Concrete and abstract structure of the sandpile group for thick trees with loops

William Chen and Travis Schedler

July 19, 2021

Abstract

We answer a question of László Babai concerning the abelian sandpile model. Given a graph, the model yields a finite abelian group of recurrent configurations which is closely related to the combinatorial Laplacian of the graph. We explicitly describe the group elements and operations in the case of thick trees with loops—that is, graphs which are obtained from trees by setting arbitrary edge multiplicities and adding loops at vertices. We do this both concretely (by describing the so-called recurrent and identity configurations) and abstractly (by computing the group's abstract structure), and define maps identifying the two.

1 Introduction

The *abelian sandpile model* (ASM) was defined in [8] for any graph, generalizing the case of a grid from [3]. This model is a prime example of *self-organized criticality* [3] which has transformed the understanding of how complexity arises in nature (see [18] for a summary of applications to modelling earthquakes, forest fires, landslides, etc.) The ASM is related to the chip-firing game, introduced by Spencer [16], and later modified by Björner, Lovász, and Shor [5]. For general references on the abelian sandpile model, see e.g. [1, 9].

We will only consider undirected graphs. An (undirected) abelian sandpile model begins with any connected undirected graph (with arbitrary edge multiplicities) with vertex set V and edge set E, together with a distinguished vertex $s \in V$ called the *sink*. Then, a *configuration* of the abelian sandpile model is an element of $\mathbb{Z}_{\geq 0}^{V_0}, V_0 := V \setminus \{s\}$, which assigns a nonnegative integer to each vertex except the sink, considered the number of grains of sand at that vertex. A configuration is *stable* if the number of grains of sand on each vertex is less than the degree of the vertex (otherwise it is unstable). Then, one obtains a dynamical system on the space of configurations, where at each time step, each unstable vertex sends one grain of sand along each incident edge. Sending grains of sand to the sink merely reduces the total number of grains. Because every vertex has a path to the sink, it turns out that this yields a well-defined *stabilization* of any configuration in finite time. The system can remain in motion by continually adding grains (e.g., at random vertices).

One basic question about any dynamical system is to find “recurrent configurations.” For sandpiles, Dhar [8] defined these to be stable configurations that can be obtained from any configuration by adding grains and stabilizing. Dhar showed that they form an abelian group (under adding and stabilizing), called “the sandpile group,” whose abstract group structure is $\mathbb{Z}^n/\Delta\mathbb{Z}^n$, where Δ is the matrix obtained from the Laplacian of the graph by deleting the row and column corresponding to the sink.

However, the concrete structure (in terms of configuration on the graph) of these recurrent configurations turns out to be quite complicated and interesting. For example, Creutz’s paper [6] displays intricate fractal patterns of the identity element of this group in the case of a square grid, which was further studied in [7, 13].

László Babai suggested that the authors study the sandpile group for “thick” graphs: graphs which may have large edge multiplicities, but whose combinatorial structure is simple. In this note, we consider *thick trees with loops*: graphs which become trees when all positive edge multiplicities are dropped to one, and all loops are discarded. We concretely describe the group elements and operations in these cases, compute the abstract group structure, and relate the two using recursive formulas. To our knowledge, this is the first result giving the complete, explicit structure of the sandpile group in any class of undirected graphs. In [10] (the grid) and [2, 12, 17] (other graphs), information about the abstract group structure for more complicated graphs was given.

The note is organized as follows: in Section 2 we recall the necessary background. In Section 3 we state our main theorem and its corollaries. In Section 4 we prove the theorem.

2 The abelian sandpile model

We review the theory of the (undirected) abelian sandpile model (ASM), following [1].

Let A be a finite connected undirected graph with arbitrary edge multiplicities, allowing loops. Precisely, $A = (V, E, g)$, where V is the set of vertices, E is the set of edges, and $g : E \to V^{(2)}$ is the incidence map. Here $V^{(2)}$ is the set of one- and two-element subsets of V (i.e., unordered pairs of vertices). We have $g(e) = \{i, j\}$ where i, j are the endpoints of e. We require that A be connected, i.e., there is a path of edges connecting any two vertices.

Definition 2.1. An (undirected) ambient space is a collection (A, s) where A is an undirected, connected graph with vertex set V and edge set E (allowing multiple edges and loops), and $s \in V$ is a distinguished vertex, called the **sink**. All other vertices are called **ordinary**. Let $V_0 := V \setminus \{s\}$ be the set of ordinary vertices.

Notation 2.2. We will use bold letters (e.g. $\boldsymbol{u}$) to denote vectors, usually in $\mathbb{Z}_{\geq 0}^{V_0}$.

Definition 2.3. A **configuration** of the ASM associated to (A, s) is a vector $\boldsymbol{u} \in \mathbb{Z}_{\geq 0}^{V_0}$, i.e., an assignment of a nonnegative integer u_i to each ordinary vertex i (called its **height**).

Definition 2.4. For any $i, j \in V$, define the **edge multiplicity** to be $e_{i,j} := \#\{a \in E : g(a) = \{i, j\}\}$.

We may now define the notion of **stability** of a configuration:

Definition 2.5. Given a configuration $\boldsymbol{u}$, a vertex $i \in V_0$ is called **stable** if $u_i < \deg(i)$. Otherwise i is called **unstable**. A configuration is called **stable** if all vertices are stable.

Definition 2.6. If $i \in V_0$ is unstable for $\boldsymbol{u}$, then define the state $\tau_i(\boldsymbol{u})_i$ by $\tau_i(\boldsymbol{u})_i := u_i - \deg(i) + e_{i,i}$ and $\tau_i(\boldsymbol{u})_j := u_j + e_{i,j}$ for $j \neq i$. Passing from $\boldsymbol{u}$ to $\tau_i(\boldsymbol{u})$ is called "toppling the vertex i."

Given any unstable configuration, one may continually perform such topplings τ_i until a stable configuration results, which we call the **stabilization**.

Theorem 2.7. *[8, 5] (cf. also [4]) Given any ambient space (A, s) and any configuration $\boldsymbol{u}$ of its ASM, then there exists a unique stable configuration $\sigma(\boldsymbol{u})$, which satisfies $\sigma(\boldsymbol{u}) = \tau_{i_1}\tau_{i_2}\cdots\tau_{i_m}(\boldsymbol{u})$ for some sequence of topplings $\tau_{i_1}, \tau_{i_2}, \ldots, \tau_{i_m}$.*

(The theorem rests on an application of the Diamond Lemma, since if two vertices are unstable at once, one may topple them in either order with the same result.)

Definition 2.8. For each configuration $\boldsymbol{u}$, we denote by $\sigma(\boldsymbol{u})$ its unique stabilization.

With this in mind, one may conclude that the set of stable configurations form a monoid under vertexwise addition:

Definition 2.9. [8] Let M be the set of stable configurations. Define the operation $\oplus$ on M by $\boldsymbol{u} \oplus \boldsymbol{v} := \sigma(\boldsymbol{u} + \boldsymbol{v})$, where $+$ is vector addition.

Lemma 2.10. *[8] The operation $\oplus$ is commutative and associative, making $(M, \oplus)$ a commutative monoid, called the* ***sandpile monoid****.*

Even the abstract structure of the sandpile monoid (without discussing concrete configurations) can be quite complicated. However, Dhar [8] noticed that the subset of "recurrent" configurations has a much simpler abstract structure:

Definition 2.11. A configuration $\boldsymbol{u}$ is called **recurrent** if, for all configurations $\boldsymbol{v}$, there exists a configuration $\boldsymbol{w}$ such that $\boldsymbol{v} \oplus \boldsymbol{w} = \boldsymbol{u}$.

Proposition 2.12. *[8] The set of recurrent configurations, G, is an abelian group under $\oplus$, called the* ***sandpile group****.*

(Actually, as noticed in [1], the above is true for any finite commutative monoid, if one replaces G by the unique minimal ideal.) In [8] (elaborated in [10]) there is a surprisingly simple formula for the abstract structure of this group:

Definition 2.13. The **toppling matrix** $\Delta = (\Delta_{ij})_{i,j \in V_0}$ is given by

$$\Delta_{ij} = \begin{cases} \deg(i) - e_{i,i}, & \text{if } i = j, \\ -e_{i,j}, & \text{otherwise.} \end{cases} \tag{2.0.1}$$

Note that any two configurations which have the same stabilization are related by an element in the row span of Δ; also, adding a large enough multiple of the sum of the rows Δ_i must give a unique stabilization. Using this, one has the

Definition 2.14. Let $\Lambda := \langle \boldsymbol{\Delta}_i \rangle_{i \in V_0}$ be the lattice spanned by the rows $\boldsymbol{\Delta}_i$ of the toppling matrix.

Theorem 2.15. *[8, 1] The natural map $\mathbb{Z}_{\geq 0}^{V_0} \twoheadrightarrow G$, $\boldsymbol{c} \mapsto \sigma(\boldsymbol{c} + \boldsymbol{e}^A)$ descends to an isomorphism $\mathbb{Z}^{V_0}/\Lambda \cong G$. In particular, $\#(G) = \det(\Delta)$.*

(Note: $G \cong \mathbb{Z}^{V_0}/\Lambda$ and $\#(G) = \det(\Delta)$ are due to [8]; the map $\boldsymbol{c} \mapsto \sigma(\boldsymbol{c} + \boldsymbol{e}^A)$ was pointed out in [1].)

Dhar found an algorithm to test for recurrence, of linear time in the combinatorial size of A (the number of edges counted without multiplicity plus the number of vertices):

Definition 2.16. Let $\boldsymbol{\beta} \in \mathbb{Z}_{\geq 0}^{V_0}$ be given by $\boldsymbol{\beta} = \sum_{i \in V_0} \boldsymbol{\Delta}_i$. That is, $\boldsymbol{\beta}_i = e_{i,s}$.

We consider adding $\boldsymbol{\beta}$ to be "toppling of the sink". One has the

Algorithm 2.17 (Burning algorithm)**.** [8, 15] To test whether a stable configuration $\boldsymbol{u}$ is recurrent, stabilize $\boldsymbol{u} + \boldsymbol{\beta}$. Each vertex can topple at most once in the stabilization process, and $\boldsymbol{u}$ is recurrent iff every vertex topples (equivalently, iff $\boldsymbol{u} \oplus \boldsymbol{\beta} = \boldsymbol{u}$).

The term "burning algorithm" (with which each vertex that topples in the process is considered "burned") was introduced in [14]. The algorithm above was discovered by [8], and the proof follows from [8] and [15].

3 The sandpile group for thick trees

Definition 3.1. Following L. Babai, we call an undirected graph T a **thick tree** if the underlying simple graph T' obtained by reducing all nonzero edge multiplicities to 1 is a tree (i.e., a connected graph without cycles). A vertex i is called a **leaf** if $\deg_{T'} i = 1$ and $i \neq s$.

Definition 3.2. An undirected graph T is called a **thick tree with loops** if, after removing all loops (dropping $e_{i,i}$ to zero for all $i \in V$), the obtained graph is a thick tree.

Recall that a graph is a tree iff, for any two vertices, there exists a unique path from one to the other. Let $i, j \in V$. We define a partial ordering $\succeq$ on V, where $i \succeq j$ if j is on the unique path of T' from i to the sink. By definition, this means that $j \succeq s$ for all $j \in V$. Also, $i \succ j$ if the condition $i \neq j$ is added. Vertex $p(j)$ is the **parent** of j if $p(j) \succ j$ and $p(j)$ is adjacent to j.

For a thick tree with loops, T, we define $\succeq$ using the underlying tree T'.

We have the following main result, whose proof is the content of the next section.

Notation 3.3. For any vertex $i \in V_0$, let $\boldsymbol{\delta}_i$ be the elementary vector given by $(\boldsymbol{\delta}_i)_j = \delta_{ij} = 1$ if $i = j$ and 0 otherwise.

Theorem 3.4. *The sandpile group* $(G \subset \mathbb{Z}_{\geq 0}^{V_0}, \oplus)$ *for a thick tree* T *with loops can be characterized as follows (recall Notation 3.3):*

(i) $G = \{\boldsymbol{u} \in \mathbb{Z}_{\geq 0}^{V_0} : \forall j \in V_0, \deg(j) > u_j \geq \deg(j) - e_{j,p(j)}\}$.

(ii) The map $\phi : G \to \prod_{j \in V_0} \mathbb{Z}/e_{j,p(j)}$, $\boldsymbol{u} \mapsto \sum_{j \in V_0} \boldsymbol{\delta}_j \big(\sum_{i \succeq j} u_i \pmod{e_{j,p(j)}}\big)$ *is a group isomorphism. In particular, the abstract group structure does not depend on which vertex is chosen to be the sink.*

(iii) The map ϕ^{-1} *may be expressed recursively as follows: For any* $j \in V_0$,

$$\phi^{-1}(\boldsymbol{v})_j = \begin{cases} \tilde{v}_j, & \text{if } j \text{ is a leaf,} \\ f_j\Big(\tilde{v}_j - \sum\limits_{i \succ j} \phi^{-1}(\boldsymbol{v})_i\Big), & \text{otherwise,} \end{cases} \tag{3.0.2}$$

where for any $m \in \mathbb{Z}/p$, *we let* $\tilde{m} \in \{0, 1, \ldots, p-1\}$ *be its preferred representative, and for any* $j \in V_0$, $f_j : \mathbb{Z} \to \{\deg(j) - e_{j,p(j)}, \ldots, \deg(j) - 1\}$ *is given (uniquely) by* $f_j(m) \equiv m \pmod{e_{j,p(j)}}$.

Corollary 3.5. *Let* $\boldsymbol{u}$ *be any configuration. Then the associated recurrent configuration* $\sigma(\boldsymbol{u} + \boldsymbol{e}^T)$ *is given recursively by*

$$\sigma(\boldsymbol{u} + \boldsymbol{e}^T)_j = \begin{cases} u_j - \lfloor u_j / e_{j,p(j)} \rfloor e_{j,p(j)} & \text{if } j \text{ is a leaf,} \\ u_j + \lambda_j e_{j,p(j)} + \sum_{i \succ j} (u_i - \sigma(\boldsymbol{u} + \boldsymbol{e}^T)_i) & \text{otherwise,} \end{cases} \tag{3.0.3}$$

where $\lambda_j = \lfloor \big(\deg(j) - u_j - \sum_{i \succ j} (u_i - \sigma(\boldsymbol{u} + \boldsymbol{e}^T)_i) - 1\big) / e_{j,p(j)} \rfloor$. *If* $\sigma(\boldsymbol{u})$ *is already recurrent, we can delete the* $\boldsymbol{e}^T$*'s.*

Proof. Simply apply the map ϕ and then the map ϕ^{-1}; we then expand f_j. □

Corollary 3.6. *The identity configuration* $\boldsymbol{e}^T$ *is given by*

$$e_j^T = \begin{cases} 0 & \text{if } j \text{ is a leaf,} \\ \lambda_j e_{j,p(j)} - \sum_{i \succ j} e_i^T & \text{otherwise,} \end{cases} \tag{3.0.4}$$

where $\lambda_j = \lfloor \big(\deg(j) + \sum_{i \succ j} e_i^T - 1\big) / e_{j,p(j)} \rfloor$.

Proof. Specialize Corollary 3.5 to the case of $\boldsymbol{u} = 0$. (Alternatively, specialize (3.0.2) to the case of $\boldsymbol{v} = 0$ and expand f_j.) □

4 Proof of Theorem 3.4

We divide Theorem 3.4 into Propositions 4.1, 4.3, and 4.4, corresponding to (refinements of) parts (i), (ii), and (iii), respectively.

Proposition 4.1. *In a thick tree T with loops, the following statements are equivalent:*

(i) Configuration $\boldsymbol{u}$ is recurrent.

(ii) For all $j \in V_0$, $\deg(j) > u_j \geq \deg(j) - e_{j,p(j)}$.

Proof. Take any stable configuration $\boldsymbol{u}$. We apply Algorithm 2.17. This means that we try to stabilize $\boldsymbol{u} + \boldsymbol{\beta}$, and test whether or not every vertex topples. Since $\boldsymbol{\beta}$ adds sand only to the vertices adjacent to the sink, one easily sees that no vertex can topple before its parent. We can view adding $\boldsymbol{\beta}$ as having "the sink topple," and thus inductively, a configuration is recurrent iff every vertex can topple once its parent topples. We conclude that every vertex fires iff, for every vertex j, the amount of chips needed to topple $(\deg(j) - u_j)$ is less than or equal to the amount of chips it inductively receives from its parent $(e_{i,p(i)})$. □

Lemma 4.2. *The lattice Λ can be characterized as*

$$\Lambda = \{\boldsymbol{v} : e_{j,p(j)} \mid \sum_{i \succeq j} v_i, \forall j \in V_0\}. \tag{4.0.5}$$

Proof. First, we show the inclusion $\subseteq$. It suffices to show that, if $\boldsymbol{v} = \boldsymbol{\Delta}_k$ for any $k \in V_0$, then $e_{j,p(j)} \mid \sum_{i \succeq j} v_i$ for any $j \in V_0$.

For $k = p(j)$, one has

$$\sum_{i \succeq j} v_i = -e_{j,p(j)} \equiv 0 \pmod{e_{j,p(j)}}. \tag{4.0.6}$$

For $k = j$, one has

$$\sum_{i \succeq j} v_i = e_{j,p(j)} \equiv 0 \pmod{e_{j,p(j)}}. \tag{4.0.7}$$

For all $k \in V_0 \setminus \{j, p(j)\}$, one has

$$\sum_{i \succeq j} v_i = 0. \tag{4.0.8}$$

The result is proved.

Next, we show the inclusion $\supseteq$. For any $j \in V_0$, consider the sum

$$\sum_{i \succeq j} \boldsymbol{\Delta}_i = -e_{j,p(j)}(\boldsymbol{\delta}_{p(j)} - \boldsymbol{\delta}_j). \tag{4.0.9}$$

A simple inductive argument on V_0 shows that these elements span the desired space $W = \{\boldsymbol{v} : e_{j,p(j)} \mid \sum_{i \succeq j} v_i, \forall j \in V_0\}$. To do this, note that if j is a leaf, $\Delta_{jj} = e_{j,p(j)}$, and one may express all elements of W as a sum of such $\boldsymbol{\Delta}_j$'s and vectors which are zero in the leaf components. □

As a consequence, we deduce the

Proposition 4.3. *One has an isomorphism* $\bar{\phi} : \mathbb{Z}/\Lambda \cong \prod_{j \in V_0} \mathbb{Z}/e_{j,p(j)}$ *which descends from the map*

$$\phi : \mathbb{Z}_{\geq 0}^{V_0} \to \prod_{j \in V_0} \mathbb{Z}/e_{j,p(j)}, \quad \phi(\boldsymbol{u})_j = \sum_{i \succeq j} u_i \pmod{e_{j,p(j)}}. \tag{4.0.10}$$

Hence, the restriction $\phi\big|_G : G \to \prod_{j \in V_0} \mathbb{Z}/e_{j,p(j)}$ *is a group isomorphism, and one has* $\phi(\boldsymbol{c}) = \phi\big|_G \circ \sigma(\boldsymbol{c} + \boldsymbol{e}^T)$*, where* $\boldsymbol{e}^T \in G$ *is the identity element, and* $\boldsymbol{c} \in \mathbb{Z}_{\geq 0}^{V_0}$ *is arbitrary.*

It remains to combine the two propositions to give an explicit inverse of the isomorphism $\phi\big|_G$ (to express the sandpile group $G \subseteq M \subseteq \mathbb{Z}_{\geq 0}^{V_0}$ in terms of configurations), which is part (iii) of Theorem 3.4.

Proposition 4.4. *The inverse of* $\phi\big|_G$ *is given by* (3.0.2).

Proof. Since $\phi\big|_G$ is an isomorphism and hence a bijection, we need only check that (i) the formula (3.0.2) indeed gives a recurrent element (i.e., an element of G) for every $\boldsymbol{v} \in \prod_{j \in V_0} \mathbb{Z}/e_{j,p(j)}$, and (ii) the composition $\phi \circ \phi^{-1}$ is the identity.

The first part (i) is obviously true by definition, using Proposition 4.1 (we defined f_j to satisfy this). Let us check (ii). To avoid confusion, let us denote the map recursively defined in (3.0.2) by $(\phi^{-1})'$ for now (so replace every instance of ϕ^{-1} with $(\phi^{-1})'$. We then compute

$$\phi \circ (\phi^{-1})'(\boldsymbol{v})_j = f_j(v_j) - \sum_{i \succ j} (\phi^{-1})'(\boldsymbol{v})_i + \sum_{i \succ j} (\phi^{-1})'(\boldsymbol{v})_i \pmod{e_{j,p(j)}} = v_j. \tag{4.0.11}$$

□

5 Acknowledgements

The authors would like to thank László Babai for introducing them to the sandpile model through lectures at the 2005 University of Chicago REU, supported by the University's VIGRE grant; in particular for explaining all of the material in Section 2, based on [1]. They are also grateful to Babai for proposing the problem, for providing many references, and for extensive comments on earlier versions of this note. The authors are grateful to the participants and organizers of the aforementioned REU, without whom this research would not have occurred. The work of the second author was partially supported by an NSF GRF.

References

[1] L. Babai and E. Toumpakari. "A structure theory of the abelian sandpile model for digraphs," in progress.

[2] H. Bai. "On the critical group of the n-cube." *Lin. Alg. Appl.* **369** (2003), 251–261.

[3] P. Bak, C. Tang, and K. Wiesenfeld. "Self-organized criticality: an explanation for the $1/f$ noise." *Phys. Rev. Lett.* **59** (1987), 381–384.

[4] N. L. Biggs. "Chip firing and the critical group of a graph." *J. Alg. Combin.* **9** (1999), 25–45.

[5] A. Björner, L. Lovász, and P. W. Shor. "Chip-firing Games on Graphs." *Europ. J. Combinatorics* **12**. (1991), 283–291.

[6] M. Creutz. "Abelian Sandpiles." *Computers in Physics* **5** (1991), 198–203.

[7] A. Dartois and C. O. Magnien. "Results and conjectures on the Sandpile Identity on a lattice." *Disc. Math. and Theoret. Comput. Sci.* **AB** (2004), 89–102.

[8] D. Dhar. "Self-organized critical state of sandpile automaton models." *Phys. Rev. Lett.* **64** (1990), no. 14, 1613–1616.

[9] D. Dhar. "Studying self-organized criticality with exactly solved models." **arXiv: cond-mat/9909009**

[10] D. Dhar, P. Ruelle, S. Sen, and D. Verma. "Algebraic aspects of Abelian sandpile models." *J. Phys. A* **28** (1995), 805–831.

[11] J. van den Heuvel. "Algorithmic aspects of a chip-firing game." *Combinatorics, Probability, and Computing* **10** (2001), 505–529.

[12] B. Jacobson, A. Niedermaier, and V. Reiner. "Critical groups for complete multipartite graphs and cartesian products of complete graphs." *J. Graph Theory* **44/3** (2003), 231–250.

[13] Y. Le Borgne and D. Rossin. "On the identity of the sandpile group." *Discrete Math.* **256** (2002), 775–790.

[14] S. N. Majumdar and D. Dhar. "Height correlations in the abelian sandpile model." *J. Phys A* **24** (1991), no. 7, L357–L362.

[15] S. N. Majumdar and D. Dhar. "Equivalence of the abelian sandpile model and the $q \to 0$ limit of the Potts model." *Physica A* **185** (1992), 129–145.

[16] J. Spencer. "Balancing vectors in the max norm." *Combinatorica* **6** (1986), 55–65.

[17] E. Toumpakari. "On the sandpile group of regular trees." *Europ. J. Combinatorics*, to appear.

[18] D. L. Turcotte. "Self-organized criticality." *Rep. Prog. Phys.* **62** (1999), 1377-1429.

W.C.: California Institute of Technology, MSC 176, Pasadena, CA 91126, USA;
`chenw@caltech.edu`

T.S.: Department of Mathematics, University of Chicago, Chicago, IL 60637, USA;
`trasched@math.uchicago.edu`