 \in \Delta \mathbb{Z}^n$. Hence $\sigma + (e_s)_0 \equiv \eta + (e_s)_0 \mod \Delta \mathbb{Z}^n$ if and only if $\sigma \equiv \eta \mod \Delta \mathbb{Z}^n$. Since $(\tilde{\sigma} + e_s)^{\circ}$ and $(\tilde{\eta} + e_s)^{\circ}$ are recurrent, by the first part of the proof we conclude that $\sigma + (e_s)_0 \equiv \eta + (e_s)_0 \mod \Delta \mathbb{Z}^n$ if and only if $(\tilde{\sigma} + e_s)^{\circ} \in (\tilde{\eta} + e_s)^{\circ} \oplus \langle \gamma_s \rangle$, as desired. □

Proof. (Theorem 12) Define a map $\phi : S(G, s)/\langle \overline{\beta_s} \rangle \to V_0/\Delta \mathbb{Z}^n$ by taking $\overline{\sigma} \mod \langle \overline{\beta_s} \rangle$ to $\sigma_0 \mod \Delta \mathbb{Z}^n$. If $\overline{\sigma} \equiv \overline{\eta} \mod \langle \overline{\beta_s} \rangle$, then by Lemma 13 we have that $\sigma = \eta \mod \Delta \mathbb{Z}^n$, so that ϕ is well-defined. The equation $\sigma_0 + \eta_0 = (\sigma + \eta)_0$ is immediate from the definition, so that ϕ is a homomorphism. The map ϕ is also surjective, since for each $\sigma \in V_0$ there is a corresponding $\tilde{\sigma} \in \mathbb{Z}^{n-1}$, and $\phi(\overline{\tilde{\sigma}} \mod \langle \overline{\beta_s} \rangle) = \sigma \mod \Delta \mathbb{Z}^n$. We now show that ϕ is injective to complete the proof that ϕ is an isomorphism. Suppose that $\sigma = \eta \mod \Delta \mathbb{Z}^n$. Then by Lemma 13 we have that $\overline{\tilde{\sigma}} = \overline{\tilde{\eta}} \mod \langle \overline{\beta_s} \rangle$ and the theorem is proved. □

With this machinery we may give a more succinct proof of the following result than that found in [1] .

Lemma 14. *Let G be an Eulerian graph. Then for any vertex s we have that $S(G, s) \cong V_0/\Delta \mathbb{Z}^n$. In particuar, the sandpile group of G is independent of choice of sink up to isomorphism.*

Proof. Since the outdegree and indegree of v are equal we see that toppling v and all of its neighbors once leaves v with the same number of chips as before. Hence $\mathbf{1}$ is the primitive period vector for G. Then $\langle \overline{\beta_s} \rangle$ is trivial and the isomorphism $S(G, s) \cong V_0/\Delta \mathbb{Z}^n$ follows by Theorem 12 . □

Note that for Eulerian graphs, the greatest common divisor M of the numbers in $\{\mathcal{T}_G(v)|v \in G\}$ is $|S(G, s)|$ where s is any vertex.

2.2 Co-Eulerian Graphs

With the progress we've made we can now make short work of the proof of Theorem 7.

Proof. (1 $\implies$ 2) We prove the contrapositive. Assume there is a sink s such that $S(G, s) \neq \langle \gamma_s \rangle$, and fix the number of chips on G to be $m = m_G$. There are two distinct cyclic subgroups C_1 and C_2 of G such that $\tilde{\sigma}_{\max} \in C_1$. Choosing an $\eta \in C_2$, we remark that η_m stabilizes if and only if $\eta_m \equiv \sigma_{\max} \mod \Delta \mathbb{Z}^n$ (since $\sigma_{\max}$ is the only stable full configuration with degree m). This occurs if and only if $\eta \in \tilde{\sigma}_{\max} \oplus \langle \gamma_s \rangle = C_1$ by Lemma 13, so we see that η_m does not stabilize.

$(2 \implies 1)$ Suppose $\langle \overline{\beta} \rangle = S(G, s)$, and let η be a full configuration with $|\eta| = m = m_G$. Since $\tilde{\eta} = \tilde{\sigma}_{\max} \mod \Delta_s \mathbb{Z}^{n-1} + \mathbb{Z}\beta$, we have that $\eta \equiv \sigma_{\max} \mod \Delta \mathbb{Z}^n$ by Lemma 13, so that η stabilizes by the least action principle.

$(2 \iff 3)$ This follows from our previous assertion that $\text{ord}(\gamma_v) = |S(G, v)|/M$.

$(2 \iff 4)$ This follows from Theorem 12. $\qquad \square$

Theorem 12 gives us that the isomorphism of $V_0/\Delta\mathbb{Z}^n$ with the entire sandpile group $S(G, s)$ for Eulerian graphs is in a sense maximal. Theorem 15 characterizes the graphs for which $V_0/\Delta\mathbb{Z}^n$ is minimal, and this dichotomy supplies the reasoning behind our naming them "co-Eulerian" graphs. Note that this theorem gives that the halting problem can be solved quickly for co-Eulerian graphs.

3   Multi-Eulerian Tours

We again build off of the counting of cyclic subgroups of unicycles conducted by [2], generalizing the BEST theorem to general strongly directed multigraphs. Let π be the primitive period vector of a directed graph $G = (V, E)$. We define a **multi-Eulerian** tour of G to be a closed path that uses each directed edge (u, v) exactly $\pi(u)$ times.

Corollary 15. *Let $G = (V, E)$ be a strongly connected directed multigraph. Fix an edge $e \in E$ and let $tail(e) = w$. Let $\mathcal{T}_G(w)$ denote the number of oriented spanning trees in G rooted at w, and let $\epsilon(G, e)$ be the number of multi-Eulerian tours of G starting with the edge e. Then*

$$\epsilon(G, e) = \frac{\mathcal{T}_G(w) \prod_{v \in V} \pi(v) \prod_{v \in V} (d_v \pi(v) - 1)!}{\prod_{v \in V} (\pi(v)!)^{d_v}}.$$

We start with the an observation. Note that by Dhar's burning test, a sandpile σ on an Eulerian graph G is recurrent if and only if $(\sigma + \beta)^o = \sigma$, and this is equivalent to $\mathbf{1} \in \ker \Delta$. A converse is also true: if G is strongly connected and $\mathbf{1} \in \ker \Delta$, then G is Eulerian. This can be seen as follows: let σ be a recurrent sandpile, and assume that G is not Eulerian. Then there is a vertex v with outdegree greater than indegree. After toppling v, we topple all neighbors of v, and notice that we are left with less chips on v than $\sigma(v)$. Hence $\mathbf{1} \notin \ker \Delta$.

Proof. We define a graph $\tilde{G}$ to be G with every edge $e' \in E$ from u to v replaced by $\pi(u)$ edges from u to v. We first relate the number of spanning trees of $\tilde{G}$ to those of G. For edge $e' \in E$ from u to v, let $B_{e'}$ denote the set of $\pi(u)$ edges from u to v replacing e' in $\tilde{G}$, where π is the primitive period vector of $\tilde{G}$. Given a graph H, let $\mathcal{K}(H, w)$ denote the set of oriented spanning trees rooted at w. Define a map $\phi : \mathcal{K}(\tilde{G}, w) \to \mathcal{K}(G, w)$ taking a spanning tree in $\tilde{G}$ rooted at w to one in G rooted at w by mapping edges of $B_{e'}$ onto the edge e' in G. For each edge e' of a tree T_w in $\mathcal{K}(G, w)$ from u to v, there are $\pi(u)$ edges from u to v in $\tilde{G}$. Hence $|\phi^{-1}(T_w)| = \prod_{u \neq w} \pi(u)$. Given a spanning tree $T'_w \neq T_w$, there is an edge e' of T'_w that is not an edge of T_w. Each tree in $\phi^{-1}(T'_w)$ is mapped by ϕ to a tree with the edge e', and no tree in $\phi^{-1}(T_w)$ is mapped to a tree with edge e'. Thus $\phi^{-1}(T'_w) \cap \phi^{-1}(T_w) = \emptyset$, and it follows that $|\mathcal{K}(\tilde{G}, w)| = |\mathcal{K}(G, w)| \prod_{u \neq v} \pi(u)$.

Note that $\pi = \frac{1}{M} (|\mathcal{K}(G, v_1)|, |\mathcal{K}(G, v_2)|, ..., |\mathcal{K}(G, v_n)|)$ where $M = \gcd(|\mathcal{K}(G, v_1)|, |\mathcal{K}(G, v_2)|, ..., |\mathcal{K}(G, v_n)|)$. Since the quantity

$$|\mathcal{K}(G, v)| \prod_{u \neq v} \pi(u) \quad = \quad |\mathcal{K}(G, v)| \prod_{u \neq v} \frac{1}{M} |\mathcal{K}(G, u)|$$

is independent of choice of v, by Lemma (10) we have that the period vector for $\tilde{G}$ is $\mathbf{1}$, so that $\tilde{G}$ is Eulerian by the observation preceding this proof.

The BEST theorem gives us the number of Eulerian tours of $\tilde{G}$, so to proceed we relate this quantity to the number of multi-Eulerian tours of G. By fixing an ordering of $B_{e'}$ for each edge $e' \in E$, we may take an Eulerian tour of $\tilde{G}$ as follows: begin a multi-Eulerian tour of G, and for the ith traversal of each edge e' in G, traverse the ith edge in $B_{e'}$. Conversely, given an Eulerian tour of $\tilde{G}$, for each traversal of an edge in $B_{e'}$, where e' is an edge from u to v, we may traverse the edge e' in G. Since $|B_{e'}| = \pi(u)$, this determines a multi-eulerian tour of G. Hence for each multi-Eulerian tour $e_{i_1} e_{i_2} \cdots e_{i_t}$ there is a corresponding ordering $B_{e_{i_1}} B_{e_{i_2}} \cdots B_{e_{i_t}}$. Note that this multi-Eulerian tour of G is determined only up to the order in which the $B_{e'}$ are chosen (where $B_{e'}$ is chosen whenever an edge in $B_{e'}$ is traversed), so that if $\pi \neq \mathbf{1}$ there are more Eulerian tours of $\tilde{G}$ than there are multi-Eulerian tours of G. To calculate the precise ratio, we fix a multi-Eulerian tour of G, say $e_{i_1} e_{i_2} \cdots e_{i_t}$ as before. Observe that for each edge e_{i_j} in the tour, there are $|B_{e_{i_j}}|!$ corresponding Eulerian tours of $\tilde{G}$, as for each $B_{e_{i_j}}$ in $B_{e_{i_1}} B_{e_{i_2}} \cdots B_{e_{i_t}}$ we assign one edge in $B_{e_{i_j}}$ to be the edge taken in the tour (each edge is only assigned once, as each edge is traversed once in an Eulerian tour). Hence there are

$$\prod_{e \in E} |B_e|! = \prod_{v \in V} (\pi(v)!)^{d_v}$$

Eulerian tours of $\tilde{G}$ starting at w for each multi-Eulerian tour of G starting at w. If we now fix a multi-eulerian tour $e e_{i_2} \cdots e_{i_t}$ of G starting from e and an eulerian tour $\tilde{e} B_{e_{i_2}} \cdots B_{e_{i_t}}$ of $\tilde{G}$ starting from an edge $\tilde{e} \in B_e$, we see that for each edge e_{i_j} with $\text{tail}(e_{i_j}) \neq w$ we have that there are $|B_{e_{i_j}}|!$ corresponding Eulerian tours of $\tilde{G}$. For the $d_w - 1$ edges $e' \neq e$ with $\text{tail}(e') = w$ there are also $|B_{e'}|!$ corresponding Eulerian tours of $\tilde{G}$, while there are only $(|B_e| - 1)!$ that correspond with e (since the first edge $\tilde{e}$ is fixed). Hence

$$\epsilon(G, e) = \frac{\epsilon(\tilde{G}, \tilde{e})}{D}$$

where

$$
\begin{aligned}
D &= (\pi(w)!)^{d_w - 1} (\pi(w) - 1)! \prod_{v \neq w} (\pi(v)!)^{d_v} \\
&= \frac{\prod_{v \in V} (\pi(v)!)^{d_v} (\pi(w)!)^{d_w - 1} (\pi(w) - 1)!}{(\pi(w)!)^{d_w}} \\
&= \frac{\prod_{v \in V} (\pi(v)!)^{d_v}}{\pi(w)}.
\end{aligned}
$$

$\square$

By Corollary 4.10 of [1], the number of Eulerian tours in $\tilde{G}$ starting with $\tilde{e}$ is

$$\epsilon(\tilde{G}, \tilde{e}) = \mathcal{T}_{\tilde{G}}(w) \prod_{v \in V} (d_v \pi(v) - 1)! = \mathcal{T}_G(w) \prod_{v \neq w} \pi(u) \prod_{v \in V} (d_v \pi(v) - 1)!.$$

It follows that

$$
\begin{aligned}
\epsilon(G, e) &= \frac{\pi(w) \epsilon(\tilde{G}, \tilde{e})}{\prod_{v \in V} (\pi(v)!)^{d_v}} \\
&= \frac{\pi(w) \mathcal{T}_G(w) \prod_{v \neq w} \pi(v) \prod_{v \in V} (d_v \pi(v) - 1)!}{\prod_{v \in V} (\pi(v)!)^{d_v}} \\
&= \frac{\mathcal{T}_G(w) \prod_{v \in V} \pi(v) \prod_{v \in V} (d_v \pi(v) - 1)!}{\prod_{v \in V} (\pi(v)!)^{d_v}}.
\end{aligned}
$$

4 Computational Aspects of Chip-Firing

4.1 Methods for checking if a full configuration stabilizes

The ideas developed thus far allow for more sophisticated methods for checking if a full configuration stabilizes. We will first need an efficient way to calculate when two recurrent sandpiles σ and η are equivalent modulus $\langle \gamma_s \rangle$. This occurs when

$$\sigma - \Delta_s \mathbf{x} + k\beta_s = \eta$$

for a nonnegative integer k and integral $\mathbf{x}$. We can search for an integral $\mathbf{x}$ by inverting Δ_s and looking for k such that

$$a_1 + kb_1$$
$$a_2 + kb_2$$
$$\vdots$$
$$a_n + kb_n$$

are all integral for the rational a_i and b_i, $1 \leq i \leq n$, that result from multiplying $\eta - \sigma - k\beta_s$ by $-(\Delta_s)^{-1}$. Clearing denominators, this becomes a system of linear congruences, and we can check for the existence of a solution in polynomial time using the Chinese Remainder Theorem.

Recall that η_k is the full configuration such that $\eta_k(i) = \eta(i)$ for $i \neq s$ and $|\eta_k| = k$, and that e_s is the identity of $S_r(G, s)$. Consider a full configuration σ with $|\sigma| = k$ and recurrent restriction $\tilde{\sigma}$, and let ξ be the recurrent sandpile $(\tilde{\sigma} + e_s)^\circ$. Since $\sigma = \eta_k \mod \Delta \mathbb{Z}^n$ if and only if $(\eta + e_s)^\circ \in C_\xi$ by Lemma 13, by the least action principle we have that σ stabilizes if and only if there is some recurrent configuration $\eta \in C_\xi$ such that $\eta_k(s) < d_s$. Since $\eta_k(s) + |\eta| = k$, the latter inequality is equivalent to the condition that $|\eta| > k - d_s$. Using this information, we develop a method for checking if σ stabilizes.

We calculate the primitive period vector π of G, and if $\pi(i)$ is small for some $i \in [1, 2, ..., n]$ then we take $s = v_i$ as the sink and obtain a corresponding γ_s with $\text{ord}(\gamma_s) = \pi(i)$ small. We can then quickly generate C_ξ, which by the recent discussion is enough to determine whether or not σ stabilizes. Note that, in particular, $\pi(i) = 1$ for all i in an Eulerian graph, so that our methods coincide with the work by Tardos [11] showing that the halting problem can be solved in polynomial time for Eulerian graphs.

If $\text{ord}(\gamma_s)$ is large for all choices of sink s but the number of cosets M of $\langle \gamma_s \rangle$ is small, we can make use of the procedure for checking if two recurrent sandpiles are in the same coset to determine if σ stabilizes. Since we are looking for any $\eta \in C_\xi$ such that $|\eta| > k - d_s$, we start with $\eta = \tilde{\sigma}_{\max}$ and work our way down, checking for all η such that $|\eta| = |\tilde{\sigma}_{\max}| - 1$, then for all η such that $|\eta| = |\tilde{\sigma}_{\max}| - 2$, and so on, until for all cyclic subgroups C of $S(G, s)$ we have checked for some $\eta \in C$. If after all this we have not found an η such that $|\eta| > k - d_s$, we conclude that σ does not stabilize. Note that M gives a heuristic for the number of checks we need to make: in particular, for M a constant that does not depend on n and a certain class of graphs, we have that we will require only a constant number of checks before we have hit upon each of the M cyclic subgroups of $S(G, s)$. In particular, our analysis coincides with the This class of graphs is defined by having the property that the degrees of recurrent configurations do not correlate too strongly with the cyclic subgroups – for instance, our method can be thwarted by a graph with a sandpile group having a cyclic subgroup that contains all of the highest degree sandpiles.

We do not even have a heuristic for classes of graphs with large M and large $|S(G, v)|/M$ for each vertex v. One example of such a class is the graph where we put four edges from v_i to v_{i+1} and six from v_{i+1} to v_i. Here $M = 2^n$ and $\min \{\mathcal{T}_G(v)|v \in G\} = 4^n$. It turns out that in this particular case we actually can use the ideas developed up to this point in clever ways to solve the halting problem on this class of graphs – in the interest of brevity we leave this to the reader. For more complicated graphs there may not be such tricks.

4.2 Reduction from an NP-Hard problem

We've shown that if M or $|S(G, v)|/M$ is appropriately small for some vertex v, then the halting problem can be tractable. We proceed by developing a reduction from an NP-hard problem to the halting problem for chip firing. However, in order to conclude that the halting problem is NP-Hard, we would require that the reduction be to a version of the halting problem such that the edges of the graph can be represented by a polynomial number of bits. The question remains open as to whether or not this is achievable.

We begin by considering a closely related chip firing game (sometimes referred to as the *dollar game*) that goes as follows: given an initial full configuration we ask if there is some sequence of topplings that results in the number of chips on each node being at least zero. Chip configurations in this context are generally referred to as divisors in the literature.

In this context, we translate the question of whether or not we can take every node out of debt to the following: given a sub-lattice L of V_0 of full rank (that is, rank $n - 1$), we say that $r(D) = -1$ if there is no $E \in \mathbb{Z}^n$ with $E \geq 0$ such that $D - E \in L$. This means we cannot bring D out of debt for all nodes. We will make use of the following Theorem by [5].

Theorem 16 ([5]). *For an arbitrary full rank sub-lattice L of A_n, the problem of deciding if $r(D) = -1$ given a point $D \in \mathbb{Z}^n$ and a basis of L is NP-hard.*

We would like to translate this to a context more closely related to the halting problem, where we have a Laplacian to work with.

Lemma 17. *For any full rank sub-lattice L of V_0 there exists a graph with Laplacian Δ such that $\Delta \mathbb{Z}^n = L$, and Δ can be computed in time polynomial in n.*

This Lemma follows almost immediately from the proof of Theorem 4.11 in ([4]), which says that for any lattice L of $\mathbb{Z}^n$ having rank n, there exists a graph whose reduced Laplacian lattice Δ_s is such that $\Delta_s \mathbb{Z}^{n-1} = L$. The proof is constructive, and is paired with an algorithm that performs integer column operations to transform L into Δ_s in polynomial time. The algorithm terminates with $|c_1| > 0$ and $|c_j| = 0$ for $j > 1$ where c_i denotes column i of Δ_s.

Proof. Let M be an integer matrix such that $M\mathbb{Z}^n = L$. We show that there exists a Laplacian Δ of a graph with the same integer column span as M. We remark that necessary and sufficient conditions for a matrix A to be a full graph laplacian are that the degree of the columns of A be zero, the diagonal entries of A be nonpositive, and the offdiagonal entries of A be nonnegative.

Since M has rank $n - 1$, there is a sequence of legal column operations that puts the last column of M to zero. More explicitly, we can use the Euclidean algorithm, adding one column to another, on the upper triangular portion of each of the first rows $n - 1$ rows of M in turn. Once we are done with this, each of the entries of the first $n - 1$ rows above the diagonal are zero. Since the degree of the last column of the resulting matrix is zero, we conclude that the last column is all zeros. With this done we apply the algorithm of Theorem 4.11 of ([4]) to the matrix M' gotten by discarding the bottom row and rightmost column of M, where the column operations specified by the algorithm are extended to column operations of M. Note that M' is an $(n - 1) \times (n - 1)$ matrix of rank $n - 1$, so that the conditions of Theorem 4.11 are fulfilled. Since all of the columns of M have degree 0, the integer column operations do not change the degree of the columns of M. Letting c_i denote the ith column of M', we have that $|c_1| > 0$ and $|c_j| = 0$ for $j > 1$. It follows that the bottom left entry of M is now negative, while the remaining entries in the bottom row are zero. Since M' is a reduced laplacian, we have that M is the full laplacian of a graph. $\square$

We now observe a useful connection between divisors and sandpiles:

Lemma 18. *Given a full rank sub-lattice L of A_n and a full configuration σ, we have that $r(\sigma) \geq 0$ if and only if there is a graph laplacian Δ with $\Delta \mathbb{Z}^n = L$ and a vector $\mathbf{x} \geq 0$ such that $\sigma_{max} - \sigma - \Delta \mathbf{x} \leq \sigma_{max}$.*

Proof. By Lemma 17 we can find a laplacian Δ of some graph such that $\Delta \mathbb{Z}^n = L$. Then $r(\sigma) \geq 0$ if and only if there is a firing script x such that $\sigma - \Delta \mathbf{x} \geq 0$. We observe that the condition $\sigma - \Delta \mathbf{x} \geq 0$ is equivalent to the condition that $\sigma - \sigma_{\max} - \Delta \mathbf{x} \geq -\sigma_{\max}$ and negating both sides yields $\sigma_{\max} - \sigma - \Delta(-\mathbf{x}) \leq \sigma_{\max}$. Since $\Delta(-\mathbf{x} + \mathbf{p}) = \Delta$ for $\mathbf{p}$ a period vector, we can find a period vector $\mathbf{p}$ such that $-\mathbf{x} + \mathbf{p} \geq 0$ and subtract $\Delta \mathbf{p}$ from both sides to get a nonnegative firing vector. $\square$

We would like to say that, given an oracle for solving the halting problem on finite directed graphs, we can solve whether or not $r(\sigma) = -1$. This is not the case in general, since we could have $(\sigma_{\max} - \sigma)(i) < 0$ for some i, and the halting problem oracle can only take as input nonnegative full configurations. This problem can be averted if we can find a nonnegative configuration η such that $\eta = \sigma_{\max} - \sigma \mod \Delta \mathbb{Z}^n$, as stabilization of this sandpile implies that $r(\sigma) \geq 1$. If we cannot find such an η, it turns out not to be a problem as computing $r(\sigma)$ in this case is trivial.

Lemma 19. *Let Δ be an $n \times n$ graph laplacian of rank $n-1$ with $\sigma \in \mathbb{Z}^n$, and suppose that for all $\mathbf{x} \in \mathbb{Z}^n$ there exists an i such that $(\sigma_{max} - \sigma - \Delta \mathbf{x})(i) < 0$. Then $r(\sigma) \geq 0$.*

Proof. The condition $(\sigma_{\max} - \sigma - \Delta \mathbf{x})(i) < 0$ is equivalent to the condition that $\sigma(i) + \Delta \mathbf{x} > \sigma_{\max}$, so that the condition of the Lemma gives us that for any sequence of topplings, there will always be a vertex v_i with greater than $\sigma_{\max}(i)$ chips. Since we may topple such vertices indefinitely, we suppose that we do so, letting $H \subseteq V$ denote the set of vertices that are toppled infinitely many times. To be more precise, on each step we topple every currently active vertex, and we continue taking steps indefinitely. Since G is finite, we have that $H \neq \emptyset$. Suppose that $H \neq V$; we will draw a contradiction. Note that G is strongly connected: this follows at once from the matrix tree theorem, as choosing any vertex v as a sink yields a reduced laplacian Δ_v of rank $n-1$, so that there is at least one spanning tree rooted at v. Since G is strongly connected, there is a vertex $v \in V - H$ and a vertex $w \in H$ with $d_{wv} > 0$. Since w topples infinitely many times in the process, so too does v, giving us our contradiction. Hence $H = V$. We can thus carry out our toppling procedure until every vertex has become active and subsequently toppled (note that in our procedure we never topple inactive vertices). At this point we have reached the desired nonnegative configuration, so that $r(\sigma) \geq 0$.

$\square$

We can thus restrict the configurations σ in Lemma 16 to those such that $\sigma_{\max} - \sigma - \Delta \mathbf{x} \geq 0$ for some $\mathbf{x}$ and retain the NP-Hardness. Lemmas 17 through 19, along with Theorem 16 give us the following.

Conjecture 20. *The problem of computing if $r(D) = -1$ reduces to the halting problem for finite, directed, strongly connected graphs.*

Proof. Given an arbitrary full rank sub-lattice L of A_n and a full configuration $\sigma \in \mathbb{Z}^n$ such that $\sigma_{\max} - \sigma - \Delta \mathbf{x} = \eta \geq 0$ for some $\mathbf{x}$, we can decide if $r(\sigma) = -1$ by computing the full laplacian Δ with $\Delta \mathbb{Z}^n = L$ and checking to see if η is equivalent to a stable configuration modulo $\Delta \mathbb{Z}^n$. The least action principle gives us that this is equivalent to checking if η is stabilizable, and so we have a reduction from the dollar game problem to the halting problem. $\square$

One issue remains: the Laplacian Δ we compute may have very large entries, corresponding with many edges existing between vertices in the underlying graph. For our efforts to bear fruit, we will need to be able to encode these edges with a polynomial number of bits. To make this precise, we define the following.

Problem. The **halting problem for chip-firing on finite, strongly connected graphs with $O(2^n)$ edges between each pair of vertices.**

Input: A finite, strongly connected multigraph G with $O(2^n)$ edges between each pair of vertices and a full chip configuration σ with $\sigma \geq 0$.

Output: A decision on whether or not σ stabilizes.

The question remains: does the NP-Hard problem of computing if $r(D) = -1$ reduce to this, or a similar, more restrictive halting problem?

References

[1] Alexander E. Holroyd, Lionel Levine, Karola Mészáros, Yuval Peres, James Propp and David B. Wilson, Chip-Firing and Rotor-Routing on Directed Graphs, *In and Out of Equilibrium 2, Progress in Probability* (2008), Vol. 60, 331–364

[2] Trung Van Pham, Period of rotor-router operation, arXiv:1403.5875v3 (2014).

[3] Anders Björner and László Lovász, Chip-Firing games on directed graphs, *J. Algebraic Combin* (1992) Vol 1. 305-328

[4] D. Perkinson, J. Perlman, and J. Wilmes, Primer for the algebraic geometry of sandpiles. *Contemporary Mathematics* (2014) Vol. 605.

[5] O. Amini and M. Manjunath, Riemann-Roch for Sublattices of the Root Lattice A_n, *Electronic Journal of Combinatorics* (2010), 17(1)

[6] Aldous, D. The random walk construction of uniform spanning trees and uniform labelled trees. *SIAM J. Disc. Math.* 3, (1990) 450–465.

[7] Broder, A. Generating random spanning trees. In *Symp. Foundations of Computer Sci.*, IEEE, New York, (1989) 442–447.

[8] John Harris, Jeffry Hirst, and Michael Mossinghoff, *Combinatorics and Graph Theory*, Undergraduate Texts in Mathematics (2nd ed.), Springer (2008).

[9] Benjamin Bond and Lionel Levine. Abelian Networks: Foundations and Examples. arXiv:1309.3445 (2014)

[10] Anders Björner and László Lovász, Chip-Firing games on directed graphs, *European Journal of Combinatorics* (1991) Vol. 12 Issue 4, 283–291

[11] Gábor Tardos. Polynomial bound for a chip firing game on graphs. *Siam J. Disc. Math.* (1988) Vol. 1, No. 3.