Séminaire Lotharingien de Combinatoire **77** (2018), Article B77g

DECOMPOSING RECURRENT STATES OF THE ABELIAN SANDPILE MODEL

MARK DUKES AND THOMAS SELIG

Abstract. The recurrent states of the Abelian sandpile model (ASM) are those states that appear infinitely often. For this reason they occupy a central position in ASM research. The set of stable configurations on a graph form a Markov chain whereby a transition from a configuration c to another c' occurs if the addition of a grain to c and the resulting sequence of topplings (if any) yields the state c'. Checking whether a stable configuration is recurrent is a far from trivial task and requires Dhar's criterion, an algorithmic process, to be used. We present several new results for classifying recurrent states of the Abelian sandpile model on graphs that may be decomposed in a variety of ways. These results represent an enormous computational saving with respect to Dhar's criterion. Furthermore, they allow us to classify, for certain families of graphs, recurrent states in terms of the recurrent states of its components. We use these decompositions to give recurrence relations for the generating functions of the level statistic on the recurrent configurations. We also interpret our results with respect to the sandpile group.

1. Introduction

The Abelian sandpile model (ASM) has attracted a considerable amount of attention down through the years, and remains a constant source of new and interesting research topics. Perhaps the most recent of these is its relation to the discrete Riemann–Roch formula for graphs [3]. One fundamental aspect of ASM research concerns the classification of the recurrent states of the model; those states that appear infinitely often in the long-time running of the model.

In order to calculate this important set of recurrent states of the ASM on a given graph one appeals to Dhar's criterion [12], an algorithmic procedure for checking when a stable configuration on a graph is recurrent. Dhar's criterion is a global one in the sense that it induces a wave of topplings to occur throughout a graph and it is indifferent to some sort of regular structure that may be inherent within the graph. Consequently, the computations required to go through all possible stable configurations on a graph and check whether the criterion holds for each can take a significant amount of time.

In this paper we present several new results classifying recurrent states of the ASM on graphs which may be decomposed in a variety of ways. The computational benefit of our

Key words and phrases. Abelian sandpile model, recurrent states, graph decomposition, level polynomial, Tutte polynomial.

Supported by the EPSRC grant EP/M015874/1: New combinatorial perspectives on the Abelian sandpile model.

results to the classification problem can be immense. The level statistic of a sandpile configuration, a quantity that is equal (up to an additive constant) to the sum of its heights, is studied with respect to these decompositions.

Alongside our classification results we give the generating functions of the level statistic (to be defined) on these graph decompositions, which we believe to represent a significant contribution in terms of computational savings. We then show the effect of these graph decompositions on the sandpile group, which is the set of recurrent states equipped with an Abelian addition operation. These results also have an added benefit in that they provide a combinatorial setting in which to interpret set products of configurations that are almost recurrent states (up to some additive vector). The studied decompositions in components of the graph are also lifted to product decompositions of the sandpile group of the whole graph into the sandpile group of the components. We almost interpret these decompositions in terms of set products of configurations (see Section 5 for details).

The motivation behind the current paper is twofold. First, we are interested in generalizing results of the first author for recurrent configurations of the sandpile model on the complete bipartite graph, and understanding how altering the graph (e.g. a doubling of edges) changes the combinatorial interpretation of the recurrent states as parallelogram polyominoes [13, 1, 2]. Second, we are interested in understanding the sandpile model on graphs that have some notion of a 'decomposition point', a vertex or edge that might afford product-style theorems to sets of recurrent states. The purpose behind this second goal is to attribute a meaning to products of combinatorial objects that represent recurrent states. Being able to explain this has the advantage that constructions on these objects (e.g. restricted lattice paths in the case of [13]) and statistics upon them can be interpreted in terms of a toppling analysis on an appropriate graph. Several other authors have considered decompositions on graphs in relation to the sandpile group and these include Lorenzini [17], Jacobson et al. [15], Berget et al. [4], and Levine [16]. In particular, Lorenzini [17] observed our Theorem 5.2 in the remark following the proof of his Proposition 1.

We now introduce the Abelian sandpile model on a graph and review some notation that is necessary in order for us to present our results. Let $G = (V, E)$ be a finite, connected, loop-free, undirected multigraph with vertex set $V = \{v_0, \ldots, v_n\}$. Let $d_i = d_i^G = \deg(v_i)$ be the degree of the vertex v_i in G. We will consider the sandpile model on the graph G in which a distinguished vertex, v_0 say, acts as a *sink*. We will indicate which vertex of a graph is being treated as a sink by writing it as a pair, e.g. (G, v_0).

Let $\mathbb{Z}_+$ be the set of non-negative integers. A *configuration* on (G, v_0) is a vector $c = (c_1, \ldots, c_n) \in \mathbb{Z}_+^n$ that assigns the number c_i to vertex v_i. We think of c_i as representing the number of grains of sand at the vertex v_i. Denote by $\mathrm{Config}_{v_0}(G)$ the set of all configurations on (G, v_0). Let $\alpha_i \in \mathbb{Z}_+^n$ be the vector with 1 in the i-th position and 0 elsewhere. Throughout this paper, we will find it useful to refer to sets of vertices in V and will use the following notation: given $i, j \in \mathbb{Z}_+$ with $i \leq j$, let $V_{[i,j]} = \{v_i, v_{i+1}, \ldots, v_j\}$. Also, for some subset $W \subseteq V$, we let $G[W]$ be the subgraph of G with vertex set W and edge set the edges of G with both endpoints in W.

We will say that a vertex v_i in a configuration $c = (c_1, \ldots, c_n) \in \mathrm{Config}_{v_0}(G)$ is *stable* if $c_i < d_i$. Otherwise it is called *unstable*. A configuration is called stable if all its non-sink vertices are stable, and we denote $\mathrm{Stable}_{v_0}(G)$ the set of all stable configurations on (G, v_0).

Unstable vertices may topple. We define the toppling operator T_i corresponding to the toppling of an unstable vertex $v_i \in V$ in a configuration $c \in \mathrm{Config}_{v_0}(G)$ by:

$$T_i(c) := c - d_i\alpha_i + \sum_{j:\{v_i,v_j\}\in E} \alpha_j, \tag{1}$$

where the sum is over all vertices adjacent to v_i, counted with multiplicity.

Performing this toppling may cause other vertices to become unstable, and we may topple these also. One can show that starting from some unstable configuration c and toppling successively unstable vertices, we eventually reach a stable configuration c' (think of the sink as absorbing grains). Moreover, this configuration c' does not depend on the sequence in which vertices are toppled. We write $c' = \sigma(c)$ and call it the *stabilization* of c.

We now define a Markov chain on the set of stable configurations of a graph (G, v_0). Let $\mu = (\mu_1, \dots, \mu_n)$ be a probability distribution on $\{1, \dots, n\}$ such that $\mu_i > 0$ for all $i \in \{1, \dots, n\}$. At each step of the Markov chain we add a grain at the vertex v_i with probability μ_i and stabilize the resulting configuration. Formally the transition matrix Q is given by:

$$\text{for all } c, c' \in \mathrm{Stable}_{v_0}(G)\,,\ Q(c, c') = \sum_{i=1}^{n} \mu_i \mathbb{1}_{\sigma(c+\alpha_i)=c'}.$$

The recurrent states for the Markov chain are the set of configurations which appear infinitely often in the long-time running of the model. Given a graph (G, v_0), we let $\mathrm{Rec}_v(G)$ be the set of recurrent states on the graph G in which the vertex v acts as the sink. We omit the subscript when it is clear which vertex is acting as the sink.

Proposition 1.1. *A configuration $c \in \mathrm{Config}_{v_0}(G)$ is* recurrent *if there exists some configuration $a \in \mathrm{Config}_{v_0}(G)$, satisfying $a_i \geq d_i$ for all $i \in \{1, \dots, n\}$, such that $c = \sigma(a)$.*

An *orientation* $\mathcal{O}$ of G is an orientation of every edge of E. If H is a subgraph of G, then $\mathcal{O}\big|_H$ will denote the restriction of the orientation $\mathcal{O}$ to the edges of H. An orientation is *acyclic* if it contains no directed cycles. A vertex $v \in V$ is a *source* of the orientation $\mathcal{O}$ if it has no incoming edges. It is easy to check that an acyclic orientation has at least one source. The following result was first stated in these terms by Biggs [6], although the author credits a previous paper [14] as having equivalent results.

Theorem 1.2. *Let $c \in \mathrm{Stable}_{v_0}(G)$. Then c is recurrent if, and only if, there exists an acyclic orientation $\mathcal{O}$ of G, such that*

(a) *The sink v_0 is the unique source of $\mathcal{O}$. We say that $\mathcal{O}$ is* sink-rooted.

(b) *We have:*

$$c_i \geq \mathrm{out}_i(\mathcal{O}) \ \text{ for all } 1 \leq i \leq n, \tag{2}$$

where $\mathrm{out}_i(\mathcal{O})$ is the number of outgoing edges from the vertex v_i in the orientation $\mathcal{O}$.

Proposition 1.1 and Theorem 1.2 provide two characterizations of recurrent states. A further characterization, in terms of the so-called burning algorithm (see [19]) will be given in the proof of Theorem 3.2. However, it is in general difficult to describe all the recurrent states for a given graph (G, v_0). Using the matrix-tree theorem, one can show that the number of recurrent states equals the number of spanning trees of the graph [19]. There are various bijective proofs of this result: in [19] the author provides a non-canonical bijection between recurrent states and spanning trees, while the authors of [5] and [8] both provided refined versions which enumerate recurrent configurations according the the level statistic.

This statistic is a popular statistic on sandpile configurations, and we will study it in conjunction with some of our decomposition theorems. Given a recurrent configuration $c \in \mathrm{Rec}_{v_0}(G)$, define the *level* of c to be

$$\mathrm{level}_{v_0}(c) := d_{v_0} - |E| + \sum_{v \in V_{[1,n]}} c_v, \tag{3}$$

where $|E|$ denotes the number of elements in the set E. From [18, Thm. 3.5] we have that if $G = (V, E)$ is a graph and $c \in \mathrm{Rec}_{v_0}(G)$, then $0 \leq \mathrm{level}_{v_0}(c) \leq |E| - |V|$. Consequently the level of a recurrent configuration is always a non-negative integer. We define the *level polynomial* of a graph G to be the generating function of the level statistic over the set of recurrent configurations on that graph;

$$\mathrm{Level}_{G,v_0}(x) := \sum_{c \in \mathrm{Rec}_{v_0}(G)} x^{\mathrm{level}_{v_0}(c)}. \tag{4}$$

Recall that the Tutte polynomial of a (connected) graph $G = (V, E)$ is defined by

$$T_G(x, y) := \sum_{S \subseteq E} (x-1)^{\mathrm{cc}(S)-1} (y-1)^{\mathrm{cc}(S)+|S|-|V|}, \tag{5}$$

where for $S \subseteq E$, $\mathrm{cc}(S)$ denotes the number of connected components of the subgraph (V, S).

The level and Tutte polynomials of a graph are related by the following well-known result that was initially proven by López [18], following a conjecture by Biggs. Subsequent combinatorial (bijective) proofs have been given, for instance by Cori and Le Borgne [8], and Bernardi [5].

Theorem 1.3. *Let G be a graph. Then we have $\mathrm{Level}_{G,v_0}(x) = T_G(1, x)$. In particular, the level polynomial is independent of the choice of sink.*

Theorem 1.3 makes some of the results of this paper such as Corollaries 3.4 or 4.2 entirely natural in terms of spanning trees of the graph. However, the structure inherent to the sandpile model, both in the case of the Markov chain transitions previously mentioned in this section and of the additive structure of the sandpile group introduced in Section 5, is much more explicit in terms of the recurrent states than it is in terms of the spanning trees. As such, the focus of this paper is very much on the explicit description of the recurrent states of the graph with respect to the decompositions studied, which is more difficult.

Example 1.4. Let G be the graph in Figure 1, where the vertex v_0 acts as the sink.

We have $\mathrm{Stable}_{v_0}(G) = \{(0,0), (0,1), (1,0), (1,1)\}$. By Theorem 1.2, the configuration $(0,0)$ cannot be recurrent. Indeed, any direction given to the edge (v_1, v_2) would result in one of those two vertices having an outgoing edge, which would contradict Condition (2) (the inequality in Theorem 1.2). For the configuration $(0,1)$, we build a sink-rooted, acyclic orientation which satisfies Condition (2) by directing the edges from v_0 to v_1, from v_0 to v_2,

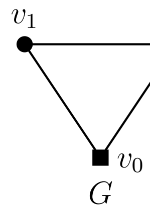


FIGURE 1. The graph G: the sink v_0 is represented by a square, the non sink vertices v_1, v_2 by circles. The squares and circles will become important later in the paper when non-sink vertices may become sink vertices.

and from v_2 to v_0, as illustrated in Figure 2. Thus, by Theorem 1.2, the configuration $(0,1)$ is recurrent.

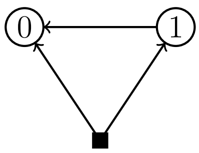


FIGURE 2. The configuration $(0,1)$ with the corresponding orientation.

By symmetry, the configuration $(1,0)$ is also recurrent. Finally, note that the orientation of Figure 2 also satisfies Condition (2) for the configuration $(1,1)$, which is therefore recurrent. Finally, we conclude that

$$\mathrm{Rec}_{v_0}(G) = \{(0,1),(1,0),(1,1)\}.$$

We may deduce from this the level polynomial for the graph G. We get $\mathrm{Level}_{G,v_0}(x) = x+2$. This may also be deduced from Theorem 1.3 since the Tutte polynomial of G is given by $T_G(x,y) = x^2 + x + y$, so that $T_G(1,x) = x+2$.

The process by which we determined that the configuration $(1,1)$ is also recurrent in Example 1.4 above leads to an interesting generalization. Given a graph (G, v_0), there is a natural partial order $\preceq$ on the set $\mathrm{Config}_{v_0}(G)$ of all configurations by setting $c \preceq c'$ if, and only if, $c_i \le c'_i$ for all $i \in \{1,\dots,n\}$.

Now notice that, if we have a recurrent configuration $c = (c_1,\dots,c_n)$ and an orientation $\mathcal{O}$ which satisfies Condition (2), then the same orientation $\mathcal{O}$ also satisfies the condition for the configuration $c + \alpha_i$, for any $i \in \{1,\dots,n\}$. This is also immediate from the definition of the Markov chain, since there is a positive probability of adding a grain at the vertex v_i, thus $Q(c, c+\alpha_i) > 0$. This leads to the following result.

Proposition 1.5. *For any graph G, the configuration $c^{\max} := (d_1-1,\dots,d_n-1)$ is recurrent. It is the maximal recurrent configuration for the partial order $\preceq$. Moreover, for any $c = (c_1,\dots,c_n)$, the interval $[c, c^{\max}]$ of the poset $(\mathrm{Rec}_{v_0}(G), \preceq)$ is given by*

$$[c, c^{\max}] = \{c_1, c_1+1, \dots, d_1-1\} \times \dots \times \{c_n, c_n+1, \dots, d_n-1\}.$$

There is a well established characterization of recurrent states of the sandpile model on the complete graph in terms of parking functions. We will make use of this in later sections, but will state it here since it serves as a good reference for considering small examples throughout the paper.

Definition 1.6. A *parking function of size n* is a sequence of non-negative integers $(a_1, \ldots, a_n)$ such that, when they are rearranged in weakly increasing order as $(b_1, \ldots, b_n)$, we have $b_i < i$ for all $1 \leq i \leq n$.

Theorem 1.7. *The set* $\mathrm{Rec}_{v_0}(K_n)$ *is characterized by the following set*

$$\mathrm{Rec}_{v_0}(K_n) = \{(n-2-a_1, \ldots, n-2-a_{n-1}) \;:\; (a_1, \ldots, a_{n-1}) \text{ if a parking function of size } n\}.$$

Parking functions can be easily written down from permuting the heights of horizontal steps of Dyck paths/sequences. Let $\mathrm{Dyck}_n = \{(a_1, \ldots, a_n) \;:\; a_i \in \{0, \ldots, i-1\} \text{ and } a_1 \leq a_2 \leq \cdots \leq a_n\}$. For example,

$$\mathrm{Dyck}_3 = \{(0,0,0), (0,0,1), (0,0,2), (0,1,1), (0,1,2)\}. \tag{6}$$

Using Theorem 1.7 in conjunction with the notion of Dyck paths/sequences allows us to write down an explicit expression for $\mathrm{Rec}_{v_0}(K_n)$:

$$\mathrm{Rec}_{v_0}(K_n) = \{(n-2-a_{\pi(1)}, \ldots, n-2-a_{\pi(n-1)}) \;:\; \pi \in \mathrm{Sym}_{n-1}, (a_1, \ldots, a_{n-1}) \in \mathrm{Dyck}_{n-1}\}. \tag{7}$$

Example 1.8. We have

$$\mathrm{Rec}_{v_0}(K_4) = \{(2-a_{\pi(1)}, 2-a_{\pi(2)}, 2-a_{\pi(3)}) \;:\; \pi \in \mathrm{Sym}_3 \text{ and } (a_1, a_2, a_3) \in \mathrm{Dyck}_3\}. \tag{8}$$

Using the list in Equation (6), we have

$$\begin{aligned}\mathrm{Rec}_{v_0}(K_4) = &\{(2,2,2), (2,2,1), (2,1,2), (1,2,2), (2,2,0), (2,0,2), (0,2,2), (2,1,1),\\ &(1,2,1), (1,1,2), (2,1,0), (2,0,1), (1,2,0), (1,0,2), (0,1,2), (0,2,1)\}.\end{aligned} \tag{9}$$

Example 1.9. To determine $\mathrm{Rec}_{v_0}(K_5)$, we list all 14 elements of Dyck_4:

$$\begin{aligned}\mathrm{Dyck}_4 = &\{(0,0,0,0), (0,0,0,1), (0,0,0,2), (0,0,0,3), (0,0,1,1), (0,0,1,2), (0,0,1,3),\\ &(0,0,2,2), (0,0,2,3), (0,1,1,1), (0,1,1,2), (0,1,1,3), (0,1,2,2), (0,1,2,3)\}\end{aligned}$$

Using this we have

$$\mathrm{Rec}_{v_0}(K_5) = \{(3-a_{\pi(1)}, 3-a_{\pi(2)}, 3-a_{\pi(3)}, 3-a_{\pi(4)}) \;:\; \pi \in \mathrm{Sym}_4 \text{ and } (a_1, a_2, a_3) \in \mathrm{Dyck}_4\}.$$

There will be 125 of these sequences, and we do not list them here.

Note that in general there are n^{n-2} parking configurations on K_n, so the set of recurrent configurations may be huge.

2. Recurrent states under edge duplication

In this section we will show how recurrent states of the sandpile model on a graph change when every edge of a graph is replaced with k copies of itself. Let $G = (V, E)$ be a graph. For a positive integer k we define $G^{(k)}$ to be the multigraph G where every edge of E is replaced with k copies of itself. That is $G^{(k)} = \left(V, E^{(k)}\right)$, where $E^{(k)}$ is the multiset $\bigcup\limits_{e \in E} \{e_1, \ldots, e_k\}$, with $e_j = e$ for all $j \in \{1, \ldots, k\}$.

Theorem 2.1. *Let* $c = (c_1, \ldots, c_n) \in \mathrm{Config}_{v_0}(G)$. *Then the following are equivalent:*

(a) $c \in \mathrm{Rec}_{v_0}\left(G^{(k)}\right)$.
(b) $\tilde{c} := (\lfloor c_1/k \rfloor, \ldots, \lfloor c_n/k \rfloor) \in \mathrm{Rec}_{v_0}(G)$.

Proof. We first show that (a) $\Rightarrow$ (b). Assume that $c \in \mathrm{Rec}_{v_0}\left(G^{(k)}\right)$. By Theorem 1.2, there exists a sink-rooted acyclic orientation $\mathcal{O}^{(k)}$ on $G^{(k)}$, such that:

$$c_i \geq \mathrm{out}_i\left(\mathcal{O}^{(k)}\right) \quad \text{for all } 1 \leq i \leq n. \tag{10}$$

Now since $\mathcal{O}^{(k)}$ is acyclic, for any edge $e \in E$, the corresponding edges $e_1, \ldots, e_k \in E^{(k)}$ must all be oriented in the same direction (if two are oriented in opposite directions, those two edges create a directed cycle). By giving the edge e the same orientation as that of the corresponding edges $e_1, \ldots, e_k$, this allows us to define an orientation $\mathcal{O}$ on G, such that

$$\mathrm{out}_i\left(\mathcal{O}^{(k)}\right) = k \cdot \mathrm{out}_i(\mathcal{O}) \quad \text{for all } 1 \leq i \leq n. \tag{11}$$

Moreover, it is easy to check that since the orientation $\mathcal{O}^{(k)}$ is sink-rooted and acyclic, so is the orientation $\mathcal{O}$. Finally, we have:

$$\mathrm{out}_i(\mathcal{O}) = \frac{1}{k}\,\mathrm{out}_i\left(\mathcal{O}^{(k)}\right) \leq \frac{1}{k} c_i = \frac{c_i}{k} \quad \text{for all } 1 \leq i \leq n,$$

by using (11) followed by (10). Since $\mathrm{out}_i(\mathcal{O})$ is an integer, this implies that

$$\mathrm{out}_i(\mathcal{O}) \leq \left\lfloor \frac{c_i}{k} \right\rfloor = \tilde{c}_i \quad \text{for all } 1 \leq i \leq n.$$

Since $\mathcal{O}$ is a sink-rooted acyclic orientation of G satisfying the above, it follows from Theorem 1.2 that $\tilde{c} \in \mathrm{Rec}_{v_0}(G)$.

We now show that (b) $\Rightarrow$ (a). Assume that c is such that $\tilde{c} \in \mathrm{Rec}_{v_0}(G)$. By Theorem 1.2, there exists a sink-rooted acyclic orientation $\mathcal{O}$ on G, such that:

$$\tilde{c}_i \geq \mathrm{out}_i(\mathcal{O}) \quad \text{for all } 1 \leq i \leq n. \tag{12}$$

Similarly to above, by orienting all edges $e_1, \ldots, e_k$ of $E^{(k)}$ corresponding to an edge e of E in the same direction as e, we get a sink-rooted acyclic orientation $\mathcal{O}^{(k)}$ on $G^{(k)}$ satisfying Equation (11). It follows that:

$$c_i \geq k \cdot \left\lfloor \frac{c_i}{k} \right\rfloor = k \cdot \tilde{c}_i \geq k \cdot \mathrm{out}_i(\mathcal{O}) = \mathrm{out}_i\left(\mathcal{O}^{(k)}\right) \quad \text{for all } 1 \leq i \leq n,$$

and as before, Theorem 1.2 implies that $c \in \mathrm{Rec}_{v_0}\left(G^{(k)}\right)$. $\square$

From Theorem 2.1 we can deduce the following formula for the level polynomial of the graph $G^{(k)}$.

Corollary 2.2. *Let k be a positive integer. We have:*

$$\mathrm{Level}_{G^{(k)},v_0}(x) = \left(1 + x + \ldots + x^{k-1}\right)^{|V|-1} \cdot \mathrm{Level}_{G,v_0}\left(x^k\right).$$

In particular, $\left|\mathrm{Rec}_{v_0}\left(G^{(k)}\right)\right| = k^{(|V|-1)} |\mathrm{Rec}_{v_0}(G)|$.

Proof. Let $c \in \mathrm{Config}_{v_0}(G^{(k)})$, and $\tilde{c} := (\lfloor c_1/k \rfloor, \ldots, \lfloor c_n/k \rfloor)$. Define $a_i := c_i - k\lfloor c_i/k \rfloor$ for $i \in \{1, \ldots, n\}$. Standard floor inequalities show that $a_i \in \{0, \ldots, k-1\}$ for all $1 \leq i \leq n$.

Thus Theorem 2.1 can be re-stated as follows: $c = (c_1, \ldots, c_n) \in \mathrm{Rec}_{v_0}\left(G^{(k)}\right)$ if, and only if, there exists a unique pair (γ, r) with $\gamma = (\gamma_1, \ldots, \gamma_n) \in \mathrm{Rec}_{v_0}(G)$, $r = (r_1, \ldots, r_n) \in \{0, \ldots, k-1\}^n$ such that $c_i = k\gamma_i + r_i$ for all $1 \leq i \leq n$.

If $c = (c_1, \ldots, c_n) \in \mathrm{Rec}_{v_0}\left(G^{(k)}\right)$ then we can write

$$\mathrm{level}_{v_0}(c) = d_{v_0}^{G^{(k)}} - |E^{(k)}| + \sum_{i=1}^{n} c_i = k d_{v_0}^{G} - k|E| + \sum_{i=1}^{n} (k\gamma_i + r_i) = k \cdot \mathrm{level}_{v_0}(\gamma) + \sum_{i=1}^{n} r_i.$$

Therefore, we can compute:

$$\begin{aligned}
\mathrm{Level}_{G^{(k)},v_0}(x) &= \sum_{c \in \mathrm{Rec}_{v_0}\left(G^{(k)}\right)} x^{\mathrm{level}_{v_0}(c)} = \sum_{\substack{\gamma \in \mathrm{Rec}_{v_0}(G) \\ 0 \leq r_1, \ldots, r_n < k}} x^{(k \cdot \mathrm{level}_{v_0}(\gamma) + r_1 + \ldots + r_n)} \\
&= \left(\sum_{0 \leq r_1, \ldots, r_n < k} (x^{r_1}) \ldots (x^{r_n}) \right) \cdot \left(\sum_{\gamma \in \mathrm{Rec}_{v_0}(G)} \left(x^k\right)^{\mathrm{level}_{v_0}(\gamma)} \right) \\
&= \left(1 + x + \ldots + x^{k-1}\right)^n \cdot \mathrm{Level}_{G,v_0}\left(x^k\right).
\end{aligned}$$

□

Example 2.3. We develop an example to show some of the computational gains which can be made through applying Theorem 2.1 and its corollary. Consider the graph G from Figure 3 below. We wish to determine the recurrent configurations of $G^{(2)}$.

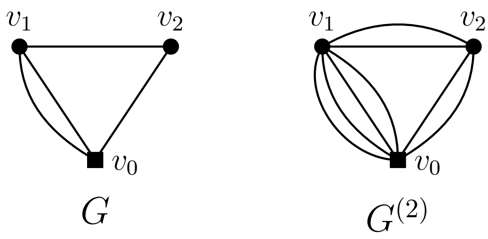


FIGURE 3. The graphs G and $G^{(2)}$.

We can show that $\mathrm{Rec}_{v_0}(G) = \{(0,1), (1,0), (1,1), (2,0), (2,1)\}$. Now Theorem 2.1 tells us that $c = (c_1, c_2) \in \mathrm{Rec}_{v_0}\left(G^{(2)}\right)$ if, and only if, there exists a unique pair (γ, α) with $\gamma = (\gamma_1, \gamma_2) \in \mathrm{Rec}_{v_0}(G)$, $\alpha = (\alpha_1, \alpha_2) \in \{0,1\}^2$ such that $c_i = k\gamma_i + \alpha_i$ for all $i = 1, 2$.

In words, recurrent configurations on $G^{(2)}$ are obtained from recurrent configurations on G by multiplying them by two and adding up to one grain at each vertex. This is made clear in Table 1.

This table allows us to determine $\mathrm{Rec}_{v_0}\left(G^{(2)}\right)$. We may also compute the level polynomials. We get

$$\mathrm{Level}_{G,v_0}(x) = x^2 + 2x + 2,$$

and thus, by Corollary 2.2

$$\mathrm{Level}_{G^{(k)},v_0}(x) = (1+x)^2 \cdot \left(\left(x^2\right)^2 + 2\left(x^2\right) + 2\right) = x^6 + 2x^5 + 3x^4 + 4x^3 + 4x^2 + 4x + 2,$$

Recurrent configurations on G	Corresponding recurrent configurations on $G^{(2)}$
$(0,1)$	$(0,2)$ $(0,3),(1,2),(1,3)$
$(1,0)$	$(2,0)$ $(2,1),(3,0),(3,1)$
$(1,1)$	$(2,2)$ $(2,3),(3,2),(3,3)$
$(2,0)$	$(4,0)$ $(4,1),(5,0),(5,1)$
$(2,1)$	$(4,2)$ $(4,3),(5,2),(5,3)$

TABLE 1. The correspondence between recurrent configurations on G and on $G^{(2)}$.

which can also be computed directly using the second column of Table 1.

3. Recurrent states on graphs having cut vertices

In this section we show how the recurrent states on a graph which contains a cut vertex may be decomposed into recurrent states on the two different parts of the graph which meet at this cut vertex. We then build on this result by showing that a graph whose set of cut vertices have a tree like structure has a recurrent configuration set that can be decomposed in a similar manner.

Definition 3.1. Let $G = (V, E)$ be a finite, connected graph. We say that $v \in V$ is a *cut vertex* of G if removing the vertex v and all edges incident to v makes G disconnected.

Theorem 3.2. *Let $G = (V, E)$ be a finite, connected graph, on the vertex set*

$$V = \{v_0, v_1, \ldots, v_{n+m}\}$$

where $m, n \geq 1$. Assume that v_n is a cut vertex of G. Let $H_1 := G[V_{[0,n]}]$ and $H_2 := G[V_{[n,n+m]}]$. Given $c = (c_1, \ldots, c_{n+m}) \in \mathrm{Config}_{v_0}(G)$, let $c^{(1)} := (c_1, \ldots, c_{n-1}, c_n - d_n^{H_2})$ and $c^{(2)} := (c_{n+1}, \ldots, c_{n+m})$, where $d_n^{H_2}$ is the degree of the vertex v_n in the graph H_2. Then

$$c \in \mathrm{Rec}_{v_0}(G) \quad \text{if and only if} \quad c^{(1)} \in \mathrm{Rec}_{v_0}(H_1) \ \text{ and } c^{(2)} \in \mathrm{Rec}_{v_n}(H_2)\,.$$

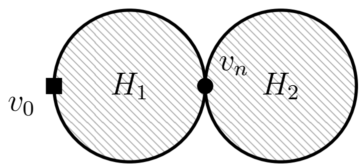


FIGURE 4. The graph G with its cut vertex v_n.

Remark 3.3. Theorem 3.2 is obvious in terms of the spanning trees of the graph. Indeed, the spanning trees of such a graph G are simply given by the "independent" products of spanning trees of H_1 and H_2. The difficulty here is to describe such decompositions in terms of the recurrent states.

Proof. First let us assume that $c^{(1)} \in \mathrm{Rec}_{v_0}(H_1)$ and $c^{(2)} \in \mathrm{Rec}_{v_n}(H_2)$. Then there exists a sink-rooted acyclic orientation $\mathcal{O}_1$ of H_1, and an acyclic orientation $\mathcal{O}_2$ of H_2 whose unique source is the vertex v_n, such that:

$$c_i^{(1)} \geq \mathrm{out}_i(\mathcal{O}_1), \text{ for all } 1 \leq i \leq n, \text{ and } c_j^{(2)} \geq \mathrm{out}_j(\mathcal{O}_2), \text{ for all } n+1 \leq j \leq n+m.$$

Since the edges of the graphs H_1 and H_2 form a partition of the edges of G, we may define an orientation $\mathcal{O} := \mathcal{O}_1 \cup \mathcal{O}_2$ of G. By construction, $\mathcal{O}$ is sink-rooted and acyclic.

For $i \in \{1, \ldots, n-1\}$, we have:

$$c_i = c_i^{(1)} \geq \mathrm{out}_i(\mathcal{O}_1) = \mathrm{out}_i(\mathcal{O}).$$

Similarly, for $j \in \{n+1, \ldots, n+m\}$, we have:

$$c_j = c_j^{(2)} \geq \mathrm{out}_j(\mathcal{O}_2) = \mathrm{out}_j(\mathcal{O}).$$

Finally, at the vertex v_n, we have:

$$c_n = c_n^{(1)} + d_n^{H_2} \geq \mathrm{out}_n(\mathcal{O}_1) + d_n^{H_2} = \mathrm{out}_n(\mathcal{O}_1) + \mathrm{out}_n(\mathcal{O}_2) = \mathrm{out}_n(\mathcal{O}),$$

since v_n is a source of the orientation $\mathcal{O}_2$. Thus the orientation $\mathcal{O}$ satisfies Condition (2), and by Theorem 1.2, the configuration c is recurrent.

To show the converse, we first recall some results concerning the so-called *burning algorithm* (see [19]). This algorithm inputs a stable configuration a on a graph $G = (V, E)$, and establishes if a is recurrent. We now describe this algorithm.

- B1. Initialise by setting $V_0^B = \{v_0\}$. The set V_i^B will be the set of vertices burnt up to and including time i.
- B2. (i) Assume for some $i \geq 0$, we have constructed the set V_i^B of vertices burnt up to and including time i.
  - (ii) Let V_i^U be the set of vertices not yet burnt and set $G_i := G[V_i^U]$.
  - (iii) Now set $\tilde{V}_i := \{v \in V_i^U;\ a_v \geq d_v^{G_i}\}$ to be the set of unstable vertices in the configuration a restricted to G_i.
  - (iv) By construction, since a is stable in the graph G, the instability of any such vertex must arise from the burning of previous vertices.
  - (v) We set $V_{i+1}^B := V_i^B \cup \tilde{V}_i$.
- B3. The algorithm terminates when we reach a time i such that no vertices can subsequently be burnt, i.e. $\tilde{V}_i = \emptyset$. It outputs $V^B := V_i^B$, which is the set of all vertices burnt beforehand.

Notice that since $V_0^B \subseteq V_1^B \subseteq V_2^B \subseteq \cdots \subseteq V$, the burning algorithm does indeed terminate. Our interest lies in the following facts.

(1) The configuration a is recurrent if, and only if, $V^B = V$, i.e. the burning algorithm applied to a burns all vertices of the initial graph.

(2) Let $a \in \mathrm{Rec}_{v_0}(G)$. Define a total order $\prec_b$ on the set of vertices V by setting $v_i \prec_b v_j$ if, when applying the burning algorithm to a, either v_i is burned before v_j, or they are burned at the same time and $i < j$. Now define an orientation $\mathcal{O}_b$ of G by orienting the edge $\{v, w\}$ from v to w if, and only if, $v \prec_b w$. Then this orientation is sink-rooted, acyclic, and satisfies:

$$a_v \geq \mathrm{out}_v^{\mathcal{O}_b}, \text{ for all } v \in V.$$

We now return to the proof of Theorem 3.2. Let $c \in \mathrm{Rec}_{v_0}(G)$. Apply the burning algorithm to c, and let $\mathcal{O}_b$ be the corresponding sink-rooted acyclic orientation as in part (2) of the above. Since the cut vertex v_n lies on all paths from the sink to vertices v_k when $k \geq n+1$, this implies that the vertex v_n must be burnt before any vertices with higher indices. Therefore, in the orientation $\mathcal{O}_b$, all edges from v_n to vertices of H_2 are oriented away from v_n.

We now define $\mathcal{O}_1 := \mathcal{O}_b|_{H_1}$ and $\mathcal{O}_2 := \mathcal{O}_b|_{H_2}$. Due to the above, it follows that v_n is the unique source of the orientation $\mathcal{O}_2$. Moreover, by construction $\mathcal{O}_1$ is sink-rooted, and both orientations are acyclic. Now let $k \in \{1, \ldots, n+m\}$. We have:

$$\text{if } k \leq n-1, \text{ then } c_k^{(1)} = c_k \geq \mathrm{out}_k(\mathcal{O}) = \mathrm{out}_k(\mathcal{O}_1); \tag{13}$$

$$\text{if } k = n, \text{ then } c_n^{(1)} = c_n - d_n^{H_2} \geq \mathrm{out}_n(\mathcal{O}) - d_n^{H_2} = \mathrm{out}_n(\mathcal{O}) - \mathrm{out}_n(\mathcal{O}_2) = \mathrm{out}_n(\mathcal{O}_1); \tag{14}$$

$$\text{if } k \geq n+1, \text{ then } c_k^{(2)} = c_k \geq \mathrm{out}_k(\mathcal{O}) = \mathrm{out}_k(\mathcal{O}_2). \tag{15}$$

By Theorem 1.2, Equations (13) and (14) imply that $c^{(1)} \in \mathrm{Rec}_{v_0}(H_1)$, while Equation (15) implies that $c^{(2)} \in \mathrm{Rec}_{v_n}(H_2)$. This completes the proof of Theorem 3.2. ☐

Theorem 3.2 implies the following decomposition for the level polynomial of a graph with a cut vertex.

Corollary 3.4. *Let $G = (V, E)$ be a finite, connected graph, on the vertex set*

$$V = \{v_0, v_1, \ldots, v_{n+m}\}$$

where $m, n \geq 1$. Assume that v_n is a cut vertex of G. Let $H_1 := G[V_{[0,n]}]$ and $H_2 := G[V_{[n,n+m]}]$. Then we have:

$$\mathrm{Level}_{G,v_0}(x) = \mathrm{Level}_{H_1,v_0}(x) \cdot \mathrm{Level}_{H_2,v_n}(x).$$

Proof. Let c be a configuration on G. Let $c^{(1)}$ (respectively $c^{(2)}$) be the configurations on H_1 (respectively H_2) as defined in Theorem 3.2. Then we have:

$$\begin{aligned}
\mathrm{level}_{v_0}(c) &= d_0 - |E| + (c_1 + \ldots + c_{n-1}) + c_n + (c_{n+1} + \ldots + c_{n+m}) \\
&= d_0 - (|E_1| + |E_2|) + \left(\sum_{i=1}^{n-1} c_i^{(1)} \right) + (c_n^{(1)} + d_n^{H_2}) + \left(\sum_{i=n+1}^{n+m} c_i^{(2)} \right) \\
&= \left(d_0 - |E_1| + \sum_{i=1}^{n} c_i^{(1)} \right) + \left(d_n^{H_2} - |E_2| + \sum_{i=n+1}^{n+m} c_i^{(2)} \right) \\
&= \mathrm{level}_{v_0}\left(c^{(1)}\right) + \mathrm{level}_{v_n}\left(c^{(2)}\right),
\end{aligned}$$

and the claim immediately follows. ☐

We note that Corollary 3.4 is in fact a consequence of the Specification Theorem 1.3, and the following result, due to Tutte [21].

Theorem 3.5. *Let G be a graph. Assume G has a cut vertex which splits G into two components H_1 and H_2. Then we have:*

$$T_G(x, y) = T_{H_1}(x, y) \cdot T_{H_2}(x, y).$$

The proof of Theorem 3.5 relies on the expression of the Tutte polynomial in terms of internal and external activities of spanning trees. However, our Theorem 3.2 gives a combinatorial explanation of this decomposition (at least at the specification $x = 1$). It is also worth noting that a similar decomposition result was given for the so-called *flow polynomial*, which is another specification of the Tutte polynomial, in [20].

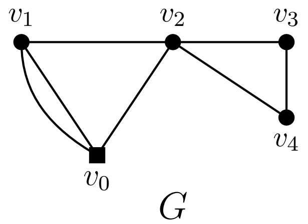


FIGURE 5. A graph G with its cut vertex v_2.

Example 3.6. Consider the graph G of Figure 5. The vertex v_2 is a cut vertex. We may thus determine the recurrent configurations by combining recurrent configurations on the subgraphs H_1 and H_2, adding two grains at the cut vertex v_2, since it has two incident edges in H_2. We show this in the Table 2, in which we also compute the level polynomials.

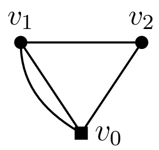
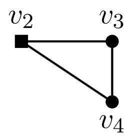
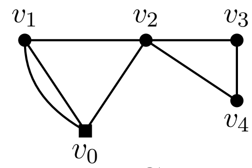
Graph	 H_1	 H_2	 G
Recurrent configurations	$(0, 1)$	$(0, 1), (1, 0), (1, 1)$	$(0, 3, 0, 1), (0, 3, 1, 0), (0, 3, 1, 1)$
	$(1, 0)$	$(0, 1), (1, 0), (1, 1)$	$(1, 2, 0, 1), (1, 2, 1, 0), (1, 2, 1, 1)$
	$(1, 1)$	$(0, 1), (1, 0), (1, 1)$	$(1, 3, 0, 1), (1, 3, 1, 0), (1, 3, 1, 1)$
	$(2, 0)$	$(0, 1), (1, 0), (1, 1)$	$(2, 2, 0, 1), (2, 2, 1, 0), (2, 2, 1, 1)$
	$(2, 1)$	$(0, 1), (1, 0), (1, 1)$	$(2, 3, 0, 1), (2, 3, 1, 0), (2, 3, 1, 1)$
Level polynomial	$x^2 + 2x + 2$	$x + 2$	$x^3 + 4x^2 + 6x + 4$ $= (x^2 + 2x + 2) \cdot (x + 2)$

TABLE 2. Applying Theorem 3.2 to determine the recurrent configurations of G.

3.1. **Cut vertices in graphs with an underlying tree-like structure.** Theorem 3.2 showed us how to determine the recurrent configurations of the sandpile model on a graph G when it has one cut vertex that splits it into two subgraphs H_1 and H_2. The subgraph H_1 contained the sink of G, and the intersection of H_1 and H_2 is the cut vertex v_n. If one considers the graphs H_1 and H_2 now as vertices, then Theorem 3.2 is the result (see Figure 6) for a graph constructed on a single, rooted edge, where: the root vertex corresponds to the subgraph H_1 containing the sink v_0 of G; the non-root vertex corresponds to the other subgraph H_2; the edge corresponds to the cut vertex v_n. In this subsection we generalize

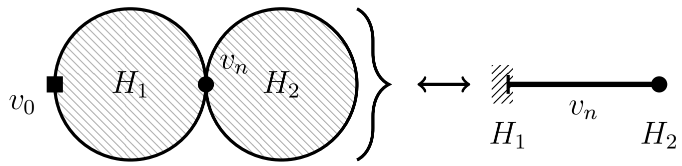


FIGURE 6. A graph with a cut vertex can be seen as a graph constructed on a single edge.

this from the case of a single rooted edge to the case of a rooted tree. First we will need some notation to be able to discuss the associate tree-structure.

Consider a tree (T, ρ), rooted at some vertex ρ. For clarity, we will refer to vertices of the tree as *nodes*, to distinguish them from vertices of the graph G. Let $\mathsf{nodes}\,(T)$ denote the node set of T. Since T is rooted, every non-root node $t \in \mathsf{nodes}\,(T) \setminus \{\rho\}$ will have an adjacent node on the (unique) path from t to the root ρ called the *parent* of t and denoted by $\mathsf{parent}(t)$. All other nodes adjacent to t are called *descendants* of t, and this set is denoted $\mathrm{desc}(t)$. If a node has no descendants then it is called a *leaf.*

We now explain what we mean for a graph G to have an underlying tree-structure (T, ρ).

Definition 3.7. We will say that the graph-sink pair (G, v_ρ) has an underlying *cv-tree structure* (T, ρ) if the following holds. The graph G can be written as

$$G = \bigcup_{t \in \mathsf{nodes}(T)} H_t,$$

where the pair (T, ρ) is a rooted tree and associated to every node $t \in \mathsf{nodes}\,(T)$ is a connected, loop-free, graph $H_t = (V_t, E_t)$. Each graph H_t has a distinguished vertex $v_t \in V_t$ such that:

(a) for all $t \in \mathsf{nodes}\,(T) \setminus \{\rho\}$, we have $V_t \cap V_{\mathsf{parent}(t)} = \{v_t\}$;
(b) if (t, t') is not an edge of T, then $V_t \cap V_{t'} = \emptyset$.

See Figure 7 for an illustration of the construction in Definition 3.7.

Remark 3.8. Every graph can be decomposed into *blocks*, where a block is a maximal 2-connected subgraph. It is well-known that this block decomposition has a tree structure as in that of Definition 3.7, and is thus a special case of a cv-tree structure (the blocks of the cv-tree structure may be graphs which are not 2-connected but on which the recurrent configurations are well known, e.g. trees). In particular, Theorems 3.9 and 5.7 allow one to reduce the analysis of recurrent configurations to the 2-connected case.

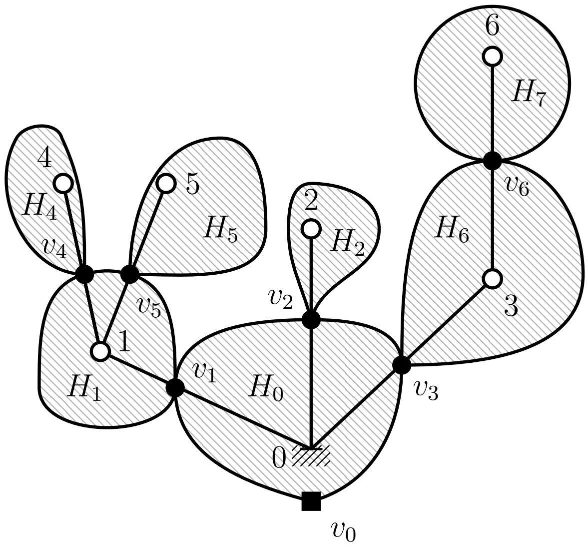


FIGURE 7. A graph with its underlying tree structure.

Theorem 3.9. *Let* $G = \bigcup_{t \in \mathsf{nodes}(T)} H_t$ *have an underlying cv-tree structure* (T, ρ)*. Let* $c \in \mathrm{Config}_{v_\rho}(G)$*. For any node* $t \in \mathsf{nodes}(T)$*, we let* $t_1, \ldots, t_k$ *be the descendants of* t *in* T*. Define the configurations* $c^{(t)}$ *on* $\mathrm{Config}_{v_t}(H_t)$ *by:*

$$c^{(t)}(v) := \begin{cases} c(v) & \text{if } v \notin \{v_{t_1}, \ldots, v_{t_k}\} \\ c(v) - d^{H_{t_i}}(v) & \text{if } v = v_{t_i} \end{cases}$$

Then $c \in \mathrm{Rec}_{v_\rho}(G)$ *if, and only if,* $c^{(t)} \in \mathrm{Rec}_{v_t}(H_t)$ *for all* $t \in \mathsf{nodes}(T)$*.*

Proof. Our proof is by induction on the number of nodes in the cv-tree $\mathsf{nodes}(T)$.

If $|\mathsf{nodes}(T)| = 1$, then the tree t is reduced to a single (root) vertex ρ, and the configuration $c^{(\rho)}$ is just the configuration c, since ρ has no descendants. In this case the statement is a tautology.

Next, let us suppose that we have proven our result for $|\mathsf{nodes}(T)| = k$ for some $k \geq 1$. Let (T, ρ) be a rooted tree with $k + 1$ nodes, and assume that $G = \bigcup_{t \in \mathsf{nodes}(T)} H_t$ has an underlying cv-tree structure (T, ρ). Fix some leaf ℓ of T. Since ℓ has no descendants, the configuration $c^{(\ell)}$ is just the restriction of c to H_ℓ. Define $G' := \bigcup_{t \in \mathsf{nodes}(T) \setminus \{\ell\}} H_t$, and let c' be the configuration on G' defined by

$$c'(v) := \begin{cases} c(v) \text{ if } v \neq v_\ell \\ c(v_\ell) - d^{H_\ell}(v_\ell) \text{ if } v = v_\ell. \end{cases}$$

Now by construction, the vertex v_ℓ is a cut vertex of the graph G, splitting it into H_ℓ and G'. We may therefore apply Theorem 3.2, to get that $c \in \mathrm{Rec}_{v_\rho}(G)$ if, and only if, $c^{(\ell)} \in \mathrm{Rec}_{v_\ell}(H_\ell)$ and $c' \in \mathrm{Rec}_{v_\rho}(G')$. But the graph G' is constructed on a rooted tree with $|\mathsf{nodes}(T)| - 1 = k$ nodes, so that we may apply the induction hypothesis to c', which combined with the above yields the desired result. □

Similarly to Corollary 3.4 in the cut vertex case, this yields a decomposition of the level polynomial for graphs with a tree-like structure.

Corollary 3.10. *Let $G = \bigcup_{t \in \mathsf{nodes}(T)} H_t$ be constructed on a rooted tree (T, ρ). Then we have*

$$\mathrm{Level}_{G,v_\rho}(x) = \prod_{t \in \mathsf{nodes}(T)} \mathrm{Level}_{H_t,v_t}(x).$$

4. Recurrent states on graphs with isthmuses

Having dealt with the cut vertex case, it is natural to ask ourselves what happens when the graph has an *isthmus*, an edge whose removal disconnects the graph.

Theorem 4.1. *Let $G = (V, E)$ be a finite, connected graph on the vertex set*

$$V = \{v_0, v_1, \dots, v_{n+m}\}$$

where $m, n \geq 1$. Assume that the edge $e = (v_n, v_{n+1})$ is an isthmus. Let $H_1 := G[V_{[0,n]}]$ and $H_2 := G[V_{[n+1,n+m]}]$. Let $c = (c_1, \dots c_{n+m}) \in \mathrm{Config}_{v_0}(G)$. Define $c^{(1)} := (c_1, \dots, c_{n-1}, c_n - 1)$ and $c^{(2)} := (c_{n+2}, \dots, c_{n+m})$. Then $c \in \mathrm{Rec}_{v_0}(G)$ if and only if $c_{n+1} = d_{n+1} - 1$, $c^{(1)} \in \mathrm{Rec}_{v_0}(H_1)$, and $c^{(2)} \in \mathrm{Rec}_{v_{n+1}}(H_2)$.

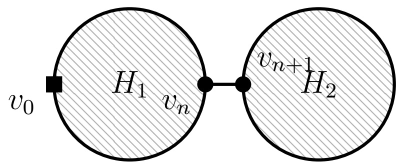


Figure 8. A graph G with an isthmus (v_n, v_{n+1}).

It is worth noting that Theorem 4.1 was proved by Lopez [18] in the special case where the isthmus is adjacent to the sink, as part of his proof of the Specification Theorem 1.3.

As in the cut vertex case, this theorem allows us to infer the following decomposition for the level polynomial.

Corollary 4.2. *Let $G = (V, E)$ be a finite connected graph on the vertex set*

$$V = \{v_0, v_1, \dots, v_{n+m}\}$$

where $m, n \geq 1$. Assume that the edge $e = (v_n, v_{n+1})$ is an isthmus of G. Let $H_1 := G[V_{[0,n]}]$ and $H_2 := G[V_{[n+1,n+m]}]$. Then we have:

$$\mathrm{Level}_{G,v_0}(x) = \mathrm{Level}_{H_1,v_0}(x) \cdot \mathrm{Level}_{H_2,v_{n+1}}(x).$$

This can also be inferred from Theorem 1.3. Indeed, if e is an isthmus, then $T_G(x,y) = x \cdot T_{G.e}(x,y)$, where $G.e$ is the graph G with the edge e contracted (reduced to a single vertex). In the case of the level polynomial, this yields $\mathrm{Level}_{G,v_0}(x) = \mathrm{Level}_{G.e,v_0}(x)$. Since contracting an isthmus creates a cut vertex at the contracted vertex, we can then apply Corollary 3.4 to get the desired result.

Rather than proving Theorem 4.1, we prove a generalization to the case where the edge (v_n, v_{n+1}) is a multi-edge. That is, the graph G is formed of two components H_1, H_2 connected by k copies of the same edge, as illustrated in Figure 9 below. We call such an edge a k-*multi-isthmus*.

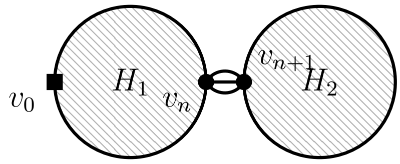


FIGURE 9. A graph G with a 3-multi-isthmus.

Theorem 4.3. *Let $G = (V, E)$ be a finite, connected graph on the vertex set*

$$V = \{v_0, v_1, \ldots, v_{n+m}\}$$

where $m, n \geq 1$. Assume that the edge $e = (v_n, v_{n+1})$ is a k-multi-isthmus for some $k \geq 1$. Let $H_1 := G[V_{[0,n]}]$ and $H_2 := G[V_{[n+1,n+m]}]$. Let $c = (c_1, \ldots c_{n+m}) \in \mathrm{Config}_{v_0}(G)$. Define $c^{(1)} := (c_1, \ldots, c_{n-1}, c_n - k)$ and $c^{(2)} := (c_{n+2}, \ldots, c_{n+m})$. Then we have $c \in \mathrm{Rec}_{v_0}(G)$ if and only if $c^{(1)} \in \mathrm{Rec}_{v_0}(H_1)$, $c^{(2)} \in \mathrm{Rec}_{v_{n+1}}(H_2)$, and

$$d_{n+1}^{H_2} = d_{n+1} - k \leq c_{n+1} \leq d_{n+1} - 1.$$

Corollary 4.4. *Let $G = (V, E)$ be a finite, connected graph on the vertex set*

$$V = \{v_0, v_1, \ldots, v_{n+m}\}$$

where $m, n \geq 1$. Assume that the edge $e = (v_n, v_{n+1})$ is a k-multi-isthmus for some $k \geq 1$. Let $H_1 := G[V_{[0,n]}]$ and $H_2 := G[V_{[n+1,n+m]}]$. Then we have

$$\mathrm{Level}_{G,v_0}(x) = (1 + x + \ldots + x^{k-1}) \cdot \mathrm{Level}_{H_1,v_0}(x) \cdot \mathrm{Level}_{H_2,v_{n+1}}(x).$$

Since Theorem 4.1 and Corollary 4.2 correspond to Theorem 4.3 and Corollary 4.4 in the special case $k = 1$, it is sufficient to prove the latter two results.

Proof. Define $I := G[V_{[n,n+1]}]$ to be the subgraph of G consisting of the multi-isthmus (v_n, v_{n+1}). Since v_n and v_{n+1} are both cut-vertices, the graph G has a cv-tree structure where the blocks are H_1, I, H_2 and the underlying tree is a path graph. The set $\mathrm{Rec}_{v_n}(I)$ of recurrent states of the block I is straightforward to compute:

$$c = (c_{n+1}) \in \mathrm{Rec}_{v_n}(I) \text{ if, and only if, } c_{n+1} \in \{0, \ldots, k-1\}, \tag{16}$$

and Theorem 4.3 immediately follows from this and Theorem 3.9.

Moreover, from Equation (16), we get that $\mathrm{Level}_{I,v_n}(x) = 1 + x + \cdots + x^{k-1}$, and applying Corollary 3.10 gives us Corollary 4.4, as desired. $\square$

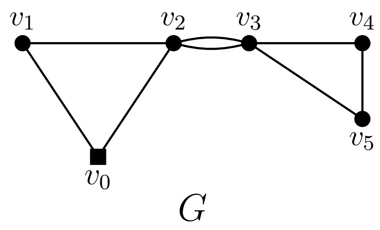


FIGURE 10. A graph G containing a 2-multi-isthmus (v_2, v_3).

Example 4.5. Consider the graph G of Figure 10. This graph can be decomposed into two components H_1 and H_2, connected by the 2-multi-isthmus (v_2, v_3). By Theorem 4.3, the recurrent configurations are given by the products of recurrent configurations on H_1 and H_2 as in the cut vertex case, which we complete by putting either two or three grains of sand at the vertex v_3. We show this in Table 3, and also compute the corresponding level polynomials.

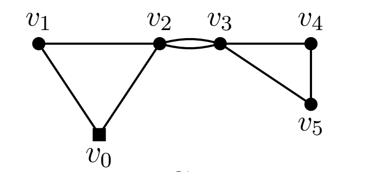
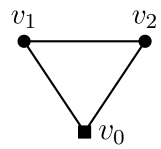
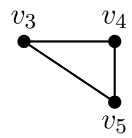
Graph	 G	 H_1	 H_2
Recurrent configurations	$(0,3,2,0,1), (0,3,3,0,1)$	$(0,1)$	$(0,1)$
	$(1,2,2,0,1), (1,2,3,0,1)$	$(1,0)$	$(0,1)$
	$(1,3,2,0,1), (1,3,3,0,1)$	$(1,1)$	$(0,1)$
	$(0,3,2,1,0), (0,3,3,1,0)$	$(0,1)$	$(1,0)$
	$(1,2,2,1,0), (1,2,3,1,0)$	$(1,0)$	$(1,0)$
	$(1,3,2,1,0), (1,3,3,1,0)$	$(1,1)$	$(1,0)$
	$(0,3,2,1,1), (0,3,3,1,1)$	$(0,1)$	$(1,1)$
	$(1,2,2,1,1), (1,2,3,1,1)$	$(1,0)$	$(1,1)$
	$(1,3,2,1,1), (1,3,3,1,1)$	$(1,1)$	$(1,1)$
Level polynomial	$x^3 + 5x^2 + 8x + 4$ $= (1+x)\cdot(x+2)\cdot(x+2)$	$x+2$	$x+2$

TABLE 3. Applying Theorem 4.3 to determine the recurrent configurations of G.

5. DECOMPOSITION OF THE SANDPILE GROUP

In this section we look at how the decomposition results from Section 3 affect the so-called sandpile group.

Let (G, v_0) be a graph, with vertex set $V = \{v_0, \ldots, v_n\}$ and edge set E. Recall that $\sigma : \mathrm{Config}_{v_0}(G) \to \mathrm{Stable}_{v_0}(G)$ denotes the stabilization operator. We define a binary operation $\oplus$ on $\mathrm{Stable}_{v_0}(G)$ by:

$$\text{for all } c, c' \in \mathrm{Stable}_{v_0}(G),\ c \oplus c' := \sigma(c + c'), \tag{17}$$

where $+$ denotes pointwise addition in $\text{Stable}_{v_0}(G)$. Dhar showed in [11] that $(\text{Rec}_{v_0}(G), \oplus)$ is an Abelian group, called the *sandpile group*. We denote it $S(G, v_0)$.

For the proofs of this section, we will need the following characterization of recurrent states, due to Dhar [11]. We first define a configuration $b \in \text{Config}_{v_0}(G)$ by $b := \sum_{i:\{v_i,v_0\}\in E} \alpha_i$, where the sum is over all vertices adjacent to the sink, counted with multiplicity. Thus the configuration b corresponds to the toppling of the sink in an initially empty configuration.

Proposition 5.1. *A configuration $c \in \text{Config}_{v_0}(G)$ is recurrent if, and only if, $\sigma(c+b) = c$. Moreover, in the stabilization of $c+b$ every vertex in $V_{[1,n]}$ topples exactly once.*

Theorem 5.2. *Let $G = (V, E)$ be a finite, connected graph, on the vertex set*

$$V = \{v_0, v_1, \ldots, v_{n+m}\}$$

where $m, n \geq 1$. Assume that v_n is a cut vertex of G. Let $H_1 := G[V_{[0,n]}]$ and $H_2 := G[V_{[n,n+m]}]$. We have:

$$S(G, v_0) \;\cong\; S(H_1, v_0) \times S(H_2, v_n),$$

where the symbol $\cong$ denotes group isomorphism, and $\times$ is the direct product of two groups.

We require two lemmas in order to prove this theorem. Given $c = (c_1, \ldots, c_{n+m}) \in \text{Config}_{v_0}(G)$, let $c^{(1)} := (c_1, \ldots, c_{n-1}, c_n - d_n^{H_2})$ and $c^{(2)} := (c_{n+1}, \ldots, c_{n+m})$, where $d_n^{H_2}$ is the degree of the vertex v_n in the graph H_2.

Lemma 5.3. *For any configurations $c, c' \in \text{Rec}_{v_0}(G)$, we have*

$$(c \oplus c')^{(2)} = c^{(2)} \oplus c'^{(2)}. \tag{18}$$

Proof. Let $c, c' \in \text{Rec}_{v_0}(G)$. Then $c \oplus c'$ is the stabilization of the configuration $c + c'$. Let us examine how this stabilization might occur. We may choose to first stabilize the configuration in H_2 except perhaps at v_n, by making only topplings at vertices in $V_{[v_{n+1},\ldots,v_{n+m}]}$. We will reach a configuration γ where all these vertices are stable. The configuration γ, restricted to $V_{[v_{n+1},\ldots,v_{n+m}]}$, is exactly the same configuration we would reach in the graph (H_2, v_n) by stabilizing the configuration $c^{(2)} + c'^{(2)}$, that is, the configuration $c^{(2)} \oplus c'^{(2)}$.

Now of all the subsequent topplings to stabilize $c+c'$, only the topplings of the cut vertex v_n will effect the configuration on $V_{[v_{n+1},\ldots,v_{n+m}]}$. But the effect of toppling v_n on these vertices is exactly the effect of toppling the sink on the graph (H_2, v_n). Since the configuration reached on (H_2, v_n) before any such toppling is $c^{(2)} \oplus c'^{(2)}$, which is recurrent, toppling the sink and then stabilizing leaves this configuration unchanged by Proposition 5.1. This completes the proof of Lemma 5.3. □

Lemma 5.4. *For any configurations $c, c' \in \text{Rec}_{v_0}(G)$, we have*

$$(c \oplus c')^{(1)} = c^{(1)} \oplus c'^{(1)} \oplus \kappa, \tag{19}$$

where $\kappa = \kappa\left(c^{(2)}, c'^{(2)}\right) := \left(d_n^{H_2} + |c^{(2)}| + |c'^{(2)}| - |c^{(2)} \oplus c'^{(2)}|\right)\alpha_n$, with $|c|$ designating the total number of grains of a configuration c, and α_n is the configuration with one grain at the vertex v_n and none elsewhere.

Proof. A crucial remark that stems from the proof of Lemma 5.3 is the following. When stabilizing the configuration $c + c'$, if the configuration is initially stabilized in H_2, then the topplings of the cut vertex v_n have no effect in H_2.

Now once we have initially stabilized $c + c'$ in H_2, we may stabilize it in H_1. The configuration to stabilize is thus the configuration $c + c'$ restricted to H_1, to which a certain number λ of grains have been added at v_n. These grains are added as a result of the original stabilization in H_2, they are the *surplus* of grains toppled into v_n through stabilizing $(c + c')^{(2)}$. Thus $\lambda = |c^{(2)}| + |c'^{(2)}| - |(c \oplus c')^{(2)}|$, and we have

$$(c \oplus c')|_{H_1} = c|_{H_1} \oplus c'|_{H_1} \oplus \lambda\alpha_n.$$

Equation (19) with $\kappa = \left(\lambda + d_n^{H_2}\right) \alpha_n$ follows from the above, with the definition $c_n^{(1)} = c_n + d_n^{H_2}$ resulting in the adjustment by an additive constant. □

Proof of Theorem 5.2. Lemmas 5.3 and 5.4 allow us to understand the operator $\oplus$ on $\mathrm{Rec}_{v_0}(G)$, and in particular its behaviour when restricted to the components H_1, H_2.

The result then follows from a group theory argument that we will now explain. Lemmas 5.3 and 5.4, combined with the Decomposition Theorem 3.2, mean that we have the following situation. We have a set $\mathcal{G} = \mathcal{H}_1 \times \mathcal{H}_2$, where $\mathcal{G}$,H_1 and $\mathcal{H}_2$ are all equipped with an Abelian group structure. However, the binary addition operation in $\mathcal{G}$ does not decompose to product addition in $\mathcal{H}_1$ and $\mathcal{H}_2$. Rather, for $(c_1, c_2), (c'_1, c'_2) \in \mathcal{G}$, we have

$$(c_1, c_2) + (c'_1, c'_2) = (c_1 + c'_1 + \kappa(c_2, c'_2), c_2 + c'_2),$$

where the function κ is described by Lemma 5.4.

Now from [7, Chapter IV] we get that such a group $\mathcal{G}$ is isomorphic to the direct product $\mathcal{H}_1 \times \mathcal{H}_2$ if there exists a function $a : \mathcal{H}_2 \to \mathcal{H}_1$ such that for any $c_2, c'_2 \in \mathcal{H}_2$, we have $\kappa(c_2, c'_2) = a(c_2) + a(c'_2) - a(c_2 + c'_2)$. In our case, simply setting $a(c_2) = \left(|c_2| + d_n^{H_2}\right) \alpha_n$ immediately yields the desired result. □

Remark 5.5. In fact, we make a slight abuse when using the group theory arguments in this case, since the configurations $\kappa, a(c_2)$ are not in general recurrent configurations on H_1. However, this difficulty can be overcome by considering the unique recurrent configurations to which they are toppling equivalent (see [10, Theorem 1]) instead.

Example 5.6. Consider the graph G of Figure 11, where the vertex v_2 is a cut vertex splitting the graph into two isomorphic components H_1, H_2. In Example 1.4, we showed that the components H_i have three recurrent configurations, which implies that $S(H_1, v_0) \cong S(H_2, v_2) \cong \mathbb{Z}/3\mathbb{Z}$. It immediately follows from Theorem 5.2 that

$$S(G, v_0) \cong \mathbb{Z}/3\mathbb{Z} \times \mathbb{Z}/3\mathbb{Z}.$$

Note that this is not a straightforward consequence of simply enumerating the recurrent configurations of G, since an Abelian group with nine elements could also be (isomorphic to) $\mathbb{Z}/9\mathbb{Z}$.

As with the transition from Theorem 3.2 to Theorem 3.9, an induction argument allows us to extend Theorem 5.2 to the case of graphs with an underlying cv-tree structure.

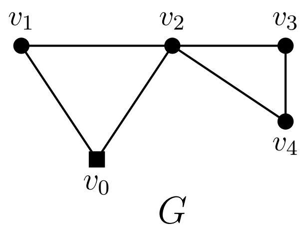


FIGURE 11. A graph G with its cut vertex v_2.

Theorem 5.7. *Let* $G = \bigcup_{t \in \mathsf{nodes}(T)} H_t$ *have an underlying cv-tree structure* (T, ρ). *We have:*

$$S(G, v_\rho) \cong \prod_{t \in \mathsf{nodes}(T)} S(H_t, v_t),$$

where $\cong$ *denotes group isomorphism, and* $\prod$ *the direct product of the groups involved.*

Remark 5.8. It is not clear that the result of Theorem 4.1 can be cast in a useful manner in terms of the sandpile group. The enumeration result in Corollary 4.2 suggests a decomposition of the form $S(G^{(k)}) = S(G) \times \mathbb{Z}_k^{n-1}$, but there are some simple examples for which this both holds and does not hold.

6. EXAMPLES

In this part, we exhibit two examples which showcase the results of this paper. They illustrate the computational gains that are made with respect to determining the recurrent configurations on a graph that can be decomposed in the terms we have described.

Example 6.1. Let G be the graph in Figure 12. Let us consider the sandpile model on G wherein vertex v_0 is the sink. Apply Theorem 3.9 to this graph. The cv-tree $(T, 0)$ structure of G is illustrated in Figure 13. The constituent graphs of the cv-tree are

$$H_i = \begin{cases} G_{[\{v_i, v_{4i+1}, v_{4i+2}, v_{4i+3}, v_{4i+4}\}]} & \text{if } 0 \leq i \leq 4 \\ G_{[\{v_i, v_{3i+6}, v_{3i+7}, v_{3i+8}\}]} & \text{if } 5 \leq i \leq 20, \end{cases}$$

and $\mathsf{nodes}(T) = \{0, 1, \ldots, 20\}$. Note that H_i is isomorphic to K_5 (respectively K_4) for all $i \leq 4$ (respectively $5 \leq i \leq 20$). The recurrent states of the sandpile model on these graphs are therefore characterized by Theorem 1.7.

Theorem 3.9 tells us that $c = (c_1, \ldots, c_{68}) \in \mathrm{Rec}_{v_0}(G)$ if, and only if, $c^{(t)} \in \mathrm{Rec}_{v_t}(H_t)$ for all $0 \leq t \leq 20$. This latter condition is equivalent to:

$$\begin{cases} (c^{(i)}_{4i+1}, c^{(i)}_{4i+2}, c^{(i)}_{4i+3}, c^{(i)}_{4i+4}) \in \mathrm{Rec}_{v_0}(K_5) & \text{for all } 0 \leq i \leq 4 \\ (c^{(i)}_{3i+6}, c^{(i)}_{3i+7}, c^{(i)}_{3i+8}) \in \mathrm{Rec}_{v_0}(K_4) & \text{for all } 5 \leq i \leq 20. \end{cases}$$

The set $\mathrm{Rec}_{v_0}(K_4)$ is explicitly listed in Equation (9), and the set $\mathrm{Rec}_{v_0}(K_5)$ is described in Example 1.9.

Now since $d^{H_i}(v_i) = 4$ for $1 \leq i \leq 4$, and $d^{H_i}(v_i) = 3$ for all $5 \leq i \leq 20$, we may conclude with our classification of recurrent states of the sandpile model on G: $c \in \mathrm{Rec}_{v_0}(G)$ if and only if

(i) $(c_1 - 4, c_2 - 4, c_3 - 4, c_4 - 4) \in \mathrm{Rec}_{v_0}(K_5)$,
(ii) $(c_{4i+1} - 3, c_{4i+2} - 3, c_{4i+3} - 3, c_{4i+4} - 3) \in \mathrm{Rec}_{v_0}(K_5)$ for all $1 \leq i \leq 4$, and
(iii) $(c_{3i+6}, c_{3i+7}, c_{3i+8}) \in \mathrm{Rec}_{v_0}(K_4)$ for all $5 \leq i \leq 20$.

We may also compute the level polynomial for the graph G of Figure 12. By Corollary 3.10, we have:

$$\mathrm{Level}_{G,v_0}(x) = \mathrm{Level}_{K_5,v_0}(x)^5 \, \mathrm{Level}_{K_4,v_0}(x)^{16}.$$

The level polynomials of the complete graphs can be computed for instance by computing their Tutte polynomials and applying Theorem 1.3. We get

$$\begin{aligned}\mathrm{Level}_{K_4,v_0}(x) &= x^3 + 3x^2 + 6x + 6 \\ \mathrm{Level}_{K_5,v_0}(x) &= x^6 + 4x^5 + 10x^4 + 20x^3 + 30x^2 + 36x + 24,\end{aligned}$$

and thus

$$\begin{aligned}\mathrm{Level}_{G,v_0}(x) &= (x^6 + 4x^5 + 10x^4 + 20x^3 + 30x^2 + 36x + 24)^5 (x^3 + 3x^2 + 6x + 6)^{16} \\ &= x^{78} + 68x^{77} + 2346x^{76} + \ldots + 22463437455746924544.\end{aligned}$$

In other words, there is one recurrent configuration with level 78, 68 recurrent configurations with level 77, and so on.

Example 6.2. Let G be the graph in Figure 14. Let us consider the sandpile model on G wherein vertex v_0 is the sink. We apply Theorems 3.9 and 4.1 to this graph. The cv-tree $(T, 0)$ structure of G and its constituent graphs are illustrated in Figure 15, where H_0 and H_1 are connected by an isthmus. Note that H_0 and H_2 are isomorphic to K_5, H_3 is isomorphic to $K_4^{(2)}$, that is K_4 with all edges doubled, and that H_1 is a graph whose recurrent configurations have already been determined in Example 3.6.

Theorems 3.9 and 4.1 tell us that $c = (c_1, \ldots, c_{14}) \in \mathrm{Rec}_{v_0}(G)$ if, and only if, $c^{(t)} \in \mathrm{Rec}_{v_t}(H_t)$ for all $0 \leq t \leq 3$, and $c_5 = 3$, since v_5 is the vertex of the isthmus (v_3, v_5) "furthest" from the sink, and has degree four. This latter condition is equivalent to:

- $(c_1^{(0)}, c_2^{(0)}, c_3^{(0)}, c_4^{(0)}), (c_8^{(2)}, c_9^{(2)}, c_{10}^{(2)}, c_{11}^{(2)}) \in \mathrm{Rec}_{v_0}(K_5)$ (20)
- $c_5 = 3$ (21)
- $(c_6^{(1)}, c_7^{(1)}) \in \mathrm{Rec}_{v_5}(H_1)$ (22)
- $(c_{12}^{(3)}, c_{14}^{(3)}, c_{15}^{(3)}) \in \mathrm{Rec}_{v_0}\left(K_4^{(2)}\right).$ (23)

The set $\mathrm{Rec}_{v_0}(K_5)$ is described in Example 1.9. Since $d^{H_1}(v_3) = 1$ and $d^{H_3}(v_4) = 6$, Equation (20) is equivalent to:

$$(c_1, c_2, c_3 - 1, c_4 - 6), (c_8, c_9, c_{10}, c_{11}) \in \mathrm{Rec}_{v_0}(K_5).$$

We have already computed the recurrent configurations of the component H_1 in Example 3.6, so $\mathrm{Rec}_{v_5}(H_1) = \big\{(0,1), (1,0), (1,1), (2,0), (2,1)\big\}$. Since $d^{H_2}(v_7) = 4$, Equation (22) is equivalent to:

$$(c_6, c_7 - 4) \in \mathrm{Rec}_{v_5}(H_1).$$

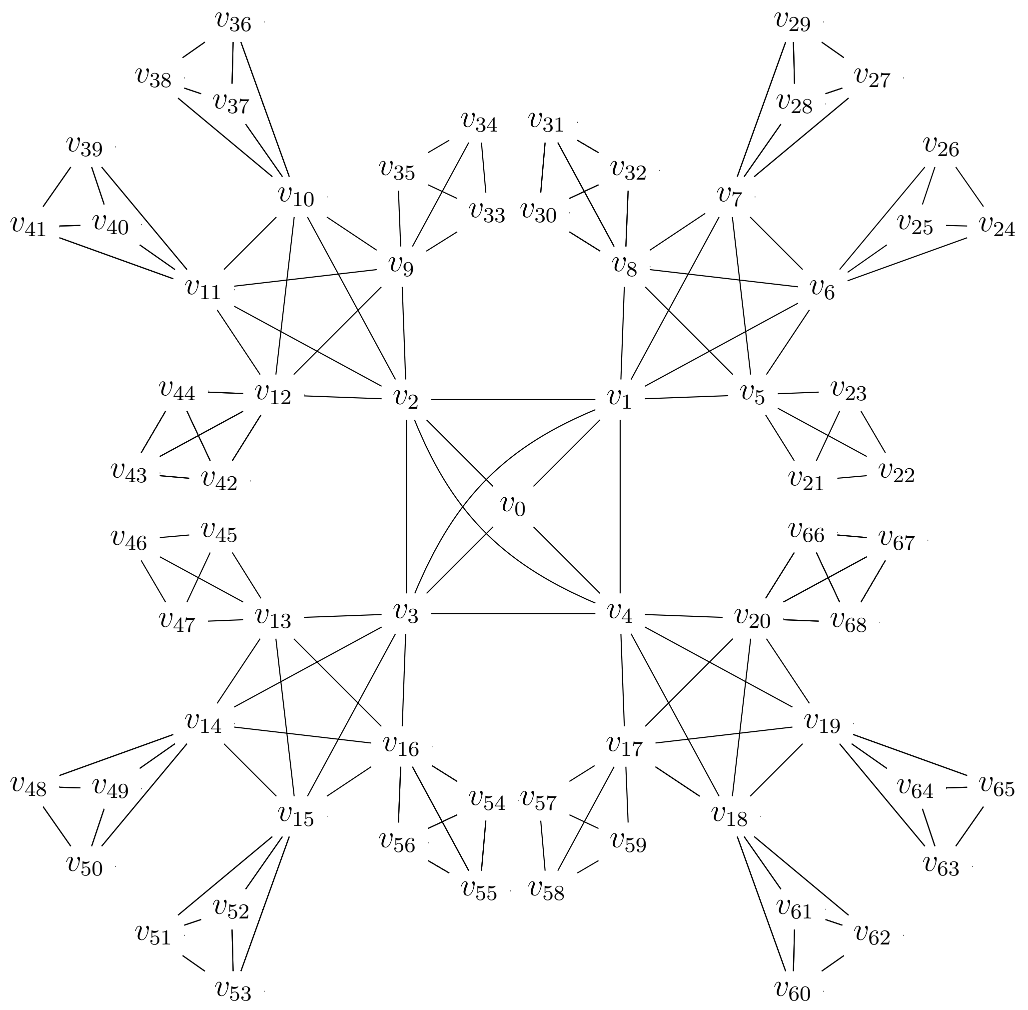


FIGURE 12. The graph used in Example 6.1.

Finally, the set $\mathrm{Rec}_{v_0}(K_4)$ is explicitly listed in Equation 9. The recurrent states of $K_4^{(2)}$ are linked to those of K_4 by Theorem 2.1, so that Equation (23) is equivalent to:

$$(\lfloor c_{12}/2 \rfloor, \lfloor c_{13}/2 \rfloor, \lfloor c_{14}/2 \rfloor) \in \mathrm{Rec}_{v_0}(K_4).$$

Combining all this, we may conclude with our classification of recurrent states of the sandpile model on G: $c \in \mathrm{Rec}_{v_0}(G)$ if and only if

- $(c_1, c_2, c_3 - 1, c_4 - 6), (c_8, c_9, c_{10}, c_{11}) \in \mathrm{Rec}_{v_0}(K_5)$,
- $c_5 = 3$,
- $(c_6, c_7 - 4) \in \mathrm{Rec}_{v_5}(H_1)$, and
- $(\lfloor c_{12}/2 \rfloor, \lfloor c_{13}/2 \rfloor, \lfloor c_{14}/2 \rfloor) \in \mathrm{Rec}_{v_0}(K_4)$.

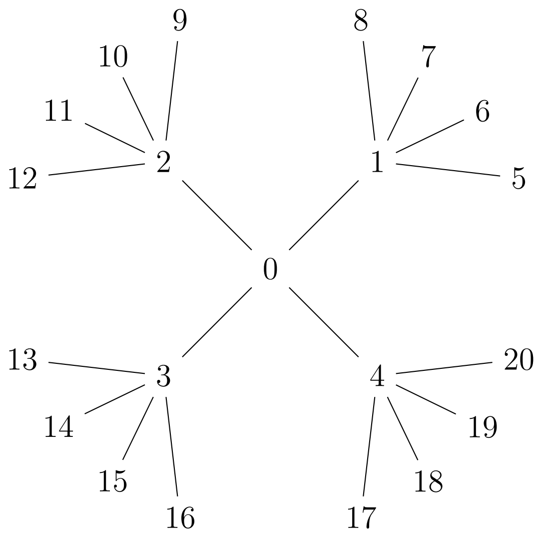


FIGURE 13. The cv-tree $(T, 0)$ of the graph G in Figure 12 and Example 6.1.

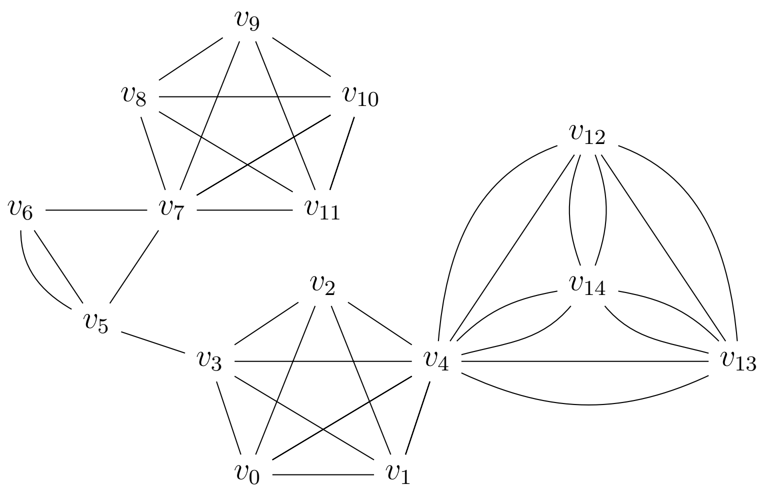


FIGURE 14. A graph G that can be decomposed as previously described.

REFERENCES

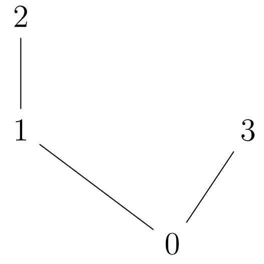
[1] J.-C. Aval, M. D'Adderio, M. Dukes, A. Hicks, and Y. Le Borgne. Statistics on parallelogram polyominoes and a q, t-analogue of the Narayana numbers. *J. Comb. Theory Ser. A* 123(1):271–286, 2014.

[2] J.-C. Aval, M. D'Adderio, M. Dukes, and Y. Le Borgne. Two operators on sandpile configurations, the sandpile model on the complete bipartite graph, and a cyclic lemma. *Adv. Appl. Math.* 73:59–98, 2016.

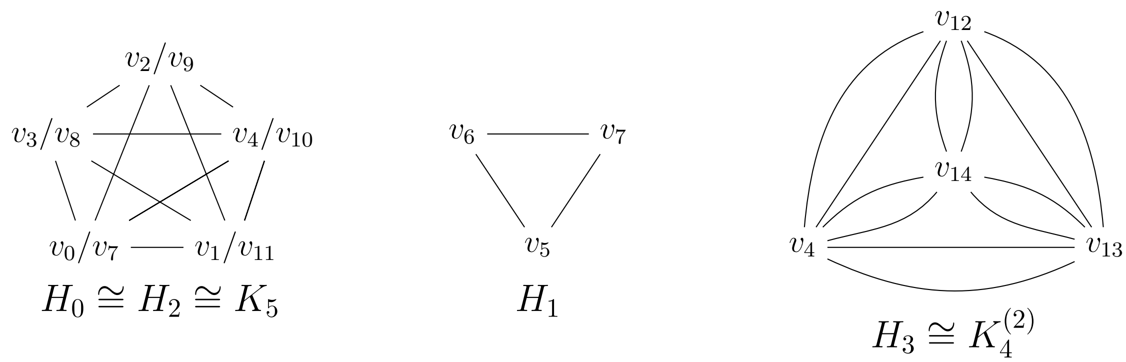
[3] M. Baker and S. Norine. Riemann–Roch and Abel–Jacobi theory on a finite graph. *Adv. Math.* 215(2):766–788, 2007.

[4] A. Berget, A. Manion, M. Maxwell, A. Potechin, and V. Reiner. The critical group of a line graph. *Ann. Combin.* 16:449–488, 2012.

[5] O. Bernardi. Tutte polynomial, subgraphs, orientations and sandpile model: new connections via embeddings. *Electron. J. Comb.* 15(1):#R109, 53 pp., 2008.



(a) The cv-tree structure of the graph in Figure 14.



(b) The constituent graphs of the cv-tree for Figure 14. In the pentagram on the left, graph H_0 consists of those labels to the left of the '/' and graph H_2 consists of those labels to the right of '/'. Both are isomorphic to K_5.

FIGURE 15. Decomposition of the graph G from Figure 14.

[6] N. Biggs. The Tutte Polynomial as a Growth Function. *J. Algebraic Combin.* 10(2):115–133, 1999.

[7] K. Brown. Cohomology of groups, volume 87 of *Graduate Texts in Mathematics*. Springer-Verlag, New York-Berlin, 1982.

[8] R. Cori and Y. Le Borgne. The sand-pile model and Tutte polynomials. *Adv. Appl. Math.* 30(1-2):44–52, 2003.

[9] R. Cori and D. Poulalhon. Enumeration of (p, q)-parking functions. *Discrete Math.* 256(3):609–623, 2002. LaCIM 2000 Conference on Combinatorics, Computer Science and Applications.

[10] R. Cori and D. Rossin. On the sandpile group of a graph. *Eur. J. Combin.* 21:447–459, 2000.

[11] D. Dhar. Self-organized critical state of sandpile automaton models. *Phys. Rev. Lett.* 64:1613–1616, 1990.

[12] D. Dhar. Theoretical studies of self-organized criticality. *Phys. A* 369(1):29–70, 2006.

[13] M. Dukes and Y. Le Borgne. Parallelogram polyominoes, the sandpile model on a complete bipartite graph, and a q, t-Narayana polynomial. *J. Comb. Theory Ser. A* 120(4):816–842, 2013.

[14] C. Greene and T. Zaslavsky. On the interpretation of Whitney numbers through arrangements of hyperplanes, zonotopes, non-Radon partitions, and orientations of graphs. *Trans. Amer. Math. Soc.* 280(1):97–126, 1983.

[15] B. Jacobson, A. Niedermaier, and V. Reiner. Critical groups for complete multipartite graphs and Cartesian products of complete graphs. *J. Graph Theory* 44:231–250, 2003.

[16] L. Levine. Sandpile groups and spanning trees of directed line graphs. *J. Combin. Theory Ser. A* 118(2):350–364, 2011.

[17] D. Lorenzini. A finite group attached to the laplacian of a graph. *Discrete Math.* 91:277–282, 1991.
[18] C. Merino. Chip firing and the Tutte polynomial. *Ann. Comb.* 1(1):253–259, 1997.
[19] F. Redig. Mathematical aspects of the Abelian sandpile model. *Lecture Notes of Les Houches Summer School 2005*, Mathematical Statistical Physics, Session LXXXIII. Elsevier, 2006.
[20] K. Sekine and C.Q. Zhang. Decomposition of the Flow Polynomial. *Graphs Combin.* 13(2):189–196, 1997.
[21] W.T. Tutte. A contribution to the theory of chromatic polynomials. *Can. J. Math.* 6:80–91, 1954.

School of Mathematics and Statistics, University College Dublin, Dublin 4, Ireland.

Department of Computer and Information Sciences, University of Strathclyde, Glasgow, G1 1XH, UK