Forum of Mathematics, Sigma (2023), **Vol. 11**:e78 1–40
doi:10.1017/fms.2023.77

CAMBRIDGE
UNIVERSITY PRESS

RESEARCH ARTICLE

Rotor-Routing Induces the Only Consistent Sandpile Torsor Structure on Plane Graphs

Ankan Ganguly[1] and Alex McDonough[2]

[1]Department of Mathematics & Statistics, Boston University, Boston, MA, USA; E-mail: ankang@bu.edu.
[2]Department of Mathematics, University of California, Davis, CA, USA; E-mail: amcd@ucdavis.edu.

Received: 4 April 2022; **Revised:** 16 June 2023; **Accepted:** 3 August 2023

2020 Mathematics Subject Classification: *Primary* – 05E18, 05C10

Abstract

We make precise and prove a conjecture of Klivans about actions of the sandpile group on spanning trees. More specifically, the conjecture states that there exists a unique 'suitably nice' sandpile torsor structure on plane graphs which is induced by rotor-routing.

First, we rigorously define a sandpile torsor algorithm (on plane graphs) to be a map which associates each plane graph (i.e., planar graph with an appropriate ribbon structure) with a free transitive action of its sandpile group on its spanning trees. Then, we define a notion of consistency, which requires a torsor algorithm to be preserved with respect to a certain class of contractions and deletions. Using these definitions, we show that the rotor-routing sandpile torsor algorithm is consistent. Furthermore, we demonstrate that there are only three other consistent algorithms on plane graphs, which all have the same structure as rotor-routing.

We also define sandpile torsor algorithms on regular matroids and suggest a notion of consistency in this context. We conjecture that the Backman-Baker-Yuen algorithm is consistent, and that there are only three other consistent sandpile torsor algorithms on regular matroids, all with the same structure.

Contents

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

1. Introduction

Let G be a finite connected multigraph. The *sandpile group* of G, which we denote $\mathrm{Pic}^0(G)$, is a finite abelian group given by the cokernal of the *graph Laplacian matrix* over the integers. A remarkable fact, which follows from Kirchhoff's matrix-tree theorem, is that the size of $\mathrm{Pic}^0(G)$ is equal to the number of *spanning trees* of G (see [Big99, Theorem 7.3]). The traditional proof of this result is non-bijective, and for many graphs, no automorphism invariant bijections exist unless G is given additional structure (see [Wag00, Theorem 8.1]). This problem is often resolved by working with *ribbon graphs*.

Let χ specify a cyclic ordering on the edges adjacent to each vertex of G. The pair (G, χ) is called a *ribbon graph*. Let $\mathcal{T}(G)$ be the set of *spanning trees* of G. Holroyd et al. demonstrated that the *rotor-routing algorithm* induces a *free transitive group action* of $\mathrm{Pic}^0(G)$ on $\mathcal{T}(G)$, which depends on the ribbon structure χ and a choice of *sink* vertex [HLM+08]. In particular, this makes $\mathcal{T}(G)$ a $\mathrm{Pic}^0(G)$-*torsor*, so we will call such an action a *sink-parameterized sandpile torsor action* on (G, χ).

On a MathOverflow post, Ellenberg asked if there exist certain classes of ribbon graphs for which natural sandpile torsor actions can be defined without requiring a sink vertex [Ell11]. Chan, Church and Grochow answered this question by showing that the rotor-routing sandpile torsor action on (G, χ) is independent of the sink vertex if and only if (G, χ) is a *plane graph* [CCG14, Theorem 2].[1] With this result in mind, we define a *sandpile torsor algorithm (on plane graphs)* to be a function which assigns a sandpile torsor action to every plane graph (G, χ) independent of the choice of sink vertex.

Another sink-parameterized sandpile torsor action is given by the *Bernardi algorithm* [Ber08], which is distinct from rotor-routing on nonplanar ribbon graphs. Baker and Wang showed that the restriction of the Bernardi algorithm to plane graphs is also a sandpile torsor algorithm [BW17, Theorem 5.1]. Furthermore, they showed that this sandpile torsor algorithm is equivalent to the rotor-routing algorithm (when restricted to plane graphs) [BW17, Theorem 7.1]. This surprising equivalence motivated the following conjecture by Klivans, which we prove in this paper.

Conjecture 1.1. *[Kli18, Conjecture 4.7.17] For plane graphs, there is only one sandpile torsor structure.*

One of the challenges in resolving this conjecture is that one must introduce a reasonable definition of *sandpile torsor structure* on plane graphs. The rotor-routing sandpile torsor algorithm follows a simple set of rules that can be applied to any plane graph. With this in mind, we posit that any definition of *sandpile torsor structure* should include some notion of consistency of induced sandpile torsor actions between different plane graphs. However, it is difficult to find a suitable consistency condition because there is no general map between the sandpile groups of different graphs which preserves their structure. The key insight that motivated the writing of this paper is the formulation of a contraction-deletion based definition of consistency (see Definition 4.3 and Figure 2).

There are two main results in this paper. In Theorem 4.6, we show that rotor-routing is *consistent*. In Theorem 5.4, we show that there are only three other consistent sandpile torsor algorithms on plane graphs, which can be obtained from rotor-routing by reversing the cyclic order on the vertices, taking the inverse action, or both. We say that these algorithms all have the same *structure* as rotor-routing. One powerful tool that we introduce is Theorem 5.8, which implies that on any 2-connected graph, it is possible to transform one spanning tree to any other by repeatedly swapping *leaf edges* with edges not on the tree (Corollary 5.16).

In Section 2, we define relevant terms and introduce the rotor-routing algorithm. In Section 3, we give several properties of rotor-routing that are applied in later sections. In Section 4, we introduce *consistency* and prove that the rotor-routing algorithm is consistent. In Section 5, we show that every other consistent sandpile torsor algorithm on plane graphs has the same structure as rotor-routing. In Section 6, we define sandpile torsor algorithms on *regular matroids* as well as a notion of *consistency* in this context. We conjecture that the Backman-Baker-Yuen matroidal sandpile torsor algorithm (constructed in [BBY19])

[1] Following the language of [BW17], we distinguish between a *plane graph*, which is a particular planar embedding, and a *planar graph* which is a graph where such an embedding exists.

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

is consistent (Conjecture 6.11) and that all other consistent matroidal sandpile torsor algorithms have the same structure (Conjecture 6.14). In Appendix A, we introduce terminology used in [HLM+08] and [CCG14] and provide proofs of the properties presented in Section 3. Finally, Appendix B is devoted to building the framework to prove the final case of Theorem 5.4.

2. Background and definitions

2.1. Divisors, ribbon graphs and the sandpile group

In this section, we use much of the notation from [CCG14]. See [CP18] and [Kli18] for more information on divisors, sandpile groups and chip-firing.

Let G be a finite, connected, undirected graph with vertices $V(G)$ and edges $E(G)$. We allow parallel edges but not loop edges. For an edge $e \in E(G)$, we write $\mathfrak{i}(e)$ for the unordered pair of vertices incident to e (in set notation). We say that edges e and f are *parallel* if $\mathfrak{i}(e) = \mathfrak{i}(f)$. For $v \in V(G)$, we write $\deg(v)$ for the number of edges incident to v. For $x, y \in V(G)$, a *path* P from x to y is a collection of distinct $(e_1, \ldots, e_k) \in E(G)^k$ for some k such that $x \in \mathfrak{i}(e_1)$, $y \in \mathfrak{i}(e_k)$, and

$$|\mathfrak{i}(e_i) \cap \mathfrak{i}(e_j)| = \begin{cases} 2 & \text{if } i = j, \\ 1 & \text{if } |i - j| = 1, \\ 0 & \text{otherwise.} \end{cases}$$

Definition 2.1. The group $\mathrm{Div}(G)$ of *divisors* of G is given by

$$\mathrm{Div}(G) := \left\{ \sum_{v \in V(G)} n_v v \mid n_v \in \mathbb{Z} \right\}.$$

The subgroup $\mathrm{Div}^0(G)$ of *degree-0 divisors*[2] of G is given by

$$\mathrm{Div}^0(G) := \left\{ \sum_{v \in V(G)} n_v v \mid n_v \in \mathbb{Z}, \sum_{v \in V(G)} n_v = 0 \right\}.$$

For any $s \in V(G)$, let

$$\mathrm{Div}^0_s(G) := \left\{ \sum_{v \in V(G)} n_v v \mid n_v \in \mathbb{Z}_{\geq 0} \text{ for } v \neq s, \sum_{v \in V(G)} n_v = 0 \right\}.$$

Note that although $\mathrm{Div}(G)$ and $\mathrm{Div}^0(G)$ are groups, $\mathrm{Div}^0_s(G)$ is a *monoid* because its nonzero elements lack inverses. The vertex s is called the *sink vertex*.

Let A be the adjacency matrix of G (i.e., the $|V(G)| \times |V(G)|$ symmetric matrix with A_{vw} equal to the number of edges between v and w). Let $\deg(G)$ be the $|V(G)| \times |V(G)|$ diagonal matrix such that $\deg(G)_{vv} = \deg(v)$ for every $v \in V(G)$. The *Laplacian matrix* Δ of G is the symmetric matrix defined by $\Delta := \deg(G) - A$.

Definition 2.2. The *sandpile group* of G, denoted $\mathrm{Pic}^0(G)$, is the quotient

$$\mathrm{Pic}^0(G) := \mathrm{Div}^0(G)/\mathrm{im}_{\mathbb{Z}}(\Delta).$$

[2] In general, the *degree* of a divisor is the sum of the coefficients on the vertices and is not related to the degree of the vertices of G. For this paper, all of our divisors will have degree 0, and any future instances of the term *degree* always refer to the degree of a vertex.

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

The sandpile group has been rediscovered in many contexts and is also called the *graph Picard group*, the *graph Jacobian*, the *critical group*, the *group of components* or the *group of bicycles*. One way to explore properties of $\mathrm{Pic}^0(G)$ is in terms of *chip-firing*, also called *the dollar game*, which we describe below.

Each element $D = \sum_{v\in V(G)} n_v v \in \mathrm{Div}^0(G)$ may be viewed as a configuration of 'chips' placed on the vertices of G, also allowing negative 'debt' chips (where the total number of chips matches the total amount of debt). Note that elements of $\mathrm{Div}_s^0(G)$ have all of their debt on the sink vertex. In this context, divisors are also called *chip configurations*. Given a chip configuration D, we may 'fire' a vertex $v \in V(G)$ by removing $\deg(v)$ chips from v and placing one chip on each of the neighbors of v. That is, v 'gifts' a chip to each of its neighbors. For our purposes, we allow a vertex to fire regardless of the number of chips, even if this puts the vertex in debt.

Each row of the Laplacian matrix corresponds to a vertex firing. Thus, the image of the Laplacian over $\mathbb{Z}$ describes an equivalence relation on chip configurations, where D and D' are *firing equivalent* if D' can be obtained from D by a sequence of vertex firings. This means that two chip configurations are firing equivalent precisely when they are both representatives of the same equivalence class of $\mathrm{Pic}^0(G)$.

Throughout this paper, we will put brackets around a divisor to indicate the corresponding equivalence class of $\mathrm{Pic}^0(G)$. As discussed in the previous paragraph, for $D, D' \in \mathrm{Div}^0(G)$, we have $[D] = [D']$, or equivalently $[D-D'] = [0]$, if and only if D and D' are firing equivalent. We consider $D, D' \in \mathrm{Div}_s^0(G)$ to be firing equivalent if they are firing equivalent as elements of $\mathrm{Div}^0(G)$.

We will also use brackets to indicate sets of integers, where $[a, b] := \{z \in \mathbb{Z} \mid a \leq z \leq b\}$. For clarity, we will write $[1, n]$ in place of the more common $[n]$ so that this is not mistaken for an equivalence class.

Lemma 2.3. *For any $s \in V(G)$ and $D \in \mathrm{Div}^0(G)$, there exists some $D' \in \mathrm{Div}_s^0(G)$ such that $[D] = [D']$.*

Proof. Let $\delta \in \mathrm{Div}_s^0(G)$ assign $\deg(v)$ chips to each $v \in V(G) \setminus s$ and $-\sum_{v\neq s} \deg(v)$ chips to s. Let δ° be the divisor we obtain from δ by repeatedly firing any vertices v that have more than $\deg(v)$ many chips until no such vertices exist.[3]

Because δ and δ° are firing equivalent, $[\delta - \delta^\circ] = 0$. Furthermore, by construction, $\delta - \delta^\circ$ has a positive number of chips at every $v \neq s$. Let $-m$ be the minimal number of chips D assigns to any vertex. Then, $D + m(\delta - \delta^\circ) \in \mathrm{Div}_s^0(G)$ and $[D + m(\delta - \delta^\circ)] = [D]$. □

A *spanning tree* of G is a maximal subset of $E(G)$ that contains no cycles. We will write $\mathcal{T}(G)$ for the set of spanning trees of G. The following is a version of Kirchhoff's matrix-tree theorem.

Theorem 2.4 (sandpile matrix-tree theorem for graphs [Big99, Theorem 7.3]).

$$|\mathrm{Pic}^0(G)| = |\mathcal{T}(G)|$$

Theorem 2.4 implies the existence of bijections between equivalence classes of $\mathrm{Pic}^0(G)$ and elements of $\mathcal{T}(G)$. However, in order to define sufficiently natural bijections, we will need to give G some additional structure (see [Wag00] and [CCG14] for further discussion). Figure 1 demonstrates Theorem 2.4 for a particular graph.

Definition 2.5. A *ribbon structure* χ on a graph G is a map from each $v \in V(G)$ to a cyclic order of the edges incident to v. A *ribbon graph* is a pair (G, χ), where G is a graph and χ is a ribbon structure on G.

Ribbon graphs are also called *combinatorial maps*. Suppose that G can be drawn in a plane with no crossings such that for every $v \in V(G)$, $\chi(v)$ gives the edges incident to v in counterclockwise order. Then, (G, χ) is called a *plane graph* or a *planar embedding*. A graph G is called *planar* if there exists some χ such that (G, χ) is a plane graph.

[3] The notation for δ and δ°, as well as the general method of this proof, comes from [HLM+08] (where δ° is called the *stabilization* of δ). The divisor δ° is well-defined by [HLM+08, Lemma 2.2] (see also [Dha90]).

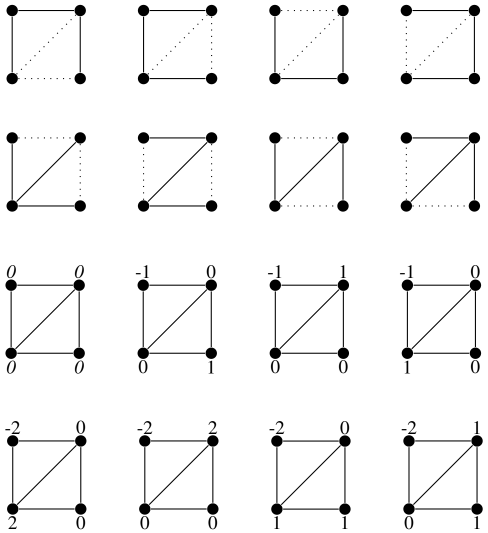

Figure 1. *Above are the eight spanning trees for the graph $K_4 \setminus e$ as well as representatives for the eight elements of* $\mathrm{Pic}^0(K_4 \setminus e)$*.*[4] *By Theorem 2.4, these sets are always the same size. However, there is not a natural choice of bijection between them.*

Suppose that $x \in V(G)$ and $e \in E(G)$ are incident (i.e., $x \in \mathfrak{i}(e)$). We will write $\chi(x, e)$ for the edge after e in the cyclic order $\chi(x)$.

Definition 2.6. Let (G, χ) and (G', χ') be two ribbon graphs. A *ribbon graph isomorphism* is a bijection, $\phi : V(G) \cup E(G) \to V(G') \cup E(G')$ such that $\phi(V(G)) = V(G')$, $\phi(E(G)) = E(G')$, and for any $e \in E(G)$, the following properties are satisfied (where $\mathfrak{i}(e) = \{x, y\}$):

1. $\mathfrak{i}(\phi(e)) = \{\phi(x), \phi(y)\}$.
2. $\phi(\chi(x, e)) = \chi'(\phi(x), \phi(e))$.
3. $\phi(\chi(y, e)) = \chi'(\phi(y), \phi(e))$.

A *ribbon graph automorphism* is a ribbon graph isomorphism from a ribbon graph to itself.

Lemma 2.7. *Any ribbon graph isomorphism ϕ from (G, χ) to (G', χ') induces a bijection from $\mathcal{T}(G)$ to $\mathcal{T}(G')$. Furthermore, ϕ induces an isomorphism ϕ_{D} from* $\mathrm{Div}^0(G)$ *to* $\mathrm{Div}^0(G')$ *which also generates isomorphisms from* $\mathrm{Div}^0_s(G)$ *to* $\mathrm{Div}^0_{\phi(s)}(G')$ *and* $\mathrm{Pic}^0(G)$ *to* $\mathrm{Pic}^0(G')$.

Proof. Let ϕ be the isomorphism in question. The bijection between $\mathcal{T}(G)$ and $\mathcal{T}(G')$ follows from the fact that ϕ induces a graph isomorphism between G and G'. Next, we define ϕ_{D} by the map

$$\sum_{v \in V(G)} n_v v \mapsto \sum_{v' \in V(G')} n_{\phi^{-1}(v')} v'.$$

It is simple to check that ϕ_{D} is an isomorphism between $\mathrm{Div}^0(G)$ and $\mathrm{Div}^0(G')$. Because $\mathrm{Div}^0_s(G)$ and $\mathrm{Div}^0_{\phi(s)}(G')$ are respective subsets of $\mathrm{Div}^0(G)$ and $\mathrm{Div}^0(G')$ which respect group operations, it follows that $\phi_{\mathrm{D}}|_{\mathrm{Div}^0_s(G)}$ is an isomorphism between $\mathrm{Div}^0_s(G)$ and $\mathrm{Div}^0_{\phi(s)}(G')$.

Finally, if Δ and Δ' denote the respective Laplacian matrices of G and G', then ϕ_{D} is also an isomorphism on the subgroups of $\mathrm{Div}^0(G)$ and $\mathrm{Div}^0(G')$ restricted to $\mathrm{im}_{\mathbb{Z}}(\Delta)$ and $\mathrm{im}_{\mathbb{Z}}(\Delta')$, respectively. Thus, ϕ_{D} induces an isomorphism between $\mathrm{Pic}^0(G)$ and $\mathrm{Pic}^0(G')$. □

[4] The representatives are the *superstable divisors* with a particular *sink*. For more information on choosing a distinguished set of representatives for the sandpile group, see [Kli18, Chapter 2].

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

In fact, Lemma 2.7 holds even if ϕ is a graph isomorphism that does not respect the ribbon structures of G and G'. However, we work exclusively with ribbon graph isomorphisms throughout this article due to the fact that the ribbon structures of graphs play a key role in the definition of sandpile torsor actions and algorithms (see the next subsection).

2.2. Sandpile torsor actions and sandpile torsor algorithms

Given a ribbon graph (G,χ) and a vertex $s \in V(G)$, we will work with a function $\widehat{\alpha}_{(G,\chi,s)} : \mathrm{Div}_s^0(G) \times \mathcal{T}(G) \to \mathcal{T}(G)$ which belongs to a specific class of *monoid actions* (i.e., maps satisfying Properties 2 and 3 in Definition 2.8).

Definition 2.8. A function of the form $\widehat{\alpha}_{(G,\chi,s)}$ is called a *sink-parameterized sandpile torsor action (on ribbon graphs)* if for any $T \in \mathcal{T}(G)$, and $D, D' \in \mathrm{Div}_s^0(G)$, the following properties are satisfied:

1. $\widehat{\alpha}_{(G,\chi,s)}(D,T) = \widehat{\alpha}_{(G,\chi,s)}(D',T) \iff [D] = [D']$.
2. $\widehat{\alpha}_{(G,\chi,s)}(0,T) = T$.
3. $\widehat{\alpha}_{(G,\chi,s)}(D+D',T) = \widehat{\alpha}_{(G,\chi,s)}(D,\widehat{\alpha}_{(G,\chi,s)}(D',T))$.
4. For any $T' \in \mathcal{T}(G)$, there exists some $D'' \in \mathrm{Div}_s^0(G)$ such that $\widehat{\alpha}_{(G,\chi,s)}(D'',T) = T'$.

Definition 2.9. The function $\widehat{\alpha}$ is called a *sink-parameterized sandpile torsor algorithm (on ribbon graphs)* if for any ribbon graph (G,χ) and $s \in V(G)$, the following conditions hold:

1. $\widehat{\alpha}_{(G,\chi,s)}$ is a sink-parameterized sandpile torsor action.
2. For any ribbon graph isomorphism ϕ from (G,χ) to (G',χ'), and any $D \in \mathrm{Div}_s^0(G)$,

$$\phi(\widehat{\alpha}_{(G,\chi,s)}(D,T)) = \widehat{\alpha}_{(G',\chi',\phi(s))}(\phi_{\mathrm{D}}(D),\phi(T)).$$

Definition 2.10. For every *plane graph* (G,χ), let $\alpha_{(G,\chi)}$ be an action of $\mathrm{Pic}^0(G)$ on $\mathcal{T}(G)$. We say α is a *sandpile torsor algorithm (on plane graphs)* if there is a sink-parameterized sandpile torsor algorithm $\widehat{\alpha}$ such that for every *plane graph* (G,χ), $s \in V(G)$, $D \in \mathrm{Div}_s^0(G)$, and $T \in \mathcal{T}(G)$,

$$\alpha_{(G,\chi)}([D],T) = \widehat{\alpha}_{(G,\chi,s)}(D,T).$$

We call a specific action of the form $\alpha_{(G,\chi)}$ a *sandpile torsor action*.

Notice that a sandpile torsor algorithm α is uniquely defined by a sink-parameterized sandpile torsor algorithm $\widehat{\alpha}$ such that for every plane graph (G,χ), $s, s' \in V(G)$, $D \in \mathrm{Div}_s^0(G)$, $D' \in \mathrm{Div}_{s'}^0(G)$, and $T \in \mathcal{T}(G)$,

$$\widehat{\alpha}_{(G,\chi,s)}(D,T) = \widehat{\alpha}_{(G,\chi,s')}(D',T) \iff [D] = [D']. \tag{1}$$

Remark 2.11. It is important that we restrict to *plane graphs* in Definition 2.10 since we claim that it is impossible for (1) to be satisfied by all *ribbon graphs*.

To prove this claim, let (G,χ) be a nonplanar ribbon graph with two vertices x and y and three edges between them. Let ϕ be the ribbon graph automorphism that swaps x and y and acts as the identity on the edges. It is straightforward to see that the induced automorphism ϕ_D maps each divisor to its inverse.

Suppose that $\widehat{\alpha}$ is a sink-parameterized sandpile torsor algorithm. By Definition 2.9, this map must satisfy the equality $\widehat{\alpha}_{(G,\chi,x)}(D,T) = \widehat{\alpha}_{(G,\chi,y)}(-D,T)$ for all $D \in \mathrm{Div}_x^0(G)$. However, $[D] \neq [-D]$ unless $[D]$ is the identity. In particular, it is not possible for $\widehat{\alpha}$ to satisfy (1). This example is also discussed in [McD21a, Figure 1].

Two well-known sink-parameterized sandpile torsor algorithms are the *rotor-routing algorithm* and the *Bernardi algorithm*. These algorithms are distinct in general, and their constructions appear unrelated [BW17, Figure 9].[5] However, both algorithms satisfy (1) [CCG14, BW17], and they both define the same

[5] In fact, Baker and Wang conjectured that for every nonplanar ribbon graph, there is a sink vertex where the two algorithms are distinct [BW17]. This conjecture was independently proven by Ding as well as by Shokrieh and Wright [Din21, SW21].

sandpile torsor algorithm [BW17, Theorem 7.1]. As discussed in Section 1, this surprising equivalence inspired Conjecture 1.1, which asks if this sandpile torsor algorithm is in some sense unique. We will not explicitly work with the Bernardi algorithm in this paper, but we encourage the curious reader to see [BW17] for its construction, as well as [Yue17] and [KLT22] for alternate perspectives when restricting to plane graphs.

Remark 2.12. It is possible to define sandpile torsor algorithms directly instead of first defining sink-parameterized sandpile torsor algorithms. However, it is convenient to introduce sink-parameterized sandpile torsor algorithms because we make heavy use of the rotor-routing algorithm (defined in Section 2.3), which is traditionally defined on ribbon graphs that have a sink vertex. In fact, the proof of one of our main results, the consistency of the rotor-routing algorithm (Theorem 4.6), involves the application of the rotor-routing algorithm to nonplanar ribbon graphs.

2.3. Rotor-routing

Let (G, χ) be a ribbon graph and $s \in V(G)$.

Definition 2.13. A *rotor configuration with sink s* is an assignment of an incident edge to every vertex of G except for s. Given a rotor configuration ρ and $x \in V(G) \setminus s$, we write $\rho\langle x\rangle$ for the edge assigned to x.

We refer to edges of the form $\rho\langle x\rangle$ as *rotors*. It is useful to visualize a rotor $\rho\langle x\rangle$ as a directed edge incident to x and oriented away from x. We now define a function r_x which rotates the rotor at x one position.

Definition 2.14. Let ρ be a rotor configuration with sink s and $x \in V(G) \setminus s$. We write $r_x(\rho)$ for the rotor configuration defined by

$$r_x(\rho)\langle v\rangle = \begin{cases} \chi(v, \rho\langle v\rangle) & \text{if } v = x, \\ \rho\langle v\rangle & \text{otherwise.} \end{cases}$$

Definition 2.15. We say that a collection of rotors forms a *directed cycle* C if there is a path along them that returns to its original vertex. Given a directed cycle C, we write $E(C)$ for the set of edges corresponding to the rotors that make up C and $V(C)$ for the set of vertices incident to these edges. A collection of rotors is *acyclic* if it contains no cycles.

Definition 2.16. Let $s \in V(G)$ and $T \in \mathcal{T}(G)$. We write T_s for the rotor configuration with sink s such that for every $x \in V(G) \setminus s$, the rotor $T_s\langle x\rangle$ is the unique edge in T incident to x along the path from x to s.

In other words, T_s describes the orientation of the edges of T such that each edge is directed toward s. The following lemma is immediate.

Lemma 2.17. *Let ρ be a rotor configuration with sink s on a ribbon graph (G, χ). There exists some $T \in \mathcal{T}(G)$ such that $T_s = \rho$ if and only if ρ is acyclic. In particular, T is the set of edges which form the rotors of ρ.*

Given a sandpile element of the form $c - s \in \mathrm{Div}^0_s(G)$ for some $c, s \in V(G)$, and a spanning tree $T \in \mathcal{T}(G)$, Algorithm 1 outputs a rotor configuration with sink s. At each step of this algorithm, a single rotor rotates according to the ribbon structure χ (on a plane graph, this would be a counterclockwise rotation). When the algorithm terminates, we obtain a new rotor configuration. See Figure 2(a) for a demonstration of Algorithm .

There is also a variant of Algorithm 1 involving *unicycles*, which we will discuss in Appendix A.

Algorithm 1 was first explored by Priezzhev et al. under the name *Eulerian walkers model* [PDDK96]. By [HLM+08, Lemma 3.10], the final rotor ρ in Algorithm 1 is acyclic. By Lemma 2.17, this means there is a unique tree $T' \in \mathcal{T}(G)$ such that $T'_s = \rho$. For the inputs $c, s \in V(G)$ and $T \in \mathcal{T}(G)$, we define the mapping $\widehat{r}_{(G,\chi,s)}(c - s, T) := T'$, where T'_s is the output of Algorithm 1.

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

Algorithm 1 Single-chip rotor-routing

```
Input: A tree $T \in \mathcal{T}(G)$ and a divisor of the form $c - s \in \mathrm{Div}^0_s(G)$.
Output: The rotor configuration $\rho$ with sink vertex $s$.
Initialize the rotor configuration $\rho = T_s$.
Initialize a ‘traveling vertex’ $x$ at $c$. We will call $x$ a chip.
while $x \neq s$ do
    Replace $\rho$ with $r_x(\rho)$.                          ▷ rotate the rotor at $x$ by one position.
    Replace $x$ with the other vertex incident to $\rho\langle x\rangle$.   ▷ move $x$ across the rotor.
end while
return $\rho$.
```

Holroyd et al. showed how this model could be used to define a sink-parameterized sandpile torsor action on any ribbon graph (G, χ). Fix any $D = \sum_{v \in V(G)} n_v v \in \mathrm{Div}^0_s(G)$ and let $N := -n_s$. Note that $D = \sum_{i=1}^{N}(c_i - s)$ for some sequence of vertices $(c_1, \dots, c_N)$. Let $T^0 = T$, and for each $i \in [1, N]$, let $T^i = \widehat{r}_{(G,\chi,s)}(c_i - s, T^{i-1})$. Then, we define $\widehat{r}_{(G,\chi,s)}(D, T) := T^N$. By [HLM+08, Corollary 2.6], T^N does not depend on the ordering of $(c_1, \dots, c_N)$. Thus, $\widehat{r}_{(G,\chi,s)}$ is a well-defined monoid action of $\mathrm{Div}^0_s(G)$ on $\mathcal{T}(G)$.

Theorem 2.18. [LL09, Theorem 2.5]. *The rotor-routing algorithm $\widehat{r}$ constructed above is a sink-parameterized sandpile torsor algorithm.*

Chan, Church and Grochow proved that, on plane graphs, the rotor-routing action does not depend on the choice of sink vertex, which implies the following result.

Theorem 2.19. [CCG14, Theorem 2]. *The rotor-routing algorithm $\widehat{r}$ satisfies (1). In particular, it defines a sandpile torsor algorithm r.*

Theorem 2.19 tells us that for any plane graph (G, χ), there is a well-defined free transitive action $r_{(G,\chi)}$ of $\mathrm{Pic}^0(G)$ on $\mathcal{T}(G)$. To evaluate $r_{(G,\chi)}(S, T)$ for some $S \in \mathrm{Pic}^0(G)$ and $T \in \mathcal{T}(G)$, we first choose a sink vertex s and some $D \in \mathrm{Div}^0_s(G)$ such that $[D] = S$ (such a D must exist by Lemma 2.3). Then, we calculate $r_{(G,\chi)}(S, T) = \widehat{r}_{(G,\chi,s)}(D, T)$ through repeated applications of Algorithm 1.

Throughout this paper, we will use the term *rotor-routing algorithm* to refer to both $\widehat{r}$ and r. Context should resolve any potential ambiguity.

3. Useful properties of rotor-routing

In the previous section, we defined the rotor-routing sandpile torsor algorithm. In this section, we introduce several properties of rotor-routing that we will use in later sections. Some of our proofs for results in this section require additional terminology which we will delay until Appendix A. As in the previous section, G is a finite connected graph which has no loops but may have multiple edges.

Lemma 3.1. *For any $D \in \mathrm{Div}^0(G)$, we can write*

$$D = \sum_{i=1}^{k}(c_i - s_i),$$

where for every i, there exists an edge f_i such that $\mathfrak{i}(f_i) = (c_i, s_i)$.

Proof. This lemma follows immediately from the definition of $\mathrm{Div}^0(G)$ and the fact that G is connected. □

Corollary 3.2. *Suppose that α and β are sandpile torsor algorithms such that for all plane graphs (G, χ), for all $f \in E(G)$ and for all $T \in \mathcal{T}(G)$, we have,*

$$\alpha_{(G,\chi)}([c-s], T) = \beta_{(G,\chi)}([c-s], T),$$

where $\{c, s\} = \mathfrak{i}(f)$. Then, $\alpha = \beta$.

Corollary 3.2 will allow us to focus on the case where the chip and sink are adjacent.

Proposition 3.3. *If there is some $f \in E(G)$ such that $\mathfrak{i}(f) = \{c, s\}$, then no rotor can make more than a full turn and no edge may be crossed more than once when applying Algorithm 1 with input T and $c - s$.*

Proposition 3.3 follows from [HLM+08, Lemma 4.9], which we will prove Appendix A. Note that this result requires an edge between c and s in G.

Proposition 3.3 holds for any ribbon graph. Next, we will give two lemmas that require planarity. We will prove these lemmas in Appendix A using ideas from [CCG14].

A defining fact about plane graphs is that any directed cycle C partitions the vertices $V(G) \setminus V(C)$ and edges $E(G) \setminus E(C)$ into two classes. In particular, they are either to the 'left' of C or to the 'right' of C (where the 'left' edges and vertices are in the interior of a cycle oriented counterclockwise or the exterior of a cycle oriented clockwise). Note that the edge case where C is a single bi-directed edge will never be an issue.

Let (G, χ) be a plane graph, $f \in E(G)$, $\mathfrak{i}(f) = \{c, s\}$, and $T \in \mathcal{T}(G)$. Suppose that $f \notin T$, but T_s contains some path $x_1, x_2, \ldots, x_k$ of rotors from c to s. Let C be the directed cycle formed by $\{f, x_1, x_2, \ldots, x_k\}$.

Lemma 3.4. *In the construction above, if Algorithm 1 is applied with input T and $c - s$, then the chip never crosses any edges to the right of C.*

Let (G, χ) be a plane graph, $f \in E(G)$, $\mathfrak{i}(f) = \{c, s\}$ and $T \in \mathcal{T}(G)$. Apply Algorithm 1 with input T and $c - s$. Suppose that it takes n steps through the while loop for the algorithm to terminate. For $k \in [0, n]$, let c_k and ρ_k be the chip location and rotor configuration, respectively, after k passes through the while loop of Algorithm 1.

Lemma 3.5. *In the construction above, suppose that for some $i \in [0, n]$, ρ_i includes a directed cycle C. Then there exists some $j \in [0, n]$ such that ρ_j includes all of the rotors of C in reverse order.*

See Appendix A for proofs of Lemmas 3.4 and 3.5.

We conclude this section with a technical lemma which gives conditions where the cyclic order of edges around a vertex can be rearranged without affecting specific rotor-routing outputs. This lemma will be used repeatedly in the proof of Theorem 4.6.

Let $c, s \in V(G)$ and $f \in E(G)$ with $\mathfrak{i}(f) = \{c, s\}$. Let $T' = r_{(G,\chi)}([c-s], T)$. Consider any $x \in V(G)$. We can write $\chi(x)$ in the form $(T_s\langle x\rangle, e_1, e_2, \ldots, e_p, T'_s\langle x\rangle, \widehat{e}_1, \ldots, \widehat{e}_l)$. Let σ be a permutation on the set $[1, p]$ and let π be a permutation on the set $[1, l]$. Define $\widetilde{\chi}$ to be a ribbon structure such that $\widetilde{\chi}(x) = (T_s\langle x\rangle, e_{\sigma(1)}, e_{\sigma(2)}, \ldots, e_{\sigma(p)}, T'_s\langle x\rangle, \widehat{e}_{\pi(1)}, \ldots, \widehat{e}_{\pi(l)})$ and $\widetilde{\chi}(v) = \chi(v)$ for all other $v \in V(G)$.

Lemma 3.6. *In the construction above, we get the following equality:*

$$\widehat{r}_{(G,\widetilde{\chi},s)}(c-s, T) = \widehat{r}_{(G,\chi,s)}(c-s, T) = r_{(G,\chi)}([c-s], T).$$

Proof. We will prove the lemma by contradiction. The second equality follows from Definition 2.10. For the first equality, suppose that for some vertex $v \in V(G)$, the rotor at v moves past $T'_s\langle v\rangle$ on $(G, \widetilde{\chi})$ (otherwise the proof works analogously after switching the roles of (G, χ) and $(G, \widetilde{\chi})$). Consider the first instance where such an event occurs. At this moment, the chip must have reached v more times than it did during rotor-routing on (G, χ). Because no edge is crossed more than once in each direction (by Proposition 3.3), the chip must enter v along an edge that it did not cross during rotor-routing on

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

(G,χ). It follows that the rotor on some vertex y that is adjacent to v must have already passed $T'_s\langle y\rangle$. This contradicts our minimality assumption. □

It is worth noting that Lemma 3.6 holds even when $(G,\widetilde{\chi})$ is a nonplanar ribbon graph.

4. Rotor-routing is consistent

In this section, we introduce the notion of *consistency* of sandpile torsor algorithms and show that the rotor-routing algorithm is consistent.

Throughout this section, (G,χ) is a plane graph, $f \in E(G)$ and $\mathfrak{i}(f)=\{c,s\}$. For $e \in E(G)$, we write $G\setminus e$ for the graph we obtain from G by *deleting* edge e and G/e for the graph we obtain by *contracting* the edge e. After contracting e, we also remove all loop edges because these cannot occur on any spanning tree and have no effect on the rotor-routing algorithm.

Definition 4.1. Let (G,χ) be a ribbon graph with $e \in E(G)$, where $\mathfrak{i}(e)=\{x,y\}$.

- Define $\chi\setminus e$ to be equal to χ, except with e removed from $\chi(x)$ and $\chi(y)$.
- Suppose that after removing edges parallel to e, $\chi(x) = (e,e_1,e_2,\ldots,e_p)$ and $\chi(y) = (e,\widehat{e}_1,\widehat{e}_2,\ldots,\widehat{e}_l)$. Define χ/e to be equal to χ except that on the contracted edge z, we have

$$(\chi/e)(z) := (e_1,e_2,\ldots,e_p,\widehat{e}_1,\widehat{e}_2,\ldots,\widehat{e}_l).$$

The following lemma is a simple exercise.

Lemma 4.2. *If (G,χ) is a plane graph, then $(G\setminus e,\chi\setminus e)$ and $(G/e,\chi/e)$ are also plane graphs.*

We will refer to any plane graph (G',χ') that can be obtained by contracting and deleting edges of (G,χ) as a *minor* of the plane graph (G,χ). Note that whenever (G',χ') is a minor of (G,χ), there is a natural surjective map from $V(G)$ to $V(G')$ and from $E(G)$ to $E(G')$. When there is no ambiguity, we will use the same labeling for the vertices and edges of G as we do for their image under this map.

Definition 4.3. A sandpile torsor algorithm α is *consistent* if for every plane graph (G,χ), every choice of $f \in E(G)$ and every choice of $T \in \mathcal{T}(G)$, the following three properties hold (where we define $\{c,s\}=\mathfrak{i}(f)$):

1. For any $e \in E(G)$ such that $\mathfrak{i}(e)\neq\{c,s\}$, if $e \in T\cap\alpha_{(G,\chi)}([c-s],T)$, then

$$\alpha_{(G,\chi)}([c-s],T)\setminus e = \alpha_{(G/e,\chi/e)}([c-s],T\setminus e).$$

2. For any $e \in E(G)$, if $e \notin T\cup\alpha_{(G,\chi)}([c-s],T)$, then

$$\alpha_{(G,\chi)}([c-s],T) = \alpha_{(G\setminus e,\chi\setminus e)}([c-s],T).$$

3. For any $e \in E(G)\setminus f$, if there is a cut vertex x such that all paths from e to f pass through x, then

$$e\in T \iff e \in \alpha_{(G,\chi)}([c-s],T).$$

The first two conditions of Definition 4.3 are illustrated by Figure 2. The third condition says that we can treat two subgraphs joined at a vertex independently.

Proposition 4.4. *Condition 3 of Definition 4.3 is satisfied by the rotor-routing algorithm.*

Proof. Let x be a cut vertex. We say x *separates* two edges or vertices g and h if all paths between g and h must contain x. We claim that there is some $D \in \mathrm{Div}^0_x(G)$ such that $[D]=[c-s]$ and $D=\sum_{v\in V}(v-x)$, where V is a multiset of vertices which x does not separate from f. Assume that the claim holds. Then, by Definition 2.10, it follows that

$$r_{(G,\chi)}([c-s],T)=\widehat{r}_{(G,\chi,x)}(D,T).$$

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

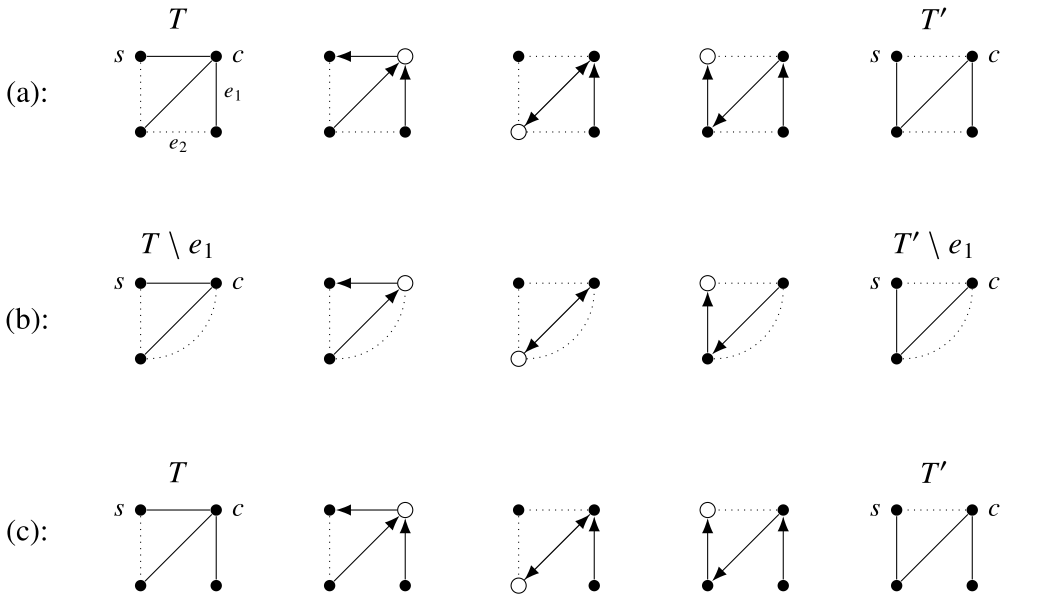

Figure 2. *An illustration of the implementation and consistency of the rotor-routing algorithm on the graph (G, χ), where χ denotes counterclockwise rotation. We denote the chip by a hollow vertex. Figure 2(a) demonstrates how Algorithm 1 can be used to compute $T' := r_{(G,\chi)}([c-s],T)$. In Figure 2(b), the rotor-routing algorithm commutes with contraction: $T'/e_1 = r_{(G/e_1,\chi/e_1)}([c-s],T \setminus e_1)$ (see Definition 4.3 (1)). In Figure 2(c), the rotor-routing algorithm commutes with deletion: $T' = r_{(G\setminus e_2,\chi\setminus e_2)}([c-s],T)$ (see Definition 4.3 (2)). See Clips 8–10 from https://www.youtube.com/watch?v=tSdVSk5o4Kg for animated examples of consistency on a larger plane graph (note that the video uses a clockwise convention for rotor-routing).*

When rotor-routing with sink x, all chips halt upon reaching x. As a result, any rotor that x separates from f will not rotate during the implementation of Algorithm 1. It follows that for any edge e that is separated from f, we have

$$e \in T \iff e \in \widehat{r}_{(G,\chi,x)}(D,T) \iff e \in r_{(G,\chi)}([c-s],T).$$

To prove the claim, we use a modified version of the proof of Lemma 2.3. If $s = x$, we can take $D = c - s$. Otherwise, let $V_1 \subseteq V(G \setminus x)$ be the set of vertices in the same component as s in $G \setminus x$, and let $V_2 = V(G \setminus x) \setminus V_1$. Let $\delta_x \in \mathrm{Div}^0_x(G)$ be the divisor which places $\deg(v)$ chips on every $v \in V_1$, 0 chips on every $w \in V_2$, and $-\sum_{v\in V_1}\deg(v)$ chips on x. Define δ°_x to be the divisor we obtain after repeatedly firing any vertex v with at least $\deg(v)$ many chips. Notice that $\delta_x - \delta^\circ_x$ has a positive value on each $v \in V_1$ and is equal to 0 on each $w \in V_2$. Furthermore, $[\delta_x - \delta^\circ_x] = 0$. We prove the claim by letting $D = c - s + \delta_x - \delta^\circ_x$. □

By requiring consistent sandpile torsor algorithms to satisfy Condition 3, we simplify many of the arguments in Section 5. However, we are not convinced that this condition is necessary.

Question 4.5. Is Condition 3 of Definition 4.3 implied by Conditions 1 and 2?

Theorem 4.6. *The rotor-routing sandpile torsor algorithm is consistent.*

Proof. We begin with a paragraph giving the outline of our argument. We reduce the result to six specific cases. If an edge e is in both T and $T' := r_{(G,\chi)}([c-s],T)$, then the rotor at e may be untouched during rotor-routing, it may complete a full rotation or it could be oriented in opposite directions in

T_s and T'_s (note that by Proposition 3.3, the rotor will never move more than a full rotation). When the edge e is in neither tree, the traveling vertex may never have crossed e, it may have crossed e in both directions or it may have only crossed e in one direction. For some of these cases, the path of the traveling vertex will change after deleting or contracting e, but we use Lemma 3.6 to show that it still crosses the same edges, just in a different order. The planarity condition comes into play for the final case (where the edge is crossed in one direction). We apply Lemma 3.5 to show that this case is never realized.

Throughout this proof, we let $\mathfrak{i}(e) = \{x, y\}$ and $T' = r_{(G,\chi)}([c-s], T)$. Let $\mathcal{R}$ be the ordered list of directed edges crossed when rotor-routing on (G, χ) with input $c - s$ and T.

We will consider three cases to prove that the first condition of Definition 4.3 holds, and then three cases to prove that the second condition holds. The theorem then follows from Proposition 4.4. For the first three cases, assume that $e \in T \cap T'$. We further assume without loss of generality that $T_s\langle x\rangle = e$. Let $T^e = r_{(G/e,\chi/e)}([c-s], T \setminus e)$ and let $\mathcal{R}_e$ be the ordered list of directed edges crossed when rotor-routing on $(G/e, \chi/e)$ with input $c - s$ and $T \setminus e$. Finally, let z be the vertex formed by contracting e. Notice that $(T \setminus e)_s\langle z\rangle = T_s\langle y\rangle$ if $y \neq s$ and z does not have a rotor in $(T \setminus e)$ when $y = s$.

Case 1: The chip never enters the vertex x while rotor-routing.

For this case, we claim that $\mathcal{R}$ and $\mathcal{R}_e$ are identical.

Assume for the sake of induction that this statement holds for the first k edges. After crossing these rotors, the chip must be on the same vertex of both graphs (or the chip is on y in G and z in G/e). By assumption, this vertex has been entered the same number of times for each graph, so the rotor here must be in the same position. It follows that the chip will also exit along the same rotor for both graphs.

This proves our claim that $\mathcal{R} = \mathcal{R}_e$. For every $v \in V(G) \setminus \{s, x, y\}$, it follows immediately from the claim that $T'_s\langle v\rangle = T^e_s\langle v\rangle$. Furthermore, by construction, we also have $T'_s\langle y\rangle = T^e_s\langle z\rangle$ and $T'_s\langle x\rangle = e$. This implies that $T^e = T' \setminus e$ as desired.

Case 2: The rotor at x spins around completely. In other words, $T_s\langle x\rangle = T'_s\langle x\rangle$, but $T_s\langle x\rangle \neq \rho\langle x\rangle$ for some intermediate rotor configuration ρ.

We begin with a brief, high-level explanation of the proof of this case. We start by showing that $x \neq c$ and $y \neq s$ to avoid any edge cases that may arise. We then carefully define an alternative (possibly nonplanar) ribbon structure $\widetilde{\chi/e}$ for the contracted graph G/e for which the steps of the rotor-routing algorithm used to compute $\widehat{r}_{(G/e,\widetilde{\chi/e},s)}(c-s, T \setminus e)$ and $\widehat{r}_{(G,\chi,s)}(c-s, T)$ are nearly identical. It is then easily argued that $\widehat{r}_{(G/e,\widetilde{\chi/e},s)}(c-s, T \setminus e) = T' \setminus e$. We then apply Lemma 3.6 to show that

$$T^e = \widehat{r}_{(G/e,\chi/e,s)}(c-s, T \setminus e) = \widehat{r}_{(G/e,\widetilde{\chi/e},s)}(c-s, T \setminus e) = T' \setminus e.$$

Now, we give a complete proof. First, we note that by Proposition 3.3, it is impossible for x to complete more than one full rotation. Let $\chi(x) = (e, e_1, \ldots, e_p)$ and $\chi(y) = (e, \widehat{e}_1, \ldots, \widehat{e}_l)$ after removing edges parallel to e, so that $(\chi/e)(z) = (e_1, \ldots, e_p, \widehat{e}_1, \ldots, \widehat{e}_l)$. Recall that we require $e \neq f$ for the first condition of Definition 4.3. Suppose that $x = c$. By assumption, the rotor must cross f from c to s at some point during rotor-routing. As soon as this happens, the algorithm terminates, and thus $T'_s\langle x\rangle = f$. This contradicts the assumption that $T'_s\langle x\rangle = e$, so we must have $x \neq c$.

Likewise, $y \neq s$. Otherwise, since $x \neq c$, for the rotor at x to make a full rotation, the chip must enter x precisely $\deg(x)$ many times. By Proposition 3.3, it must enter x along every edge, including the edge e from y to x, before the rotor makes a full rotation. However, this is clearly a contradiction as the rotor-routing algorithm halts as soon as the chip reaches $y = s$.

Let $T_s\langle y\rangle = \widehat{e}_i$ and $T'_s\langle y\rangle = \widehat{e}_j$. By Proposition 3.3, each edge may only be crossed once in each direction. By assumption, the chip exits x along each incident edge. Because $x \neq c$, this means that x must be entered its degree many times. Thus, the chip must traverse every edge incident to x in both

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

directions. In particular, the rotor at y must cross e, which implies that $i \geq j$. When rotor-routing on (G, χ), keep an ordered list of every edge other than e that the chip crosses when moving out of either x or y and call this list $\mathcal{E}$. This list must consist precisely of the edges

$$\{e_1, \ldots, e_p\} \cup \{\widehat{e}_1, \ldots, \widehat{e}_j\} \cup \{\widehat{e}_{i+1}, \ldots, \widehat{e}_l\}.$$

Furthermore, the last edge in $\mathcal{E}$ must be $\widehat{e}_j$. This is because after exiting x for the final time, the rotor enters y across e, and $T'_s\langle y\rangle = \widehat{e}_j$.

We now define an alternate ribbon structure on G/e, which we will call $\widetilde{\chi/e}$. For each vertex $v \neq z$, let $(\widetilde{\chi/e})(v) = (\chi/e)(v)$. Let $(\widetilde{\chi/e})(z)$ begin with $\mathcal{E}$ and then be ordered arbitrarily. Let $\mathcal{R} - \overline{e}$ be the sublist of $\mathcal{R}$ that is obtained after removing any directed edges between x and y. Finally, let $\widetilde{\mathcal{R}_e}$ be the ordered list of directed edges crossed when rotor-routing on $(G/e, \widetilde{\chi/e})$ with input $c - s$ and $T \setminus e$. We claim that $\mathcal{R} - \overline{e}$ and $\widetilde{\mathcal{R}_e}$ are identical.

Suppose that the claim holds and let $\widetilde{T} = \widehat{r}_{(G/e, \widetilde{\chi/e}, s)}(c - s, T \setminus e)$. For all $v \in V(G) \setminus \{x, y, s\}$, we know that $T'_s\langle v\rangle = \widetilde{T}_s\langle v\rangle$. Furthermore, by construction, we also have $T'_s\langle y\rangle = \widetilde{T}_s\langle z\rangle$ and $T'_s\langle x\rangle = e$. The fact that $T'_s\langle y\rangle = \widetilde{T}_s\langle z\rangle$, along with our construction of $\widetilde{\chi/e}$, implies that $\widetilde{\chi/e}$ satisfies the requirements of Lemma 3.6 (when comparing to χ/e on G/e). Thus, we know that $\widetilde{T} = T^e$. The result follows.

Now we just need to prove the claim that $\mathcal{R} - \overline{e} = \widetilde{\mathcal{R}_e}$. Assume for the sake of induction that the first k rotors match. After crossing these rotors, the chip must be on the same vertex of both graphs (or the chip is on z for G/e and on either x or y for G). If the chip is on some $v \neq z$, then the argument is analogous to the argument from Case 1. Thus, it suffices to consider the case where the chip is on z.

Out of the first k entries of $\mathcal{R} - \overline{e}$, let a be the number of edges that exit x or y. Then, by assumption, for the first k entries of $\widetilde{\mathcal{R}_e}$, there must be a edges that exit z. By construction, the next edge crossed in either case must be the $(a + 1)$th edge of $\mathcal{E}$. It follows by induction that $\mathcal{R} - \overline{e} = \widetilde{\mathcal{R}_e}$.

Case 3: The rotor e is directed differently for T_s and T'_s. In other words, $T'_s\langle y\rangle = e$.

This case is similar to the previous case, and we will use the same notation for $\chi(x)$ and $\chi(y)$. Suppose that $T_s\langle y\rangle = \widehat{e}_i$ and $T'_s\langle x\rangle = e_j$. As before, let $\mathcal{E}$ be the ordered list of edges other than e that the chip crosses when moving out of x or y. This list must consist precisely of the edges

$$\{\widehat{e}_{i+1}, \ldots, \widehat{e}_l\} \cup \{e_1, \ldots, e_j\}.$$

Furthermore, after exiting y for the final time across e, the chip enters x. This means that the last edge in $\mathcal{E}$ must be e_j. Similarly to the previous case, we define a ribbon structure $\widetilde{\chi/e}$ on G/e. For $v \neq z$, let $(\widetilde{\chi/e})(v) = (\chi/e)(v)$, and let $(\widetilde{\chi/e})(z)$ begin with $\mathcal{E}$ and then be ordered arbitrarily.

The rest of the argument is analogous to the argument used for Case 2. In particular, we define $\mathcal{R} - \overline{e}$ and $\widetilde{\mathcal{R}_e}$ and then show that they are identical. We also use this idea to show that $\widetilde{\chi/e}$ satisfied the requirements of Lemma 3.6. The only slight difference from the previous case is that the final positions of the rotors at x and y swap.

For the last three cases, $e \notin T \cup T'$. For these cases, we define ${}^eT = r_{(G\setminus e, \chi\setminus e)}([c - s], T)$. We need to show that ${}^eT = T'$. Also, let ${}_e\mathcal{R}$ be the ordered list of directed edges crossed when rotor-routing on $(G \setminus e, \chi \setminus e)$ with input $c - s$ and T.

Case 4: The chip never crosses e.

Since the chip never crosses e, removing this edge has no effect on the output of the rotor-routing algorithm. We can use an analogous argument to what we used for Case 1. In particular, we show that $\mathcal{R} = {}_e\mathcal{R}$, and the result follows.

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

Case 5: The edge e is crossed in both directions, but $T'_s\langle x\rangle \neq e \neq T'_s\langle y\rangle$.

First, we observe that $\mathfrak{i}(e) \neq \{c, s\}$; otherwise, it is impossible for the chip to cross e in both directions. Thus, c and s must remain incident on $G \setminus e$. This allows us to apply Proposition 3.3 to say that no edge is crossed more than once in each direction.

Define $\chi(x)$ and $\chi(y)$ as in Case 2. Without loss of generality, suppose that the first time the chip crosses e, it moves from y to x. Let ρ be the first rotor configuration reached while rotor-routing on (G, χ) (with input $c - s$ and T) such that $\rho\langle y\rangle = e$. Further, suppose that $\rho\langle x\rangle = e_q$. We introduce a new ribbon structure $\widetilde{\chi}$ on G. For $v \in V(G) \setminus x$, let $\widetilde{\chi}(v) = \chi(v)$, and let

$$\widetilde{\chi}(x) = (e, e_{q+1}, e_{q+2}, \dots, e_p, e_1, e_2, \dots, e_q).$$

Let E_1 be the set of edges after $T_s\langle x\rangle$ and before $T'_s\langle x\rangle$ with respect to χ and E_2 be the set of edges after $T_s\langle x\rangle$ and before $T'_s\langle x\rangle$ with respect to $\widetilde{\chi}$. Because we only rearranged the position of e, we must have $\mathrm{E}_1 \setminus e = \mathrm{E}_2 \setminus e$. Furthermore, $e \in \mathrm{E}_1$ by assumption, and $e \in \mathrm{E}_2$ by construction (because the rotor-routing algorithm is equivalent on (G, χ) and $(G, \widetilde{\chi})$ until the chip crosses e from x to y). It follows that $\mathrm{E}_1 = \mathrm{E}_2$.

By the reasoning in the previous paragraph, $\widetilde{\chi}$ satisfies the conditions for Lemma 3.6, which means that $\widehat{r}_{(G,\widetilde{\chi},s)}(c - s, T) = T'$.

Let $\widetilde{\mathcal{R}}$ be the ordered list of directed edges crossed when rotor-routing on $(G, \widetilde{\chi})$ with input $c - s$ and T. Let $\widetilde{\mathcal{R}} - e$ be the remaining edges of $\widetilde{\mathcal{R}}$ after removing e (but not edges parallel to e). We claim that $\widetilde{\mathcal{R}} - e = {}_e\mathcal{R}$. This follows from construction, because when the chip crosses e from y to x in $\widetilde{\mathcal{R}}$, we know by construction that e_q is the rotor at x. Therefore, the next step of the algorithm is to cross back along e in the other direction. Thus, it is as though this edge did not exist.

Given our claim, the result follows from the same reasoning we used for the previous cases.

Case 6: The edge e is only crossed in one direction.

We will use Lemma 3.5 to show that this case is never realized. This is the one case where the planarity condition is vital.

Without loss of generality, suppose that the chip crosses e from x to y. Then, by assumption, it must reach x again without crossing e. Consider the next occurrence when the chip returns to x. Let the rotor configuration at this moment be ρ.

Consider the sequence $(x, y, v_1, v_2, \dots)$ of vertices where $\rho\langle y\rangle = v_1$ and each v_i satisfies $\mathfrak{i}(\rho\langle v_i\rangle) = \{v_i, v_{i+1}\}$. Since there are a finite number of vertices, and the chip has just returned to x, there must be some j such that $x \in \mathfrak{i}(\rho\langle v_j\rangle)$. Thus, the rotors incident to these vertices form a directed cycle, which does not include the sink vertex. By Lemma 3.5, this cycle must reverse before the rotor-routing algorithm terminates. In order for the cycle to reverse, the chip must cross e in the other direction. This gives a contradiction. □

We end this section with an example developed by Tóthmérész, which demonstrates one of the subtleties of Definition 4.3. In particular, Theorem 4.6 fails if we remove the requirement that c and s are adjacent.

Example 4.7. [Tót23, Remark 20] Figure 3 gives a pair of vertices c and s, and edge e, and a spanning tree T, such that $e \in T \cap r_{(G,\chi)}([c - s], T)$, but $r_{(G,\chi)}([c - s], T) \neq r_{(G/e,\chi/e)}([c - s], T \setminus e)$. The reason that this example does not contradict Theorem 4.6 is because c and s are not connected by an edge. In particular, Proposition 3.3 is no longer satisfied.

Tóthmérész's note (which was written after the first version of our paper) also includes an alternative proof of Theorem 4.6 (see [Tót23, Proposition 17]).

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

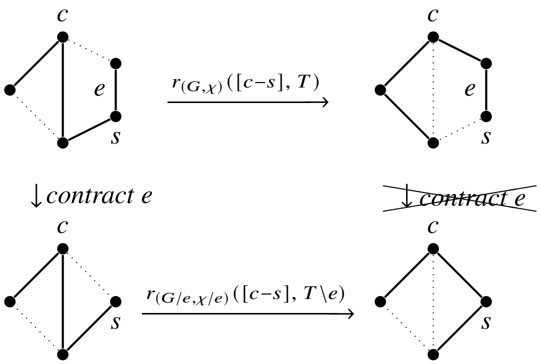

Figure 3. *This figure shows the importance of the source and sink vertices being adjacent in the definition of consistency. As usual, the ribbon graph is oriented counterclockwise.*

5. Uniqueness of consistent torsor algorithms

In this section, we classify the consistent sandpile torsor algorithms. In particular, we show that every consistent sandpile torsor algorithm must either be equivalent to rotor-routing, or one of three related algorithms which we say have the same *structure* as rotor-routing.

For any ribbon graph (G,χ), let $\overline{\chi}$ be the reverse cyclic order around each vertex. Notice that if (G,χ) is a plane graph, then $(G,\overline{\chi})$ is a plane graph as well (simply reflect the planar embedding of (G,χ) to get a planar embedding of $(G,\overline{\chi})$).

Definition 5.1. Suppose α is a sandpile torsor algorithm. Define $\overline{\alpha}$, α^{-1} and $\overline{\alpha}^{-1}$ such that for any plane graph (G,χ), $S\in \operatorname{Pic}^0(G)$ and $T\in\mathcal{T}(G)$, we have

$$\alpha_{(G,\chi)}(S,T)=\overline{\alpha}_{(G,\overline{\chi})}(S,T)=\alpha^{-1}_{(G,\chi)}(-S,T)=\overline{\alpha}^{-1}_{(G,\overline{\chi})}(-S,T).$$

If α is the rotor-routing algorithm, then $\overline{\alpha}$ reverses the direction in which the rotors turn, α^{-1} switches the role of the chip and sink, and $\overline{\alpha}^{-1}$ makes both of these changes.

Proposition 5.2. *If α is a (consistent) sandpile torsor algorithm, then $\overline{\alpha}$, α^{-1} and $\overline{\alpha}^{-1}$ are distinct (consistent) sandpile torsor algorithms.*

Proof. It is straightforward to show that the defining properties of sandpile torsor actions, as well as consistency, are preserved if we replace α with any of the other three possibilities.

Let E_3 be the triple edge with a planar ribbon structure. Reversing the ribbon structure on E_3 is a ribbon graph automorphism that is the identity on $V(E_3)$ and $\operatorname{Pic}^0(E_3)$ but is not the identity on $E(E_3)$ or $\mathcal{T}(E_3)$. It follows that $\alpha_{(E_3)}\neq\overline{\alpha}_{(E_3)}$ and $\alpha^{-1}_{(E_3)}\neq\overline{\alpha}^{-1}_{(E_3)}$.

Let C_3 be the circuit with three edges and its unique ribbon structure. Then, $\operatorname{Pic}^0(C_3)\cong\mathbb{Z}/3\mathbb{Z}$, which means that there are elements of $\operatorname{Pic}^0(G)$ that are not equal to their own inverse. It follows that $\alpha_{(C_3)}\neq\alpha^{-1}_{(C_3)}$ and $\overline{\alpha}_{(C_3)}\neq\overline{\alpha}^{-1}_{(C_3)}$.

By combining the arguments from the previous two paragraphs, we know that for any two of these algorithms, there exist plane graphs for which the sandpile torsor actions are distinct. □

Definition 5.3. Sandpile torsor algorithms α and β have the same *sandpile torsor structure* if $\beta\in\{\alpha,\overline{\alpha},\alpha^{-1},\overline{\alpha}^{-1}\}$.

Our goal will be to prove the following version of Conjecture 1.1.

Theorem 5.4. *Every consistent sandpile torsor algorithm on plane graphs has the same sandpile torsor structure as the rotor-routing algorithm.*

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

We provide the proof of Theorem 5.4 at the end of the section. The first step of the proof is to note that we can restrict our attention to *2-connected* plane graphs (i.e., plane graphs with no cut vertices). In particular, we show the following.

Lemma 5.5. *If α and β are consistent sandpile torsor algorithms such that $\alpha_{(G,\chi)} = \beta_{(G,\chi)}$ for all 2-connected plane graphs (G,χ), then $\alpha = \beta$.*

Proof. Suppose for the sake of contradiction that there is some plane graph (G,χ) such that $\alpha_{(G,\chi)} \neq \beta_{(G,\chi)}$. By the contrapositive of Corollary 3.2, this means that for some adjacent vertices c and s, and some $T \in \mathcal{T}(G)$, we have

$$\alpha_{(G,\chi)}([c-s],T) \neq \beta_{(G,\chi)}([c-s],T).$$

We can assume (G,χ) is not 2-connected, or else the contradiction is immediate. Let f be an edge such that $\mathfrak{i}(f) = \{c,s\}$. By Condition 3 of consistency (Definition 4.3), any edges separated from f after removing cut vertices must be unchanged when acting on T by $[c-s]$. For all such edges, we contract the edges that are in T and delete the edges that are not. We are left with a 2-connected plane graph (G',χ') and a spanning tree $T' \in \mathcal{T}(G')$. By Conditions 1 and 2 of consistency, the inequality is preserved. In particular,

$$\alpha_{(G',\chi')}([c-s],T') \neq \beta_{(G',\chi')}([c-s],T').$$

This is a contradiction. □

In general, the rotor-routing algorithm can have a complicated global effect on a spanning tree. Our next goal toward a proof of Theorem 5.4 will involve defining a distinguished set of (sandpile group element, spanning tree) pairs called *source-turn pairs* (as well as a slightly less restrictive set of *single-step pairs*). These pairs are useful because they have the following two properties:

- The rotor-routing action on source-turn pairs and single-step pairs has a simple geometric characterization.
- To show that sandpile torsor algorithm is equivalent to the rotor-routing algorithm, it suffices to show that the two sandpile torsor algorithms agree on source-turn pairs (see Corollary 5.18).

We first define source-turn pairs and single-step pairs. Given a graph G and a spanning tree $T \in \mathcal{T}(G)$, a *leaf vertex* is a vertex incident to only one edge of T, whereas a *leaf edge* is an edge of T incident to a leaf vertex.

Definition 5.6. Let (G,χ) be a 2-connected ribbon graph, $T \in \mathcal{T}(G)$ and $c,s \in V(G)$.

- The pair $(c-s,T)$ is called a *single-step pair* at c if the rotor-routing algorithm terminates after a single rotor turns one position. We call the map from T to $r_{(G,\chi)}([c-s],T)$ a *single-step move* at c.
- The pair $(c-s,T)$ is called a *source-turn pair* at c if c is a leaf vertex of T and the equality $\{c,s\} = \mathfrak{i}(\chi(c,T_s\langle c\rangle))$ is satisfied. We call the map from T to $r_{(G,\chi)}([c-s],T)$ a *source-turn move* at c.

Lemma 5.7. *If $(c-s,T)$ is a source-turn pair, then $(c-s,T)$ is also a single-step pair.*

Proof. If c is a leaf vertex, then the unique edge of T incident to c must be equal to $T_s\langle c\rangle$. The result follows from the definition of rotor-routing. □

Recall from Definition 2.14 that for $x \in V(G)\setminus s$, the function r_x rotates the rotor at x one position. It is immediate from definition that when $(c-s,T)$ is a single-step pair, we have $r_{(G,\chi)}([c-s],T_s) = r_c(T_s)$. Furthermore, if $T_s\langle c\rangle = g$ and $r_c(T_s)\langle c\rangle = f$, then $r_{(G,\chi)}([c-s],T) = T\setminus g \cup f$. We say that $(c-s,T)$ is a *single-step pair from g to f* (where a *source-turn pair from g to f* is defined analogously). See Figure 4 for some examples or single-step pairs and source-turn pairs.

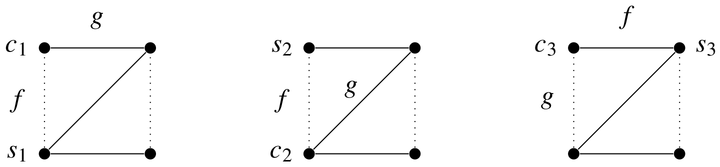

Figure 4. *Let T be the spanning tree depicted above (where the graph is given a counterclockwise ribbon structure). The pair $(c_1 - s_1, T)$ is a source-turn pair and a single-step pair from g to f. The pair $(c_2 - s_2, T)$ is a single-step pair from g to f, but not a source-turn pair. The pair $(c_3 - s_3, T)$ is neither a source-turn pair nor a single-step pair. However, $(s_3 - c_3, T)$ is a reverse single-step pair from f to g (see Definition 5.19).*

We now introduce the key result which we use to prove Corollary 5.18:

Theorem 5.8. *Let (G, χ) be a 2-connected ribbon graph. For any $T^{start}, T^{goal} \in \mathcal{T}(G)$, there exists a sequence of source-turn moves whose composition maps T^{start} to T^{goal}.*

Before proving the theorem, we give a high level outline of our argument.

Step 1: Fix an arbitrary *root vertex* $\mathfrak{r} \in V(G)$ and transform each $T \in \mathcal{T}(G)$ into rotor configurations $T_{\mathfrak{r}}$.
Step 2: Observe that source-turn moves on trees correspond to *source-rotations* in the corresponding rotor configurations (see Definition 5.12).
Step 3: Introduce a partial ordering on $\mathcal{T}(G)$ such that T^{goal} is the unique minimal spanning tree.
Step 4: Use an inductive argument to show that for any $T \neq T^{goal}$, it is possible to use source-rotations to transform $T_{\mathfrak{r}}$ into some $T'_{\mathfrak{r}}$ such that T' is less than T in the partial order on $\mathcal{T}(G)$.
Step 5: Because T^{goal} is the unique minimal spanning tree, the theorem follows.

In order to make this argument precise, it is convenient to introduce a few definitions and straightforward results. In particular, the inductive argument in Step 4 requires an understanding of the conditions under which source-turn moves applied to minors of (G, χ) correspond to source-turn moves in (G, χ). We establish these conditions in Lemmas 5.14 and 5.15, so it is convenient to delay the proof of Theorem 5.8 until immediately after the proof of Lemma 5.15.

Definition 5.9. Given a graph G, a spanning tree $T \in \mathcal{T}(G)$ and a vertex $\mathfrak{r} \in V(G)$, we write $x \prec_{T_{\mathfrak{r}}} y$ if the path from x to $\mathfrak{r}$ along T passes through y.

It is straightforward to check that $\prec_{T_{\mathfrak{r}}}$ always induces a partial ordering on $V(G) \setminus \mathfrak{r}$ and that a vertex is minimal if and only if it is a leaf vertex of T (other than $\mathfrak{r}$).

Lemma 5.10. *Let (G, χ) be a 2-connected ribbon graph and $T \in \mathcal{T}(G)$. For $e \in T$ with $\mathfrak{i}(e) = \{x, y\}$, $x \prec_{T_{\mathfrak{r}}} y$ if and only if $T_{\mathfrak{r}}\langle x \rangle = e$.*

Proof. This lemma follows immediately from Definitions 2.16 and 5.9. □

Lemma 5.11. *Let (G, χ) be a 2-connected ribbon graph, $T \in \mathcal{T}(G)$, $g \in T$, $f \in E(G) \backslash T$, $\mathfrak{i}(f) = \{c, s\}$, $\mathfrak{i}(g) = \{c, x\}$ and $\chi(c, g) = f$. The pair $(c - s, T)$ is a single-step pair from g to f if and only if $c \prec_{T_s} x$.*

Proof. For the if direction, Lemma 5.10 says that $T_s\langle x \rangle = g$. Thus, Algorithm 1 terminates after a single step and the output tree is $T \setminus g \cup f$. For the only if direction, $c \not\prec_{T_s} x$ implies that $T_s\langle x \rangle \neq g$. If the rotor-routing algorithm terminates after a single step, the resulting tree must include the edge g. Thus, it is not a single-step pair from g to f. □

Lemma 5.11 suggests that we can characterize single-step moves and source-turn moves in terms of rotors constructed from the underlying spanning tree T. We now make this characterization explicit in Definition 5.12 and show the correspondence between the two representations of single-step (or source-turn) moves in Lemma 5.13.

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

Definition 5.12. Let (G, χ) be a 2-connected ribbon graph with $T \in \mathcal{T}(G)$ and $c, \mathfrak{r} \in V(G)$.

- If $r_c(T_\mathfrak{r})$ is acyclic, then we say that $T_\mathfrak{r}$ is *rotatable at* c. We call the map from $T_\mathfrak{r}$ to $r_c(T_\mathfrak{r})$ a *rotation at* c.
- For $g, f \in E(G)$, we specify that $T_\mathfrak{r}$ is *rotatable at* c *from* g *to* f if $T_\mathfrak{r}$ is rotatable at c, $g = T_\mathfrak{r}\langle c\rangle$, and $f = \chi(c, g)$.
- The terms *source-rotatable at* c *(from* g *to* f*)* and *source-rotation* are defined analogously with the added requirement that c is a leaf vertex.

Lemma 5.13. *Let $T \in \mathcal{T}(G)$ and $\mathfrak{r} \in V(G)$. The configuration $T_\mathfrak{r}$ is rotatable (resp. source-rotatable) at c from g to f if and only if there exists a single-step pair (resp. source-turn pair) $(c - s, T)$ from g to f and $T_\mathfrak{r}\langle c\rangle = g$.*

Proof. Suppose that $T_\mathfrak{r}$ is rotatable at c from g to f. By Definition 5.12, this implies that $g = T_\mathfrak{r}\langle c\rangle$, $f = \chi(c, g)$ and $r_c(T_\mathfrak{r})$ is acyclic. Let $s = \mathfrak{i}(f) \setminus c$ and $x = \mathfrak{i}(g) \setminus c$. Let P be the unique path from c to s in T. If $g \notin P$, then $T \setminus g \cup f$ must contain a cycle, which contradicts the condition that $r_c(T_\mathfrak{r})$ is acyclic. Thus, $g \in P$ and $c \prec_{T_s} x$. By Lemma 5.11, $(c - s, T)$ is a single-step pair from g to f.

Alternatively, suppose that there exists an $s \in V(G)$ such that $(c - s, T)$ is a single-step pair from g to f and $T_\mathfrak{r}\langle c\rangle = g$. A single step through the rotor-routing algorithm produces the rotor configuration $r_c(T_s)$. By Lemma 5.11, we know that $c \prec_{T_s} x$, which implies $T_s\langle c\rangle = g$ by Lemma 5.10. Because $T_\mathfrak{r}\langle c\rangle = T_s\langle c\rangle$, it follows that $r_c(T_\mathfrak{r})\langle c\rangle = r_c(T_s)\langle c\rangle = f$. This implies that $r_c(T_\mathfrak{r}) = (T \setminus g \cup f)_\mathfrak{r}$, which is acyclic by Lemma 2.17.

The proof is analogous when working with source-rotatability and source-turn pairs noting that the c will remain a source vertex after rotating the rotor at c from g to f. □

For the induction step of Theorem 5.8, we need to describe how (source-)rotatability is impacted by taking deletions and contractions.

Lemma 5.14. *Let (G, χ) be a 2-connected ribbon graph with $c, \mathfrak{r} \in V(G)$. Suppose that $T \in \mathcal{T}(G)$ and $T_\mathfrak{r}$ is (source-)rotatable at c from g to f on (G, χ).*

- *For any $e \in E(G) \setminus (T \cup f)$, the configuration $T_\mathfrak{r}$ is (source-)rotatable at c from g to f on $(G \setminus e, \chi \setminus e)$.*
- *For any $e \in T \setminus g$, the configuration $(T \setminus e)_\mathfrak{r}$ is (source-)rotatable at c from g to f on $(G/e, \chi/e)$.*

Proof. Both of these results follow directly from Definition 5.12. All of the defining properties of (source) rotatability still hold after contracting an edge in T or deleting an edge not in T (as long as this edge is not f or g). □

Lemma 5.15. *Let (G, χ) be a 2-connected plane graph with two spanning trees $T, T' \in \mathcal{T}(G)$ and some $\mathfrak{r} \in V(G)$. Fix a connected $F \subset T \cap T'$ with at least one edge incident to $\mathfrak{r}$. Let $(\widetilde{G}, \widetilde{\chi})$ be the ribbon graph obtained by contracting the edges in F, and $\widetilde{\mathfrak{r}}$ be the vertex these edges contract into.*

If there exists a sequence of source-rotations that take $(T \setminus F)_{\widetilde{\mathfrak{r}}}$ to $(T' \setminus F)_{\widetilde{\mathfrak{r}}}$ on $(\widetilde{G}, \widetilde{\chi})$, then there exists a sequence of source-rotations that take $T_\mathfrak{r}$ to $T'_\mathfrak{r}$ on (G, χ) without turning any rotors corresponding to edges in F.

Proof. Because F has an edge incident to $\mathfrak{r}$, each vertex of $(\widetilde{G}, \widetilde{\chi})$ other than $\widetilde{\mathfrak{r}}$ has a single preimage in (G, χ). Furthermore, for any $\widetilde{T} \in \mathcal{T}(\widetilde{G})$, and any $c \neq \widetilde{\mathfrak{r}} \in V(\widetilde{G})$, it is immediate that $\widetilde{T} \cup F \in \mathcal{T}(G)$ and $\widetilde{T}_{\mathfrak{r}'}\langle c\rangle = (\widetilde{T} \cup F)_\mathfrak{r}\langle c\rangle$.

Fix some tree $T'' \in \mathcal{T}(G)$ and suppose there exists a vertex $c \in V(\widetilde{G}) \setminus \widetilde{\mathfrak{r}}$ and edges $f, g \in E(\widetilde{G})$ such that $(T'' \setminus F)_{\widetilde{\mathfrak{r}}}$ is source-rotatable at c from g to f in $(\widetilde{G}, \widetilde{\chi})$. Then by Definition 5.12, $g = (T'' \setminus F)_{\widetilde{\mathfrak{r}}}\langle c\rangle = T''_\mathfrak{r}\langle c\rangle$, and $f = \widetilde{\chi}(c, g) = \chi(c, g)$. Lastly,

$$r_c((T'' \setminus F)_{\widetilde{\mathfrak{r}}}) = ((T'' \setminus F) \cup f \setminus g)_{\widetilde{\mathfrak{r}}} = (T'' \cup f \setminus g)_\mathfrak{r} \setminus F = r_c(T''_\mathfrak{r}) \setminus F.$$

Thus, $r_c(T''_\mathfrak{r}) = r_c((T'' \setminus F)_{\widetilde{\mathfrak{r}}}) \cup F$. Given that $r_c((T'' \setminus F)_{\widetilde{\mathfrak{r}}})$ is acyclic (by Definition 5.12) and that F contracts to a single vertex, $r_c(T''_\mathfrak{r})$ can only contain a cycle if F contains a cycle. However, $F \subseteq T$,

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

where T is a spanning tree, so this is impossible. Therefore, $r_c(T''_{\mathfrak{r}})$ is acyclic. Then, by Definition 5.12, $T''_{\mathfrak{r}}$ is rotatable at c from g to f in (G,χ). Lastly, $T''_{\mathfrak{r}}$ is source-rotatable at c because if $c \neq \widetilde{\mathfrak{r}}$ is a leaf of $T'' \setminus F$, then it is a leaf in T''.

Given that for any $\widetilde{T} \in \mathcal{T}(\widetilde{G})$, $\widetilde{\mathfrak{r}}$ has no rotors in $\widetilde{T}_{\widetilde{\mathfrak{r}}}$, this implies that any sequence of source-rotations that take $(T \setminus F)_{\widetilde{\mathfrak{r}}}$ to $(T' \setminus F)_{\widetilde{\mathfrak{r}}}$ on $(\widetilde{G}, \widetilde{\chi})$ must also take $T_{\mathfrak{r}}$ to $T'_{\mathfrak{r}}$ on (G,χ). Furthermore, the rotors corresponding to edges in F are not present in $(\widetilde{G}, \widetilde{\chi})$, so they are not rotated by any of the source rotations taking $T_{\mathfrak{r}}$ to $T'_{\mathfrak{r}}$ on (G,χ). □

Now, we are ready to prove Theorem 5.8.

Proof of Theorem 5.8. First, choose an arbitrary *root vertex* $\mathfrak{r} \in V(G)$ which we will fix throughout this proof. Our goal will be to show that there is a series of source-rotations whose composition maps from $T^{start}_{\mathfrak{r}}$ to $T^{goal}_{\mathfrak{r}}$. The theorem then follows from repeated applications of Lemma 5.13.

Next, we set up an induction argument. When G consists of a single edge, the result is trivial. Consider the following inductive assumption.

Inductive Assumption: Suppose (G', χ') is a 2-connected proper minor of (G,χ). For any pair $\widehat{T}^{start}, \widehat{T}^{goal} \in \mathcal{T}(G')$, and any vertex $\mathfrak{r}' \in V(G')$, there is a series of source-rotations whose composition maps from $\widehat{T}^{start}_{\mathfrak{r}'}$ to $\widehat{T}^{goal}_{\mathfrak{r}'}$.

Let ξ be the map from $\mathcal{T}(G) \times (V(G) \setminus \mathfrak{r}) \to \{0,1\}$ defined as follows:

$$\xi(T,x) = \begin{cases} 0 & \text{if } T_{\mathfrak{r}}\langle x\rangle = T^{goal}_{\mathfrak{r}}\langle x\rangle, \\ 1 & \text{otherwise.} \end{cases}$$

Notice that $\xi(T,x) = 0$ for all x if and only if $T = T^{goal}$. We can use ξ and the vertex partial order from Definition 5.9 to define a partial order on spanning trees, which we will write as $\prec_\xi$ (or $\preceq_\xi$ for the corresponding weak partial order).

$$\begin{aligned} T \preceq_\xi T' \iff & \text{ For all } x \in (V(G) \setminus \mathfrak{r}) \text{ such that } \xi(T,x) > \xi(T',x), \text{ there exists} \\ & y \in (V(G) \setminus \mathfrak{r}) \text{ such that } \xi(T,y) < \xi(T',y) \text{ and } x \prec_{T^{goal}_{\mathfrak{r}}} y. \end{aligned}$$

We now show that $\prec_\xi$ is a partial order. Irreflexivity and asymmetry are immediate; we just need to show transitivity. Let T, T' and T'' be spanning trees such that $T \prec_\xi T'$ and $T' \prec_\xi T''$. Let x be an arbitrary vertex such that $\xi(T,x) > \xi(T'',x)$ (if no such vertex exists, then $T \prec_\xi T''$ is immediate).

First, suppose that $\xi(T',y) = \xi(T,y)$ for all y where $x \prec_{T^{goal}_{\mathfrak{r}}} y$. Then, $\xi(T',x) \geq \xi(T,x) > \xi(T'',x)$. Because $T' \prec_\xi T''$, there is some z such that $x \prec_{T^{goal}_{\mathfrak{r}}} z$ and $\xi(T',z) < \xi(T'',z)$. Furthermore, because $\xi(T',z) = \xi(T,z)$, we also have $\xi(T,z) < \xi(T'',z)$.

Otherwise, there must be some y such that $x \prec_{T^{goal}_{\mathfrak{r}}} y$ and $\xi(T,y) < \xi(T',y)$. We can take y to be a maximal such vertex (with respect to $\prec_{T^{goal}_{\mathfrak{r}}}$). If $\xi(T',y) \leq \xi(T'',y)$, then $\xi(T,y) < \xi(T'',y)$. Otherwise, because $T' \prec_\xi T''$, there is some z such that $y \prec_{T^{goal}_{\mathfrak{r}}} z$ and $\xi(T',z) < \xi(T'',z)$. By maximality of y and the definition of $\prec_\xi$, we know that $\xi(T,z) \leq \xi(T',z)$. It follows that $\xi(T,z) < \xi(T'',z)$. Since x was arbitrary, $x \prec_{T^{goal}_{\mathfrak{r}}} y$, and $x \prec_{T^{goal}_{\mathfrak{r}}} z$, we know that $T \prec_\xi T''$.

The partial ordering $\prec_\xi$ is used to describe deviations from $T^{goal}_{\mathfrak{r}}$, prioritizing rotors closest to the root. Because $|\mathcal{T}(G)|$ is finite and T^{goal} is the unique minimal spanning tree, it suffices to show that given any $T \in \mathcal{T}(G) \setminus T^{goal}$, there is a sequence of source-rotations from $T_{\mathfrak{r}}$ to some $T''_{\mathfrak{r}}$ such that $T'' \prec_\xi T$.

Suppose that there exists some $x \in V(G)$ such that $\xi(T,x) = 1$ and x is a leaf vertex of T. Then, we can freely rotate the rotor at x using source-rotations until we get a configuration $T''_{\mathfrak{r}}$ such that $\xi(T'',x) = 0$. Because we only rotated the rotor at x, it follows that $T'' \prec_\xi T$ and we are done.

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

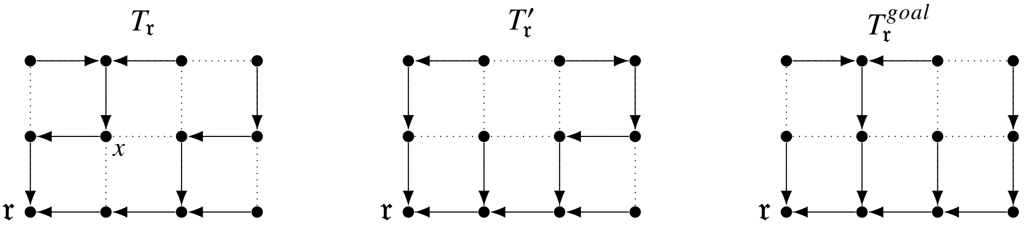

Figure 5. *In this example, with $\mathfrak{r}$ as our root vertex, there are no source-turn moves that immediately bring T closer to T^{goal}. However, if we choose a vertex x such that every $v \prec_{T_\mathfrak{r}} x$ is in the correct position, then we can turn these rotors until x is the source of a source-turn pair, and then turn the rotor at x. This gives us the spanning tree T'. Notice that $T' \prec_\xi T$ as desired.*

Otherwise, there must exist some $x \in V(G) \setminus \mathfrak{r}$ such that $\xi(T,x) = 1$ and for all $v \prec_{T_\mathfrak{r}} x$, we have $\xi(T,v) = 0$. Let $g = T_\mathfrak{r}\langle x \rangle$, $f = \chi(x,g)$ and $\{x,y\} = \mathfrak{i}(f)$. Consider the sets

$$V^x = \{v \mid v \prec_{T_\mathfrak{r}} x\} \text{ and } E^x = \{T_\mathfrak{r}\langle v \rangle \mid v \in V^x\}.$$

By assumption, all of the rotors associated with vertices in V^x are already in their final position and are directed toward x. This implies that

$$V^x \subseteq \{v \mid v \prec_{T_\mathfrak{r}^{goal}} x\}.$$

We now introduce the following claim:

Claim: There exists some $T' \in \mathcal{T}(G)$ such that x is a leaf of T', and $T'_\mathfrak{r}$ can be reached from $T_\mathfrak{r}$ after a sequence of source-rotations at vertices in V^x.

If the claim is true, then we can rotate the rotor at x until it reaches $T^{goal}\langle x \rangle$ and call the new tree T''. By construction, $\xi(T'',x) < \xi(T',x) = \xi(T,x)$. Furthermore, since all source-rotations were at vertices in $V_x \subseteq \{v \mid v \prec_{T_\mathfrak{r}^{goal}} x\}$, it follows that $\xi(T'',v) = \xi(T,v)$ for all $v \not\prec_{T_\mathfrak{r}^{goal}} x$. Thus, $T'' \prec_\xi T$.

All that is left is to prove the claim. By Lemma 5.10, the set $T \setminus E^x$ has only one edge incident to x. Because (G,χ) is 2-connected, x is not a cut vertex. Thus, there exists a spanning tree $T' \in \mathcal{T}(G)$ such that $(T \setminus E^x) \subseteq T'$ and x is a leaf edge of T'. Let F be the set of edges in $T \setminus E^x$. Notice that F consists of all of the edges of T that are on the same component of T as $\mathfrak{r}$ when the tree is severed at the point x. In particular, F is connected and contains at least one edge incident to $\mathfrak{r}$.

If we contract all of the edges in F, we obtain a new ribbon graph $(\widetilde{G},\widetilde{\chi})$ such that $V(\widetilde{G}) = V^x \cup x$. Furthermore, $T \cap E(\widetilde{G})$ and $T' \cap E(\widetilde{G})$ are both in $\mathcal{T}(\widetilde{G})$. Let $\widetilde{\mathfrak{r}}$ be the vertex formed by contraction. The vertices other than $\widetilde{\mathfrak{r}}$ cannot be cut vertices of $(\widetilde{G},\widetilde{\chi})$ because they are not cut vertices of (G,χ). If $\widetilde{\mathfrak{r}}$ is a cut vertex, we can consider each 2-connected component separately. In either case, we can apply the induction hypothesis.

By the induction hypothesis, there is a sequence of source-rotations whose composition maps from $T \cap E(\widetilde{G})$ to $T' \cap E(\widetilde{G})$ on $(\widetilde{G},\widetilde{\chi})$. Thus, by Lemma 5.15, there is a sequence of source-rotations that take $T_\mathfrak{r}$ to $T'_\mathfrak{r}$ on (G,χ) without turning any rotors corresponding to edges in F. The claim follows from the fact that the edges in $T \setminus F$ correspond to the rotors at vertices in V^x. □

Notice that it is possible to rotate a rotor at a leaf vertex to any position through a series of source-turn moves. This means that the ribbon structure is not actually relevant for Theorem 5.8. In particular, we get the following corollary, which may be of independent interest.

Corollary 5.16. *Let G be a 2-connected graph and $T^{start}, T^{goal} \in \mathcal{T}(G)$. For some $k \in \mathbb{Z}_{\geq 0}$, there exists a sequence $(T^0 = T^{start}, T^1, \ldots, T^k = T^{goal}) \in \mathcal{T}(G)^{k+1}$ such that for every $i \in [1,k]$, we have $|T^{i-1} \setminus T^i| = |T^i \setminus T^{i-1}| = 1$ and $T^{i-1} \setminus T^i$ is a leaf edge of T^{i-1}.*

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

Remark 5.17. Corollary 5.16 is a generalization of a well-known matroidal result. If we removed the requirement that $T^{i-1} \setminus T^i$ is a leaf edge, then the result would follow from the fact that the *fundamental graph* of any *connected* matroid is connected [Oxl06, Proposition 4.3.2], and the *cycle matroid* of a 2-connected graph is connected [Oxl06, Proposition 4.1.7].

Corollary 5.18. *Let α be a consistent sandpile torsor algorithm and ρ be the rotor-routing algorithm. The following are equivalent:*

1. $\alpha = r$.
2. *For all 2-connected plane graphs (G, χ) and all single-step pairs $(c - s, T)$, we have*

$$\alpha_{(G,\chi)}([c - s], T) = r_{(G,\chi)}([c - s], T).$$

3. *For all 2-connected plane graphs (G, χ) and all source-turn pairs $(c - s, T)$, we have*

$$\alpha_{(G,\chi)}([c - s], T) = r_{(G,\chi)}([c - s], T).$$

Proof. (1) → (2) is immediate. (2) → (3) follows from Lemma 5.7. (3) → (1) follows from Theorem 5.8 and Lemma 5.5. □

Corollary 5.18 greatly simplifies the task of proving that a sandpile torsor algorithm is equivalent to rotor-routing and will be essential in our proof of Theorem 5.4. Before proving this theorem, we give one more definition that will aid our proof.

Definition 5.19. We say that $(s - c, T)$ is a *reverse single-step pair from f to g* if $f \in T$, $g \notin T$, and $(c - s, T \setminus f \cup g)$ is a single-step pair from g to f.

Lemma 5.20. *If $(s - c, T)$ is a reverse single-step pair from f to g, then $r_{(G,\chi)}([s - c], T) = T \setminus f \cup g$.*

Proof.

$$\begin{aligned} r_{(G,\chi)}([s - c], T) &= r_{(G,\chi)}\big([s - c], r_{(G,\chi)}([c - s], T \setminus f \cup g)\big) \\ &= r_{(G,\chi)}([0], T \setminus f \cup g)) = T \setminus f \cup g. \end{aligned}$$

□

We are now ready to prove the majority of Theorem 5.4. However, the final case of the theorem is particularly involved, so we relegate its proof to Appendix B.

Proof of Theorem 5.4. Let α be an arbitrary consistent sandpile torsor algorithm. We will prove the theorem by induction, but we first consider a few special plane graphs.

Let C_3 be the 3-cycle along with its unique ribbon structure, and let E_3 be the triple edge with a planar ribbon structure. Each of these plane graphs has two sandpile torsor actions. On C_3, one of these actions is equivalent to both r_{C_3} and $\overline{r}_{C_3}$, whereas the other is equivalent to both $r^{-1}_{C_3}$ and $\overline{r}^{-1}_{C_3}$. On E_3, one of the actions is equivalent to both r_{E_3} and $\overline{r}^{-1}_{E_3}$, whereas the other is equivalent to both $\overline{r}_{E_3}$ and $r^{-1}_{E_3}$. This means that α_{C_3} and α_{E_3} will simultaneously match precisely one of the four sandpile torsor algorithms with the same structure as rotor-routing. For this proof, we assume that $\alpha_{C_3} = r_{C_3}$ and $\alpha_{E_3} = r_{E_3}$. By Proposition 5.2, we can make this assumption without loss of generality.

Let (G, χ) be an arbitrary 2-connected plane graph. For the induction hypothesis, assume that $\alpha_{(G',\chi')} = r_{(G',\chi')}$ for all proper minors (G', χ') of (G, χ). By Corollary 5.18, it suffices to show that $\alpha_{(G,\chi)}([c - s], T) = r_{(G,\chi)}([c - s], T)$ for every source-turn pair $(c - s, T)$. Throughout this proof, $(c - s, T)$ is an arbitrary source-turn pair from g to f and

$$\widehat{T} := \alpha_{(G,\chi)}([c - s], T).$$

To prove the theorem, we need to show that $\widehat{T} = r_{(G,\chi)}([c - s], T) = T \setminus g \cup f$.

First, suppose that there exists some $e \neq g$ such that $e \in T \cap \widehat{T}$. This implies that $\mathfrak{i}(e) \neq \{c, s\}$ because $T_s\langle c\rangle = g$ (if $\mathfrak{i}(e) = \{c, s\}$ and $e \in T$, then $T_s\langle c\rangle = e \neq g$, which presents a contradiction). By the condition that α is consistent, we must have $\alpha_{(G/e,\chi/e)}([c-s], T\setminus e) = \widehat{T}\setminus e$. By the induction hypothesis, $\alpha_{(G/e,\chi/e)}([c-s], T\setminus e) = r_{(G/e,\chi/e)}([c-s], T\setminus e)$. Furthermore, it follows from Lemmas 5.14 and 5.13 that $(c-s, T)$ is a source-turn pair for $(G/e, \chi/e)$. This means that $r_{(G/e,\chi/e)}([c-s], T) = (T\setminus g \cup f)\setminus e$. It follows that $\widehat{T} = T \setminus g \cup f = r_{(G,\chi)}([c-s], T)$. Similarly, suppose that there exists some $e \neq f$ such that $e \notin T \cup \widehat{T}$. Then, by an analogous argument, we have $\alpha_{(G\setminus e,\chi\setminus e)}([c-s], T) = \widehat{T}$ and $\widehat{T} = T \setminus g \cup f = r_{(G,\chi)}([c-s], T)$. In both cases, the theorem holds by induction. Therefore, we may assume that the only shared edges or non-edges of T and $\widehat{T}$ are in $\{f, g\}$. In particular, we have the relation

$$E(G) \setminus (T\Delta\widehat{T}) \subseteq \{f, g\}, \tag{2}$$

where Δ denotes the symmetric difference operator. Note that this argument still holds when $(c-s, T)$ is a single-step pair or a reverse single-step pair. This leaves us with four cases to consider corresponding to the four subsets of $\{f, g\}$.

Case 1: We have $f \in \widehat{T}$ and $g \in \widehat{T}$.

In this case, the number of edges of G must be one less than double the size of each spanning tree.

Let $\mathfrak{i}(g) = \{x, c\}$. Because $g \in T \cap \alpha_{(G,\chi)}([c-s], T)$, we can apply Definition 4.3 and the induction hypothesis to show that

$$\alpha_{(G,\chi)}([c-s], T) \setminus g = \alpha_{(G/g,\chi/g)}([c-s], T \setminus g) = r_{(G/g,\chi/g)}([c-s], T \setminus g).$$

Because $(c-s, T)$ is a source-turn pair (and thus also a single-step pair) from g to f, we must have $T_s\langle c\rangle = g$. By Definition 5.9 and Lemma 5.10, this implies that the path of edges $(e_1, e_2, \ldots, e_k)$ on T from x to s does not contain g or any edges parallel to g. Furthermore, these edges must also form a path from z to s on G/g (where z is the vertex formed by contracting g). By Lemma 3.4, any edges to the right of the directed cycle formed by $(e_1, e_2, \ldots, e_k, f)$ are in $r_{(G/g,\chi/g)}([z-s], T \setminus g) = \widehat{T} \setminus g$ if and only if they are in $T \setminus g$. This implies that for any such edge e, we have $e \notin \{f, g\}$ and $e \notin T\Delta\widehat{T}$, which contradicts (2). Thus, there can be no edges to the right of the cycle. In particular, this means that $(\chi/g)(z, f) = e_1$ and $\chi(c, f) = g$.

By Definition 5.6, we know that $\chi(c, g) = f$. Together, the facts that $\chi(c, g) = f$ and $\chi(c, f) = g$ imply that c is a vertex of degree 2. This means that a single firing of vertex c sends $2c - s - x$ to 0. Thus, $[2c - s - x] = [0]$ and $[c - s] = [x - c]$.

Let $T' = T \setminus g \cup f$. It follows from Definition 5.6 that $(c - x, T')$ is a source-turn pair from f to g. Suppose that $\alpha_{(G,\chi)}([c-x], T') = T$. Then, this would mean that $\alpha_{(G,\chi)}([x-c], T) = T'$. However, we showed that

$$\alpha_{(G,\chi)}([x-c], T) = \alpha_{(G,\chi)}([c-s], T) = \widehat{T} \neq T'.$$

Because $\alpha_{(G,\chi)}([c-x], T') \neq T$, this action must swap every edge of T' other than f and g. Furthermore, since the number of edges in G is one less than double the size of each spanning tree, both f and g must be in $\alpha_{(G,\chi)}([c-x], T')$. It follows that $\alpha_{(G,\chi)}([c-x], T') = \widehat{T}$.

Notice that the edges $(g, e_k, e_{k-1}, \ldots, e_1)$ form a directed cycle on the graph G/f. Using our earlier argument, we can conclude that there are no edges to the right of this cycle. Furthermore, the rotors of this cycle are oriented in the opposite direction as they were on G/g. Together with the fact that c is a degree 2 vertex, this implies that $\{e_1, e_2, \ldots, e_k, f, g\}$ forms a cycle in G with no edges on either side. In particular, this means that G is a cycle.

It follows that $\widehat{T} = \{f, g\}$. The only cycle with a 2-edge spanning tree is the 3-cycle C_3. Thus, $\alpha_{(G,\chi)} = r_{(G,\chi)}$ by our assumption that $\alpha_{C_3} = r_{C_3}$.

Case 2: We have $f \notin \widehat{T}$ and $g \in \widehat{T}$.

Let $\mathfrak{i}(g) = \{x, c\}$. By the same argument that we used in the previous proof, c has degree 2 and $[c - s] = [x - c]$. This means that

$$\alpha_{(G,\chi)}([x - c], T) = \alpha_{(G,\chi)}([c - s], T) = \widehat{T}.$$

Because $f \notin T \cup \widehat{T}$, we can use consistency to conclude that $\alpha_{(G\setminus f,\chi\setminus f)}([x - c], T) = \widehat{T}$. However, g is a cut edge of $G \setminus f$. Thus, $[x - c] = 0$ on $G \setminus f$. This is a contradiction by (2), since $T \neq \widehat{T}$ unless $E(G) = \{f, g\}$. If $E(G) = \{f, g\}$, we still have a contradiction because $[c - s]$ is a nontrivial element of $\mathrm{Pic}^0(G)$, so it cannot fix any spanning trees. Thus, Case 2 is impossible.

Case 3: We have $f \notin \widehat{T}$ and $g \notin \widehat{T}$.

This case is a bit more complicated than the previous two, so we begin with a brief overview of our argument.

Step 1: First, we show that for any source turn pair $(c''-s'', T'')$, we either have $\alpha_{(G,\chi)}([c''-s''], T'') = r_{(G,\chi)}([c'' - s''], T'')$ or $\alpha_{(G,\chi)}([2(c'' - s'')], T'') = r_{(G,\chi)}([c'' - s''], T'')$ (see (3)).

Step 2: This observation allows us to define a sequence of sandpile elements such that the action by $\alpha_{(G,\chi)}$ on T'' rotates the rotor at c'' all the way around until returning to T''. In particular, these sandpile elements must add to the identity (see (4)).

Step 3: Using properties of the sandpile group, we prove that all the sandpile elements defined in the previous step have a particular form.

Step 4: Next, we use the fact that $\alpha_{(G,\chi)}$ must always output a spanning tree in order to deduce that there must be exactly three edges incident to c.

Step 5: After this, we show that when Case 3 occurs and c is incident to exactly three edges, these must be the only edges of G. In particular, it follows that $(G, \chi) = E_3$ (a planar embedding of the triple edge).

Step 6: Finally, we know that $r_{E_3} = \alpha_{E_3}$ by a base case of our induction.

Let $T' = T \setminus g$. By assumption, $\alpha_{(G,\chi)}([c - s], T) \neq T' \cup f$. Thus, $\alpha_{(G,\chi)}([s - c], T' \cup f) \neq T$. However, note that $(s - c, T' \cup f)$ is a reverse single-step pair. Because (2) holds for reverse single-step pairs, every edge other than f and g must swap. In addition, because the total number of edges in each spanning tree is consistent, neither f nor g can be in $\alpha_{(G,\chi)}([s - c], T' \cup f)$. This means that $\alpha_{(G,\chi)}([s-c], T' \cup f) = \widehat{T}$ and $\alpha_{(G,\chi)}([c-s], \widehat{T}) = T' \cup f$. We obtain the following chain of equalities:

$$\alpha_{(G,\chi)}([2c - 2s], T) = \alpha_{(G,\chi)}([c - s], \widehat{T}) = T' \cup f = r_{(G,\chi)}([c - s], T).$$

Note that the reasoning in the previous paragraph holds for any source-turn pair. In particular, if $(c'' - s'', T'')$ is a source-turn pair, then

$$\alpha_{(G,\chi)}([a(c'' - s'')], T'') = r_{(G,\chi)}([c'' - s''], T'') \text{ for some } a \in \{1, 2\}. \tag{3}$$

Let $\chi(c) = (e_1, e_2, \ldots, e_k)$, where $e_1 = g$, and set $\mathfrak{i}(e_i) = \{c, s_i\}$. Note that $e_2 = f$, $s_2 = s$, and the s_i are allowed to coincide. Because $(c - s, T)$ is a source-turn pair, it follows that $(c - s_{i+1}, T' \cup e_i)$ is also a source-turn pair for any $i \in [1, k]$. Thus, by (3), there exists a function $\phi : [1, k] \to \{1, 2\}$ such that for all $i \in [1, k]$,

$$\alpha_{(G,\chi)}([\phi(i)(c - s_{i+1})], T' \cup e_i) = r_{(G,\chi)}([c - s_{i+1}], T' \cup e_i),$$

where $i+1$ is replaced with 1 for $i=k$. It follows that

$$\alpha_{(G,\chi)}\left(\left[\sum_{i=1}^{k}\phi(i)(c-s_i)\right],T\right)=r_{(G,\chi)}\left(\left[\sum_{i=1}^{k}(c-s_i)\right],T\right)=r_{(G,\chi)}([0],T)=T, \qquad (4)$$

where we use the fact that

$$\left[\sum_{i=1}^{k}(c-s_i)\right]=[0],$$

because after firing c, we return to the identity. Since α is free, this implies that

$$\left[\sum_{i=1}^{k}\phi(i)(c-s_i)\right]=\left[\sum_{i=1}^{k}(\phi(i)-1)(c-s_i)\right]=[0].$$

The divisor $\sum_{i=1}^{k}(\phi(i)-1)(c-s_i)$ has -1 chips on each s_i for which $\phi(i)=2$, chips on c equal to the number of such i, and no chips elsewhere. By [McD21a, Lemma 17], and the assumption that c is not a cut vertex, this divisor is only equivalent to the identity if $\phi(i)=2$ for all i or $\phi(i)=1$ for all i. By the assumption that we are in Case 3, $\phi(1)\neq 1$, which means that $\phi(i)=2$ for all $i\in[1,k]$.

Next, we consider the possible values of k (the number of edges incident to c). Recall that $\widehat{T}=\alpha_{(G,\chi)}([c-s],T)$. Because $(c-s,T)$ is a source-turn pair, $e_j\notin T$ for $j>1$. Furthermore, since we are in Case 3, this implies that $e_j\in\widehat{T}$ if and only if $j>2$. If $k\leq 2$, then $\widehat{T}$ has no edges connecting to c, which is impossible. Since $e_1\notin\widehat{T}$, there must be a path from s_1 to some other $s_i\neq s_1$ along $\widehat{T}$ that does not pass through the vertex c. If $k>4$, then by pigeonhole principle, there must be a pair s_j and s_{j+1} such that neither vertex is s_1 or s_i.

We know that $\phi(j)=2$, which means that $\alpha_{(G,\chi)}([c-s_{j+1}],T'\cup e_j)\neq T'\cup e_{j+1}$. The only other possibility is that

$$\alpha_{(G,\chi)}([c-s_{j+1}],T'\cup e_j)=\widehat{T}\cup\{e_1,e_2\}\setminus\{e_j,e_{j+1}\}.$$

However, $\widehat{T}\cup\{e_1,e_2\}\setminus\{e_j,e_{j+1}\}$ contains e_1, e_i and a second path from s_1 to s_i. This means that it must contain a cycle, which is a contradiction.

When $k=4$, the situation is similar. The argument that we used for $k>4$ still works if we can show that there is a path along $\widehat{T}$ from s_1 to s_2 or s_4 that does not pass through c. In particular, we let $j=3$ in the first case and $j=2$ in the second. We will show that such a path always exists (where throughout this paragraph, all of our paths are along $\widehat{T}$). First, we note that if every path from s_1 to s_4 goes through c, then there must be a path P_1 from s_1 to s_3 that does not pass through c. If there is a path from s_3 to s_2 that does not pass through c, then by combining that path with P_1, we get a path from s_1 to s_2 that does not pass through c as desired. Otherwise, there must be a path P_2 from s_2 to s_4 that does not pass through c. Furthermore, note that s_4 and s_2 are on opposite sides of the cycle $P_1\cup\{e_1,e_3\}$. By planarity, this implies that P_1 and P_2 must intersect at some point v. If we combine the path from s_1 to v along P_1 and the path from v to s_2 along P_2, we get a path from s_1 to s_2 that does not pass through c.

We have now shown that we must have $k=3$ for this case to be possible. Recall that $\widehat{T}:=\alpha_{(G,\chi)}([c-s_2],T)=(E(G)\setminus T)\setminus e_2$. Because $k=3$, $\widehat{T}$ contains exactly one edge incident to c, namely e_3. In particular, $([c-s_1],\widehat{T})$ is a source-turn pair. Let $\widetilde{T}=\alpha_{(G,\chi)}([c-s_1],\widehat{T})$. By (2), there are only 2 possibilities for $\widetilde{T}$: either $\widetilde{T}=\widehat{T}\setminus e_3\cup e_1$ or $\widetilde{T}=T'\cup e_2$.

In either case, we have the following:

$$\alpha_{(G,\chi)}([c-s_3],\widetilde{T})=\alpha_{(G,\chi)}([(c-s_3)+(c-s_1)+(c-s_2)],T)=\alpha_{(G,\chi)}([0],T)=T.$$

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

If $\widetilde{T} = \widehat{T} \setminus e_3 \cup e_1$, then $e_2 \notin \widetilde{T} \cup T$. Thus, by consistency, we have

$$\alpha_{(G\setminus e_2, \chi\setminus e_2)}([c - s_3], \widetilde{T}) = T.$$

By our induction hypothesis,

$$\alpha_{(G\setminus e_2, \chi\setminus e_2)}([c - s_3], \widetilde{T}) = r_{(G\setminus e_2, \chi\setminus e_2)}([c - s_3], \widetilde{T}) = \widetilde{T} \setminus e_1 \cup e_3.$$

This is a contradiction because $e_3 \notin T$. It follows that we must have $\widetilde{T} = T' \cup e_2$. Notice that $(c - s_3, T' \cup e_2)$ is a source-turn pair and $r_{(G,\chi)}([c - s_3], T' \cup e_2) = T' \cup e_3 \neq T$. By (2), this means that $\alpha_{(G,\chi)}([c - s_3], T' \cup e_2)$ cannot share any edges with $T' \cup e_2$, and the only shared non-edge is e_3. However, we showed above that

$$\alpha_{(G,\chi)}([c - s_3], T' \cup e_2) = \alpha_{(G,\chi)}([c - s_3], \widetilde{T}) = T = T' \cup e_1.$$

It follows that $E(G) = \{e_1, e_2, e_3\}$ and, by the 2-connectedness condition, (G, χ) must be E_3, a planar embedding of the triple edge. Thus, $\alpha_{(G,\chi)} = r_{(G,\chi)}$ by our assumption that $\alpha_{E_3} = r_{E_3}$.

Case 4: We have $f \in \widehat{T}$ and $g \notin \widehat{T}$.

Most of the work for this case is given in Appendix B, and there is an overview of our argument just before Lemma B.1. If (G, χ) is a *telescope graph* (see Definition B.8), then $\alpha_{(G,\chi)}([c - s], T) = r_{(G,\chi)}([c - s], T)$ by Lemma B.13. If (G, χ) is not a telescope graph, then $\alpha_{(G,\chi)}([c - s], T) = r_{(G,\chi)}([c - s], T)$ by Lemma B.12. □

Definition 5.21. A *consistent sandpile torsor action* on a plane graph (G, χ) is a sandpile torsor action obtained from a consistent sandpile torsor algorithm.

Corollary 5.22. *Let (G, χ) be a 2-connected plane graph. For $k \geq 1$, let E_k be the plane graph with 2 vertices and k parallel edges. For $k \geq 1$, let C_k be the cycle graph with k edges (where we define $C_1 = E_1$ and $C_2 = E_2$).*

- *If $(G, \chi) = E_1 = C_1$ or $(G, \chi) = E_2 = C_2$, then (G, χ) has exactly one consistent sandpile torsor action.*
- *If $(G, \chi) = E_k$ or $(G, \chi) = C_k$ for $k \geq 3$, then (G, χ) has exactly two consistent sandpile torsor actions.*
- *If (G, χ) is any other 2-connected plane graph, then (G, χ) has exactly four consistent sandpile torsor actions.*

Proof. We showed in Theorem 5.4 that every consistent sandpile torsor action must be equivalent to $r_{(G,\chi)}, \overline{r}_{(G,\chi)}, r^{-1}_{(G,\chi)}$ or $\overline{r}^{-1}_{(G,\chi)}$.

If (G, χ) is 2-connected, but not equal to C_k for some k, then it must have a vertex c with degree at least 3. By 2-connectedness, there exists some $T \in \mathcal{T}(G)$ such that c is a leaf vertex of T. Let g be the unique edge of T incident to x. Furthermore, let $f = \chi(c, g)$, $h = \chi(c, f)$, and $\{c, s\} = \mathfrak{i}(f)$. It follows from the definition of rotor-routing that

$$r_{(G,\chi)}([c - s], T) = T \setminus g \cup f = \overline{r}_{(G,\chi)}([c - s], T \setminus g \cup h).$$

Because $T \neq T \setminus g \cup h$, we can conclude that $r_{(G,\chi)} \neq \overline{r}_{(G,\chi)}$ and $r^{-1}_{(G,\chi)} \neq \overline{r}_{(G,\chi)}$ for any 2-connected (G, χ) not equal to C_k for some k.

If (G, χ) is 2-connected, but not equal to E_k for some k, then it must have at least three vertices. If we remove all but one edge out of every set of parallel edges, the resulting graph must contain a cycle or there would be a cut edge. After adding in the parallel edges (taking the innermost edge of each), the cycle remains and forms a face of G that is bounded by at least three edges. Let C be the set of edges

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

which bound this face and choose an arbitrary $f \in C$. Since $C \setminus f$ is acyclic, there must exist some $T \in \mathcal{T}(G)$ such that $C \setminus f \in T$. Suppose that when considering the edges of C in counterclockwise order, the edge h comes just before f, and the edge g comes right after f. Furthermore, let $c = \mathfrak{i}(f) \cap \mathfrak{i}(g)$ and $s = \mathfrak{i}(f) \cap \mathfrak{i}(h)$. It follows from the definition of rotor-routing that

$$r_{(G,\chi)}([c-s], T) = T \setminus g \cup f \text{ and } \overline{r}^{-1}_{(G,\chi)}([c-s], T) = \overline{r}_{(G,\chi)}([s-c], T) = T \setminus h \cup f.$$

Because $T \setminus g \cup f \neq T \setminus h \cup f$, we can conclude that $r_{(G,\chi)} \neq \overline{r}^{-1}_{(G,\chi)}$ and $r^{-1}_{(G,\chi)} \neq \overline{r}_{(G,\chi)}$ for any 2-connected (G, χ) not equal to E_k for any k.

When $(G, \chi) = E_k$ for some k, the actions $r_{(G,\chi)}$ and $\overline{r}^{-1}_{(G,\chi)}$ as well as $\overline{r}_{(G,\chi)}$ and $r^{-1}_{(G,\chi)}$ are equivalent since both vertices are incident to the same edges, but in the opposite order. When $(G, \chi) = C_k$ for some k, the actions $r_{(G,\chi)}$ and $\overline{r}_{(G,\chi)}$ as well as $r^{-1}_{(G,\chi)}$ and $\overline{r}^{-1}_{(G,\chi)}$ are equivalent since reversing the ribbon structure has no effect on C_k.

The result follows if we show that $r_{(G,\chi)} \neq r^{-1}_{(G,\chi)}$ when (G, χ) is a 2-connected ribbon graph other than $E_1 = C_1$ or $E_2 = C_2$. In particular, we need to show that for some $D \in \mathrm{Div}^0(G)$, we have $[2D] \neq [0]$. There are a few ways to show this, but we will use the chip-firing perspective.

First, suppose that every vertex of G has degree at most 2. By our previous reasoning, (G, χ) must be equal to C_k for some k. It is well-known and straightforward to check that $\mathrm{Pic}^0(C_k) = \mathbb{Z}/k\mathbb{Z}$ ([CP18, Problem 2.7]). Thus, for $k > 2$, there are elements not equal to their own inverse. This means that we can assume that there is some $s \in V(G)$ such that $\deg(s) > 2$. Let $c \in V(G)$ be any other vertex. We will show that $[2c - 2s] \neq [0]$ and the result follows. First, recall that $[2c - 2s] = [0]$ if and only if there is a sequence of firing moves from $2c - 2s$ to 0. Suppose that such a sequence of firings exist. Since firing every vertex is equivalent to firing no vertices, we can assume that not every vertex is fired. The only way to reduce the number of chips on a vertex is by firing it, which means that c must fire. After c fires, any adjacent vertices other than s have a positive number of chips, so they must also fire. By recursion, it follows that a vertex v must fire if there is a path from c to v that does not pass through s. By 2-connectedness, this means that every vertex other than s must fire at least once. After these firings, s has $\deg(s) - 2$ chips on it, which is positive by assumption. Thus, s must fire as well. However, this is a contradiction because we assumed that not every vertex is fired. Thus, $[2c - 2s] \neq [0]$ and $r_{(G,\chi)} \neq r^{-1}_{(G,\chi)}$. □

6. Extension to regular matroids

Many properties of graphs can be generalized to *regular matroids*. In this section, we will explore how the ideas from the previous sections can be applied to regular matroids and we give conjectures for extensions of Theorems 4.6 and 5.4.

For this section, we assume that the reader is familiar with oriented matroid theory (for necessary definitions, see e.g. [Oxl06, BLVS+99]). However, we will not need any of the results from previous sections.

6.1. The BBY algorithm

Suppose that $M = (E, \mathcal{B}, \chi)$ is an oriented matroid that is also regular. In particular, E is the *ground set*, $\mathcal{B}$ is the set of *bases* and χ is a *chirotope* which gives M its orientation.

Remark 6.1. Throughout this section, all of our regular matroids will be oriented even if we do not explicitly say so. This orientation is arbitrary and has only a minor cosmetic effect on the results.

Bacher, de la Harpe and Nagnibeda showed that the sandpile group of a graph can also be described in terms of a graph's *flow lattice* and *cut lattice* [BLHN97]. The analogues on regular matroids are the *circuit lattice* and *cocircuit lattice*. Consider the set of *signed circuits* of M, which we write as vectors

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

in $\{+1,-1,0\}^E$. These vectors generate a sublattice of $\mathbb{Z}^E$ called the *circuit lattice*. Similarly, the *signed cocircuits* of M generate a sublattice of $\mathbb{Z}^E$ called the *cocircuit lattice*.

Definition 6.2. The *sandpile group* of a regular matroid M, written $\mathcal{S}(M)$, is the quotient of $\mathbb{Z}^E$ by the direct sum of the circuit and cocircuit lattices.

This sandpile group is also called the *Jacobian* or *critical group* of M. When M is the *cycle matroid* associated with a connected graph G, the groups $\mathcal{S}(M)$ and $\mathrm{Pic}^0(G)$ are isomorphic (see [BLHN97, Big99, McD21b] for more details on this connection).

Theorem 6.3 (sandpile matrix-tree theorem for regular matroids [Mer99, Theorem 4.6.1]). *For any regular matroid $M = (E, \mathcal{B}, \chi)$, we have $|\mathcal{S}(M)| = |\mathcal{B}|$.*

For a graph G, we defined a sandpile torsor action to be a free transitive action of $\mathrm{Pic}^0(G)$ on $\mathcal{T}(G)$. Additionally, for our action to be 'natural' on a graph, we needed to introduce a ribbon structure and restrict to plane graphs. The analogue for regular matroids is to find a free transitive action of $\mathcal{S}(M)$ on $\mathcal{B}$ after introducing some additional structure on M. Backman, Baker and Yuen constructed a family of such actions which depend on a choice of *acyclic circuit and cocircuit signatures* [BBY19, Yue18]. We will describe their construction below, but first we need a few definitions.

For convenience, we will sometimes express elements of $\mathbb{Z}^E$ as integer linear combinations of ground set elements and sometimes as integer vectors. For example, if $E = \{e_1, e_2, e_3, e_4, e_5\}$, then $e_1 + 3e_3$ and $(1,0,3,0,0)$ represent equivalent elements of $\mathbb{Z}^E$. Given $D \in \mathbb{Z}^E$, we write $[D]$ for the equivalence class of $\mathcal{S}(M)$ containing D. Notice that $\mathcal{S}(M)$ is generated by elements of the form $[e]$ for $e \in E$.

For each circuit (resp. cocircuit), there are two signed circuits (resp. cocircuits) which differ by multiplication by -1. A *circuit signature* (resp. *cocircuit signature*) is a choice of one signed circuit (resp. cocircuit) for each unsigned circuit (resp. cocircuit). A circuit signature (resp. cocircuit signature) is called *acyclic* if no positive linear combination of signed circuits (resp. cocircuits) adds to zero. We will call (σ, σ^*) a *pair of acyclic signatures* if σ is an acyclic circuit signature and σ^* is an acyclic cocircuit signature.

Definition 6.4. A *matroidal sandpile torsor algorithm* α is a function whose input is a regular matroid $M = (E, \mathcal{B}, \chi)$ and a pair of acyclic signatures (σ, σ^*), and whose output is a free transitive action of $\mathcal{S}(M)$ on $\mathcal{B}$. For a specific M and signatures (σ, σ^*), we write $\alpha_{(M,\sigma,\sigma^*)}$ for the free transitive action, which we call a *matroidal sandpile torsor action* on M.

Remark 6.5. The acyclic signatures in Definition 6.4 play a similar role to the ribbon structure in Definition 2.8; they are necessary to resolve ambiguity and allow us to define 'natural' torsor actions. Definition 6.4 should also require a form of automorphism invariance, but this detail is more distracting than enlightening for our purposes.

Backman, Baker and Yuen introduced a matroidal sandpile torsor algorithm which will denote BBY [BBY19]. Their construction is a generalization of the *Bernardi algorithm* which we discussed in Section 2 [Yue17].

The key insight for defining BBY is that, given a pair of acyclic signatures, every $B \in \mathcal{B}$ can be associated with a $\{0,1\}^E$ vector such that these vectors are all distinct as elements of $\mathcal{S}(M)$. By Theorem 6.3, this means that every equivalence class of $\mathcal{S}(M)$ is associated with a unique element of $\mathcal{B}$. Thus, we obtain a free transitive action through pointwise addition. All that is left is to show how to associate a basis with a $\{0,1\}^E$ vector.

For any $e \notin B$, there is a unique circuit contained in $e \cup B$, which is called the *fundamental circuit of* e (see [Oxl06, Corollary 1.2.6]). Similarly, for any $e \in B$, there is a unique cocircuit contained in $e \cup (E \setminus B)$, which is called the *fundamental cocircuit of* e. To determine whether the e entry of our vector should be 1 or 0, first check whether or not e is in B. If $e \notin B$, then consider the fundamental circuit C of e. The signature σ contains one of the the two signed circuits associated with C. If e is positive in this signed circuit, then our e entry is 1; otherwise, our e entry is 0. Similarly, if $e \in B$, then

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

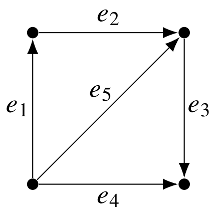

Figure 6. *In Example 6.8, we consider the oriented matroid represented by the oriented graph above.*

consider the fundamental cocircuit of e and assign the value based on the associated signed cocircuit in σ^*. See Example 6.8 for a demonstration of this algorithm.

Remark 6.6. The construction given by Backman, Baker and Yuen is stated differently than our construction. In particular, the original formulation involves the set $\mathcal{G}$ of *circuit-cocircuit minimal orientations*. Nevertheless, these two perspectives are equivalent (see [McD21b, Section 6.5]).

Theorem 6.7 [BBY19, Theorem 1.2.2]. *The above construction produces a matroidal sandpile torsor algorithm.*

Backman, Baker and Yuen prove this theorem geometrically, using *zonotopal tilings*.

Example 6.8. The oriented graph in Figure 6 *represents* a regular matroid, which we call M. The pair (σ, σ^*) below is a pair of acyclic signatures for M.[6]

$$\sigma = \begin{array}{c} \begin{array}{ccccc} e_1 & e_2 & e_3 & e_4 & e_5 \end{array} \\ (\ +1,\ +1,\ 0,\ 0,\ -1\ ) \\ (\ +1,\ +1,\ +1,\ -1,\ 0\ ) \\ (\ 0,\ 0,\ +1,\ -1,\ +1\ ) \end{array} \quad \sigma^* = \begin{array}{c} \begin{array}{ccccc} e_1 & e_2 & e_3 & e_4 & e_5 \end{array} \\ (\ +1,\ -1,\ 0,\ 0,\ 0\ ) \\ (\ +1,\ 0,\ -1,\ 0,\ +1\ ) \\ (\ +1,\ 0,\ 0,\ +1,\ +1\ ) \\ (\ 0,\ +1,\ -1,\ 0,\ +1\ ) \\ (\ 0,\ +1,\ 0,\ +1,\ +1\ ) \\ (\ 0,\ 0,\ +1,\ +1,\ 0\ ). \end{array}$$

Suppose that we wish to compute $\mathrm{BBY}_{(M,\sigma,\sigma^*)}([e_3], \{e_2, e_3, e_5\})$. First, we find the $\{0,1\}^E$ vector associated with $\{e_2, e_3, e_5\}$ with respect to the pair (σ, σ^*). To do this, we look at the fundamental circuit or cocircuit of each element. This gives the vector $(1, 0, 1, 0, 1)$. Next, we add $(0, 0, 1, 0, 0)$ and get $(1, 0, 2, 0, 1)$. Finally, we need to find which basis is associated with the equivalence class of $\mathcal{S}(M)$ containing $(1, 0, 2, 0, 1)$. Through trial and error (or more clever means), one can find that $[(1, 0, 2, 0, 1)] = [(1, 1, 0, 1, 1)]$, and this vector is associated with the basis $\{e_1, e_3, e_4\}$. It follows that

$$\mathrm{BBY}_{(M,\sigma,\sigma^*)}([e_3], \{e_2, e_3, e_5\}) = \{e_1, e_3, e_4\}.$$

6.2. Consistency conjectures

Let $M = (E, \mathcal{B}, \chi)$ be an oriented regular matroid and $e \in E$. Define $M \setminus e$ and M/e as given in [BLVS+99, Propositions 3.3.1 and 3.3.2]. It is well-known that $M \setminus e$ and M/e are also regular matroids ([Oxl06, Proposition 3.2.5]). Furthermore, as discussed in [BLVS+99, Section 3.3], for any pair (σ, σ^*) of acyclic signatures on M, there is a pair $(\sigma \setminus e, \sigma^* \setminus e)$ of induced acyclic signatures on $M \setminus e$ and a pair $(\sigma/e, \sigma^*/e)$ of induced acyclic signatures on M/e.

Notice that the representatives of equivalence classes of $\mathcal{S}(M)$ are generated by the elements of E. The following definition is a matroidal analogue to consistency (Definition 4.3).

[6] We obtain σ and σ^* by ordering the elements of E and then setting the minimal nonzero element of each circuit and cocircuit to be positive. This method will always produce a pair of acyclic signatures, but not all acyclic signatures are produced this way (see [Yue18, Example 5.1.4]).

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

Definition 6.9. A matroidal sandpile torsor algorithm α is *consistent* if for every regular matroid $M = (E, \mathcal{B}, \chi)$, pair of acyclic signatures (σ, σ^*), $B \in \mathcal{B}$, and $f \in E$, the following equalities hold (where $B' = \alpha_{(M,\sigma,\sigma^*)}([f], B)$):

1. For $e \in (B \cap B') \setminus f$, we have
$$\alpha_{(M/e,\sigma/e,\sigma^*/e)}([f], B \setminus e) = B' \setminus e.$$
2. For any $e \notin B \cup B' \cup f$, we have
$$\alpha_{(M\setminus e,\sigma\setminus e,\sigma^*\setminus e)}([f], B) = B'.$$

Remark 6.10. It is entirely possible that Definition 6.9 needs to be tweaked a bit in order for Conjectures 6.11 and 6.14 to be plausible. For example, it may be necessary (or convenient) to include an analogue of Condition 3 from Definition 4.3.

Conjecture 6.11. *The* BBY *matroidal sandpile torsor algorithm is consistent.*

For a circuit signature σ, let $\overline{\sigma}$ be the circuit signature made up of the signed circuits not in σ. Define $\overline{\sigma}^*$ similarly for cocircuit signatures.

Definition 6.12. Suppose α is a matroidal sandpile torsor algorithm. Define α', α'', and α''' such that for any regular matroid $M = (E, \mathcal{B}, \chi)$ and pair of acyclic signatures (σ, σ^*), we have
$$\alpha_{(M,\sigma,\sigma^*)} = \alpha'_{(M,\overline{\sigma},\sigma^*)} = \alpha''_{(M,\overline{\sigma},\overline{\sigma}^*)} = \alpha'''_{(M,\sigma,\overline{\sigma}^*)}.$$

Definition 6.13. Matroidal sandpile torsor algorithms α and β have the same *structure* if $\beta \in \{\alpha, \alpha', \alpha'', \alpha'''\}$.

Conjecture 6.14. *Every consistent matroidal sandpile torsor algorithm has the same structure as* BBY.

A. Sink-free rotor configurations and unicycles

Here, we introduce *sink-free rotor configurations* and *unicycles*, which were explored in both [HLM+08] and [CCG14]. These objects allow us to prove the remaining results from Section 3.

Definition A.1. A *sink-free rotor configuration* is an assignment of an incident edge to every vertex of G.

We obtain a sink-free rotor configuration from a rotor configuration by assigning a rotor to the sink vertex.

Definition A.2. A *unicycle* is a pair (ρ^*, x) where ρ^* is a sink-free rotor configuration with exactly one directed cycle and x is a vertex on this cycle.

If we are given a unicycle (ρ^*, x), we can get a new unicycle by replacing $\rho^*\langle x\rangle$ with $\chi(x, \rho^*\langle x\rangle)$ and then replacing x with the other vertex incident to the new value of $\rho^*\langle x\rangle$. In other words, we apply one step of Algorithm 1. By [HLM+08, Lemma 3.3], this will always output a new unicycle. This action of Algorithm 1 on unicycles is what Holroyd et al. call the *rotor-routing process*. For clarity, we will only use the word *process* when working with unicycles.

Lemma A.3 [HLM+08, Lemma 4.9]. *Let (ρ^*, x) be a unicycle on a ribbon graph with m edges. If we iterate the rotor-routing process $2|E(G)|$ times, the chip traverses each edge of G exactly once in each direction, each rotor makes exactly one full turn and the final unicycle is (ρ^*, x).*

Let (G, χ) be a ribbon graph, $T \in \mathcal{T}(G)$ and $c, s \in V(G)$. Suppose that there is some $f \in E(G)$ such that $\mathfrak{i}(f) = \{c, s\}$. Let T_s^{*f} be the sink-free rotor configuration defined by:
$$T_s^{*f}\langle x\rangle := \begin{cases} T_s\langle x\rangle & \text{if } x \neq s, \\ f & \text{if } x = s. \end{cases} \tag{5}$$

Lemma A.4. *The pair* (T_s^{*f}, c) *is a unicycle.*

Proof. The rotor configuration T_s contains a directed path from c to s and no cycles. After adding a rotor from s to c, we have a unique cycle which must contain c. □

This leads directly to a unicycle interpretation of the rotor-routing algorithm. Suppose that Algorithm 1 requires n iterations of the while loop to terminate given the inputs T and $c - s$. For each $k \in [0, n]$, let ρ_k be the rotor configuration after k iterations of the while loop in Algorithm 1. Similarly, let (ρ_k^*, c_k) be the unicycle obtained from (T^{*f}, c) after k steps of the rotor-routing process.

Lemma A.5. *For any* $k \in [0, n]$, *we have*

$$\rho_k^*\langle v\rangle = \begin{cases} \rho_k\langle v\rangle & \text{for } v \neq s \\ f & \text{for } v = s. \end{cases}$$

Proof. This follows immediately from the fact that, until the chip leaves s on the $(n+1)$ step of the rotor-routing process, the rotor at s remains fixed and does not affect the path of the chip or the rotation of the other rotors. □

Proof of Proposition 3.3. For each k, let (ρ_k^*, c_k) denote the kth step of the rotor-routing process initialized at (T^{*f}, c) and suppose that n is the smallest integer such that $c_n = s$. As a consequence of Lemma A.3, the chip must visit every vertex before the unicycle returns to its original value. Thus, this must happen after step n. Then by Lemma A.3, no rotor can have completed more than a full rotation by step n. By the equivalence established in Lemma A.5, this implies that no rotor can complete more than a full rotation before Algorithm 1 terminates. □

For a sink-free rotor configuration ρ^* with a unique directed cycle C, let $\overline{\rho^*}$ be the sink-free rotor configuration obtained by reversing the direction of the rotors that make up C and keeping the others fixed.

Proposition A.6 [CCG14, Proposition 9]. *The following are equivalent:*

- (G, χ) *is a plane graph.*
- *For every unicycle* (ρ^*, x) *on* (G, χ), *repeated applications of the rotor-routing process eventually produce the unicycle* $(\overline{\rho^*}, x)$.

Lemma A.7 [HLM+08, Corollary 4.11]. *Let* (G, χ) *be a plane graph and* (ρ^*, x) *be a unicycle. Suppose that the rotor-routing process is performed until we obtain the unicycle* $(\overline{\rho^*}, x)$. *In the process of rotor-routing, the chip crosses every edge to the left of* C *in both directions and no edges to the right of* C.

Remark A.8. The authors of [CCG14] and [HLM+08] use a clockwise convention, so their arguments relating to Proposition A.6 and Lemma A.7 switch left and right vertices. When G is restricted to the square lattice, Lemma A.7 was first proven in [PPS98].

We conclude with proofs of Lemmas 3.4 and 3.5.

Proof of Lemma 3.4. Let T_s^{*f} be the sink-free rotor configuration defined in (5) and let $\overline{T_s^{*f}}$ be the sink-free rotor configuration defined above Proposition A.6. Apply the rotor-routing process to (T_s^{*f}, c) until the chip reaches s. This must occur before we reach the unicycle $(\overline{T_s^{*f}}, c)$ because the rotor at s must move to reach $\overline{T_s^{*f}}$. By Lemma A.7, none of the edges to the right of C will be crossed. The proof follows from Lemma A.5 □

Proof of Lemma 3.5. For each $k \in \mathbb{Z}_{\geq 0}$, let (ρ_k^*, c_k) be the sink-free rotor configuration obtained after k steps of the rotor-routing process initialized at (T^{*f}, c). Let $m := |E(G)|$ and let C_k be the unique directed cycle in ρ_k^*. It is immediate that, for $k \in [0, n]$, the cycle C_k is also a directed cycle of ρ_k if and only if $s \notin V(C_k)$. Thus, the lemma follows if we can show that for any $i \in [0, n]$ such that $s \notin C_i$, there exists some $j \in [0, n]$, where $(\rho_j^*, c_j) = \overline{(\rho_i^*, c_i)}$.

By Proposition A.6 and Lemma A.3, there exists such a j in the interval $[0, 2m - 1]$. If $i > j$, then the result is trivially true, so we may assume without loss of generality that $i < j$. If s is to the right of C_i, then by Lemma A.7, the cycle will reverse before s is reached and the result follows. Alternatively, if s is to the left of C_i, then the unicycle (ρ_j^*, c_j) is not reached until after the chip enters s for the $\deg(s)$th time. We claim that $c_{2m-1} = s$. It follows from the claim that $j > 2m - 1$ because the chip only reaches s a total of $\deg(s)$ times after $2m$ steps of the rotor-routing process (by Lemma A.3). This is a contradiction and the result follows.

To prove the claim, we first note that there must be a directed edge from c_{2m-1} to $c_{2m} = c_0 = c$ in $\rho_{2m}^* = \rho_0^*$. If $c_{2m-1} \notin V(C_0)$, then C_0 is also a directed cycle in ρ_{2m-1}^*, and it follows that $C_0 = C_{2m-1}$. This is impossible because $c_{2m-1} \in V(C_{2m-1})$ by Definition A.2. The claim follows from the fact that s is the only vertex in $V(C_0)$ whose rotor is directed toward c in ρ_0^*. □

B. Case 4 of Theorem 5.4

Case 4 of Theorem 5.4 is a lot more complicated than the other cases because no edges are shared between T and $\alpha_{(G,\chi)}([c - s], T)$, so we cannot directly apply the definition of consistency. In this appendix, we prove the results necessary for this final case.

At a high level, our proof of Case 4 of Theorem 5.4 is as follows. We suppose there exists a counterexample: a non-rotor-routing consistent sandpile torsor algorithm α. In Section B.1, we use several lemmas to construct highly restrictive conditions that α must satisfy. Then in Section B.2, we show that the conditions established in Section B.1 give rise to a parameterized class of graphs that we call *telescope graphs*. Then if (G, χ) (as defined in the statement of case 4) is not a telescoping graph and $\alpha_{(G,\chi)} \neq r_{(G,\chi)}$, then α trivially violates the conditions established in Section B.1. We finish with a direct proof $\alpha_{(G,\chi)} = r_{(G,\chi)}$ if (G, χ) is a telescope graph.

B.1. Properties of a potential counterexample

Throughout Section B.1, assume G is a 2-connected plane graph such that $|E(G)| = 2|V(G)| - 2$. In other words, any spanning tree of G contains exactly half the edges of G. We additionally assume that α is a consistent sandpile torsor algorithm such that $\alpha_{(G',\chi')} = r_{(G',\chi')}$ for any proper minor (G', χ') of (G, χ).

Throughout this subsection, we will allow $(c - s, T)$ to denote *any* single-step pair or source-turn pair as needed by the argument. Note that this differs from the definition of $(c - s, T)$ given in the statement of Case 4 in the proof of Theorem 5.4.

The goal of this subsection is to establish Corollary B.7, which states that if $\alpha_{(G,\chi)}$ disagrees with $r_{(G,\chi)}$ on any single-step pair $(c - s, T)$, then for any T' such that $(c - s, T')$ is a single step pair, $\alpha_{(G,\chi)}([c - s], T') = E(G) \setminus T'$. A similar result holds for reverse single-step pairs. We do this in several steps:

Step 1: We first show that for any single-step pair $(c - s, T)$, if $\alpha_{(G,\chi)}([c - s], T) \neq r_{(G,\chi)}([c - s], T)$, then we must have $\alpha_{(G,\chi)}([c - s], T) = E(G) \setminus T$. An analogous result holds for reverse single-step pairs (see Lemmas B.1 and B.2).

Step 2: Given the conditions of Step 1, we show that $\alpha_{(G,\chi)}([c - s], T') = E(G) \setminus T'$ for any T' that can be reached from T via single-step moves and reverse single-step moves that do not affect the rotor at g (see Lemma B.4).

Step 3: Using similar arguments to those used to prove Theorem 5.8, we show that the set of trees listed in Step 2 includes all trees T' for which $(c - s, T')$ is a single-step pair or for which $(s - c, T')$ is a reverse single-step pair (see Lemma B.5).

Step 4: The corollary follows directly.

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

Lemma B.1. *Let $(c - s, T)$ be a single-step pair from g to f. Then,*

$$\alpha_{(G,\chi)}([c-s],T) \in \{T \setminus g \cup f, E(G) \setminus T\}.$$

Proof. Let $\widehat{T} = \alpha_{(G,\chi)}([c-s],T)$. Suppose that $\widehat{T} \neq T \setminus g \cup f$. Then, by the same logic we used in the proof of Theorem 5.4, we know that $V(G) \setminus (T\Delta\widehat{T}) \subseteq \{f,g\}$ (i.e., (2) holds for all single-step pairs, not just source-turn pairs). Furthermore, by the condition that $2|T| = |E(G)|$, and since all spanning trees are the same size, we must have

$$\widehat{T} \in \{E(G) \setminus T, (E(G) \setminus T) \setminus f \cup g\}.$$

This means that we just need to show that $\widehat{T} \neq (E(G) \setminus T) \setminus f \cup g$. This is true for precisely the reasoning we used in Case 2 of Theorem 5.4. □

Lemma B.2. *Let $(s - c, T)$ be a reverse single-step pair from f to g. Then,*

$$\alpha_{(G,\chi)}([s-c],T) \in \{T \setminus f \cup g, E(G) \setminus T\}.$$

Proof. Let $\widehat{T} = \alpha_{(G,\chi)}([s-c],T)$. Suppose that $\widehat{T} \neq T \setminus f \cup g$. Then, by the same logic we used in the proof of Theorem 5.4, we know that $V(G) \setminus (T\Delta\widehat{T}) \subseteq \{f,g\}$ (i.e., (2) also holds for reverse single-step pairs). Furthermore, by the condition that $2|T| = |E(G)|$, and since all spanning trees are the same size, we must have

$$\widehat{T} \in \{E(G) \setminus T, (E(G) \setminus T) \setminus g \cup f\}.$$

This means that we just need to show that $\widehat{T} \neq (E(G) \setminus T) \setminus g \cup f$. However, we cannot use the same reasoning we used in Case 2 of Theorem 5.4. Instead we give a new proof by contradiction.

First, suppose that $x = s$ (where $\mathfrak{i}(g) = \{x, c\}$) so that f and g are parallel. By the definition of reverse single-step pairs, $(c - s, T \setminus f \cup g)$ is a single-step pair from g to f. By Lemma B.1, we must have $\alpha_{(G,\chi)}([c-s], T \setminus f \cup g) \in \{T, (E(G) \setminus T) \setminus g \cup f\}$. In the first case, we have $\widehat{T} = T \setminus f \cup g$ and the lemma follows. Otherwise, $(E(G) \setminus T) \setminus g \cup f$ must be a tree, which means that $E(G) \setminus T$ is also a tree (since f and g are parallel). Furthermore, since $x = s$, it is immediate that $c \prec_{(E(G)\setminus T)_s} x$. By Lemma 5.11, this implies that $(c - s, E(G) \setminus T)$ is a single-step pair from g to f.

By Lemma B.1, we must have

$$\alpha_{(G,\chi)}([c-s], E(G) \setminus T) \in \{(E(G) \setminus T) \setminus g \cup f, T\}.$$

In the first case, it follows that

$$\alpha_{(G,\chi)}([c-s], E(G) \setminus T) = \alpha_{(G,\chi)}([c-s], T \setminus f \cup g),$$

which implies that $E(G) \setminus T = T \setminus f \cup g$. This is impossible unless $E(G) = \{f, g\}$, in which case the lemma is trivial. Thus, we must have $\alpha_{(G,\chi)}([c-s], E(G) \setminus T) = T$ and $\alpha_{(G,\chi)}([s-c], T) = E(G) \setminus T$.

Now we consider the case where $x \neq s$ (where once again, $\mathfrak{i}(g) = \{x, c\}$). Suppose for the sake of contradiction that $\widehat{T} = (E(G) \setminus T) \setminus g \cup f$. Because $g \notin T \cup \widehat{T}$, it follows by consistency that

$$\widehat{T} = \alpha_{(G\setminus g,\chi\setminus g)}([s-c],T) = r_{(G\setminus g,\chi\setminus g)}([s-c],T).$$

We can now find a contradiction by applying Algorithm 1 with the sink at c and the chip starting at s. Notice that $\widehat{T}_c\langle s\rangle = T_c\langle s\rangle = f$. Thus, the rotor at s must rotate completely around. Furthermore, by Proposition 3.3, the chip must enter s precisely $\deg(s) - 1$ times (where we subtract 1 because the chip starts at s). There is no rotor at c, so the chip can never pass through f to s. Thus, the chip must enter s along every other incident edge.

By Lemma 5.11 and the definition of reverse single-step pairs, we know that on G, $c \prec_{(T\setminus f\cup g)_s} x$. In particular, this means that there is a path of edges in T from x to s which does not pass through c. It follows that there is some $h \in T \setminus f$ that is incident to s.

By our earlier reasoning, the chip must cross h to s at some point during rotor-routing. It follows from Proposition 3.3 that this rotor must turn completely around. However, this means that h must be the final position of this rotor. Because $h \in T$, it follows from (2) that $h \notin (E(G) \setminus T) \setminus g \cup f$. Thus, we have a contradiction and $\widehat{T} \neq (E(G) \setminus T) \setminus g \cup f$. □

Lemma B.3. *Let $(s-c, T)$ be a reverse single-step pair from f to g (which implies that $(c-s, T \setminus f \cup g)$ is a single-step pair from g to f). Then,*

$$\alpha_{(G,\mathcal{X})}([s-c],T) = E(G) \setminus T \Longleftrightarrow \alpha_{(G,\mathcal{X})}([c-s], T \setminus f \cup g) = (E(G) \setminus T) \setminus g \cup f.$$

Proof. If $|E(G)| = 2$, the result is trivial. Otherwise, for the forward direction, we apply Lemma B.1. This lemma says that it suffices to show that $\alpha_{(G,\mathcal{X})}([c-s], T \setminus f \cup g) \neq T$. If this inequality does not hold, then we have

$$T \setminus f \cup g = \alpha_{(G,\mathcal{X})}([c-s+s-c], T \setminus f \cup g) = \alpha_{(G,\mathcal{X})}([s-c],T) = E(G) \setminus T.$$

This is a contradiction because $T \setminus f \cup g \neq E(G) \setminus T$ unless $|E(G)| = 2$.

The reverse direction is analogous after applying Lemma B.2. □

Lemma B.4. *Suppose that $(c-s,T)$ is a single-step pair such that $\alpha_{(G,\mathcal{X})}([c-s],T) = E(G) \setminus T$. Further, suppose that for some $c', s' \in V(G)$ and $h, k \in E(G) \setminus \{f, g\}$, the pair $(c'-s', T)$ is a single-step or reverse single-step pair from h to k. Then,*

$$\alpha_{(G,\mathcal{X})}([c-s], T \setminus h \cup k) = (E(G) \setminus T) \setminus k \cup h.$$

Proof. By Lemmas B.1 and B.2, we know that

$$\alpha_{(G,\mathcal{X})}([c'-s'],T) \in \{T \setminus h \cup k, E(G) \setminus T\}.$$

Furthermore, if $\alpha_{(G,\mathcal{X})}([c'-s'],T) = E(G) \setminus T$, then $[c'-s'] = [c-s]$. This is false because

$$r_{(G,\mathcal{X})}([c-s],T) = T \setminus g \cup f \neq T \setminus h \cup k = r_{(G,\mathcal{X})}([c'-s'],T).$$

It follows that $\alpha_{(G,\mathcal{X})}([c'-s'],T) = T \setminus h \cup k$.

By the condition that $T_s\langle c' \rangle = h$, and because $(c'-s',T)$ is a single-step pair, we know that $T_s\langle v \rangle = (T \setminus h \cup k)_s \langle v \rangle$ for all $v \in V(G) \setminus c'$. In particular, $T_s\langle c \rangle = (T \setminus h \cup k)_s\langle c \rangle = g$. Thus, by Lemma 5.11, $(c-s, T \setminus h \cup k)$ is a single-step pair from g to f. By an analogous argument, $(c'-s', T \setminus g \cup f)$ is a single-step pair from h to k.

Suppose that the result does not hold (i.e., $\alpha_{(G,\mathcal{X})}([c-s], T \setminus h \cup k) \neq (E(G) \setminus T) \setminus k \cup h$). By Lemma B.1, this means that $\alpha_{(G,\mathcal{X})}([c-s], T \setminus h \cup k) = \widetilde{T}$, where $\widetilde{T} := T \setminus \{h, g\} \cup \{k, f\}$. By Definition 5.19, we know that $(s'-c', \widetilde{T})$ is a reverse single-step pair from k to h. By Lemma B.2,

$$\alpha_{(G,\mathcal{X})}([s'-c'], \widetilde{T}) = \widetilde{T} \setminus k \cup h = T \setminus g \cup f \neq E(G) \setminus T$$
$$\text{or } \alpha_{(G,\mathcal{X})}([s'-c'], \widetilde{T}) = (E(G) \setminus T) \setminus \{k, f\} \cup \{h, g\} \neq E(G) \setminus T.$$

However, we also have

$$E(G) \setminus T = \alpha_{(G,\mathcal{X})}([c-s],T) = \alpha_{(G,\mathcal{X})}([c'-s'+c-s+s'-c'],T) =$$
$$\alpha_{(G,\mathcal{X})}([c-s+s'-c'], T \setminus h \cup k) = \alpha_{(G,\mathcal{X})}([s'-c'], \widetilde{T}).$$

This gives a contradiction and we have proven the result. □

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

The following lemma is a variant of Theorem 5.8 where one rotor must remain fixed, but we are allowed to use single-step and reverse single-step moves, not just source-turn moves. The proof is a bit technical but uses the same idea as the proof of Theorem 5.8.

Lemma B.5. *Let $T^{start}, T^{goal} \in \mathcal{T}(G)$ be any two spanning trees such that $T_s^{start}\langle c\rangle = T_s^{goal}\langle c\rangle = g$. There exists a sequence $\{(c_i - s_i, T^i)\}_{i\in[0,k]}$ such that $T^0 = T^{start}$, $T^k = T^{goal}$, and for every $i \in [1,k]$,*

- ○ *$(c_i - s_i, T^{i-1})$ is a single-step pair or reverse single-step pair;*
- ○ *$T^i = r_{(G,\chi)}([c_i - s_i], T^{i-1})$;*
- ○ *$T_s^i\langle c\rangle = g$; and*
- ○ *$c_i \neq s$.*

Before giving the proof, we give the following Lemma, which is a variation of Lemma 5.15 under which the contracted edges are not incident to the root.

Lemma B.6. *Let (G,χ) be a 2-connected plane graph with two spanning trees $T, T' \in \mathcal{T}(G)$ and some $\mathfrak{r} \in V(G)$. Fix a connected $F \subset T \cap T'$ which does not contain any edges incident to $\mathfrak{r}$. Let $(\widetilde{G}, \widetilde{\chi})$ be the ribbon graph obtained by contracting the edges in F, and z be the vertex these edges contract into.*

If there exists a sequence of source-rotations that take $(T \setminus F)_{\mathfrak{r}}$ to $(T' \setminus F)_{\mathfrak{r}}$ in $(\widetilde{G}, \widetilde{\chi})$ without turning the rotor at z, then there exists a sequence of source-rotations that take $T_{\mathfrak{r}}$ to $T'_{\mathfrak{r}}$ in (G,χ) without turning any rotors corresponding to edges in F.

Proof. The proof follows analogously to the proof of Lemma 5.15. The only difference is that a source-rotation at z of a rotor configuration $\widetilde{T}_{\mathfrak{r}}$ on $(\widetilde{G}, \widetilde{\chi})$ might not correspond to a source-rotation of $(\widetilde{T} \cup F)_{\mathfrak{r}}$ on (G,χ). This is not a problem by the condition that the rotor at z remains stationary. □

Proof of Lemma B.5. Recall the partial order $<_\xi$ from the proof of Theorem 5.8 setting $\mathfrak{r} = s$. By Lemma 5.7, all source-turn pairs are also single-step pairs. For simplicity, we will call a tree $T \in \mathcal{T}(G)$ *valid* if $T_s\langle c\rangle = g$. For the sake of induction, suppose that that for all 2-connected proper minors (G', χ') containing c and s (allowing minors where disjoint sets of vertices merge into c and s), the results of the lemma hold.

To prove this lemma, it suffices to prove that for every valid $T \in \mathcal{T}(G) \setminus T^{goal}$, there exists a valid $T'' \in \mathcal{T}(G)$ such that $T'' <_\xi T$, and it is possible to move from T_s to T''_s through a series of single-step and reverse single-step moves which never change the rotor at c.

Suppose there exists some $y \in V(G) \setminus s$ such that $\xi(T,y) = 1$ and y is a leaf vertex of T. Then, we can freely rotate the rotor at y until we get a tree T'' such that $T''\langle y\rangle = 0$. Then, $T'' <_\xi T$ and we are done.

Otherwise, let y be a minimal element with respect to $<_{T_s}$ such that $\xi(T,y) = 1$. By minimality, $\xi(T,v) = 0$ for all $v <_{T_s} y$. As in the proof of Theorem 5.8, we define

$$V^y := \{v \mid v <_{T_s} y\} \text{ and } E^y := \{T_s\langle v\rangle \mid v \in V^y\}.$$

By the same argument that we used in the proof of Theorem 5.8, $V^y \subseteq \{v \mid v <_{T_s^{goal}} y\}$. We will consider two cases (where $\mathfrak{i}(g) = \{x, c\}$).

Case 1: $x \nless_{T_s} y$.

Because $x \nless_{T_s} y$, it follows that $x \notin V^y$. Furthermore, if $c <_{T_s} y$, then the path from c to y along T must pass through g (and therefore x), which would imply that $x <_{T_s} y$. Thus, we must have $c \nless_{T_s} y$, which implies that $c \notin V^y$ and $g \notin E^y$. Then, by the claim stated in the proof of Theorem 5.8, there exists some $T' \in \mathcal{T}(G)$ that can be reached from a series of source-turn moves at vertices in V^y such that y is a leaf of T'. Since $c \notin V^y$, this implies it is possible to go from T to T' without moving the rotor at c. Since y is a leaf of T', we can rotate the rotor at y freely to get a spanning tree T'' such that $T''\langle y\rangle = T^{goal}\langle y\rangle$. By construction, $T'' <_\xi T$ and the lemma follows.

Case 2: $x \preceq_{T_s} y$.

Let P be the path of edges in T from c to y and let $V(P)$ be the vertices of this path. By definition of the partial order $\preceq_{T_s}$, we must have $g \in P$. Let $(G/P, \chi/P)$ be the plane graph formed by contracting the edges of P and let z be the contracted vertex. Since G is 2-connected, z is the only vertex of G/P that can be a cut vertex. Furthermore, s cannot be contracted into z because the vertices that make up P are all less than or equal to y with respect to $\preceq_{T_s}$. Let $(\widehat{G/P}, \widehat{\chi/P})$ be the graph we obtain after removing any edges and vertices that can only be reached from s through paths that pass through z. We do not remove z. In other words, $\widehat{G/P}$ is the maximal 2-connected subgraph of G/P which contains s.

Let $\widetilde{T}$ be the restriction of T to $E(\widehat{G/P})$. By 2-connectedness of $\widehat{G/P}$, there exists some $\widetilde{T}' \in \mathcal{T}(\widehat{G/P})$ such that z is a leaf vertex. Furthermore, by the (stronger) inductive claim we proved in Theorem 5.8 (see the inductive assumption with $\widehat{T}^{start} = \widetilde{T}$, $\widehat{T}^{goal} = \widetilde{T}'$ and $\mathfrak{r}' = z$), we know that $\widetilde{T}'$ can be reached from $\widetilde{T}$ by a series of source-turn moves at vertices that do not involve z. Let

$$T' := \widetilde{T}' \cup (T \cap (E(G) \setminus E(\widehat{G/P}))).$$

In other words, T' is equivalent to $\widetilde{T}'$ when restricted to $E(\widehat{G/P})$, and the remaining edges match the edges of T. By Lemma B.6, there exist a sequence of source-turn moves which send T to T' on (G, χ) and do not involve any rotors in P (including g).

Finally, we need to rotate the rotor at y to the correct position. Fix $h = T_s\langle y\rangle = T_s'\langle y\rangle$. Notice that y is not a leaf of T' because it is incident to h and an edge of P. Nevertheless, we claim that there exists $T'' \in \mathcal{T}(G)$ such that $\xi(T'', y) = 0$, and we can reach T'' from T' through a series of single-step or reverse single-step moves at y. Given this claim, it follows that $T'' \prec_\xi T$ because $\xi(T'', y) < \xi(T, y)$ and $\xi(T'', v) = \xi(T, v)$ for every $v \in E(G) \setminus (V^y \cup s)$.

To prove the claim, we first consider the set $\widetilde{V} = V(G) \setminus V(\widehat{G/P})$. By definition, any $v \in \widetilde{V}$ must satisfy $v \prec_{T_s} w$ for some $w \in V(P)$. Furthermore, for any $w \in V(P)$, we have $w \prec_{T_s} y$. It follows that any $v \in \widetilde{V}$ must satisfy $v \prec_{T_s} y$ and, by assumption, this means that $T_s\langle v\rangle = T_s^{goal}\langle v\rangle$.

Suppose that $T_s^{goal}\langle y\rangle \in E(G) \setminus E(\widehat{G/P})$. Then, by the results of the previous paragraph, we can follow rotors of T_s^{goal} until we return to y. This implies that T^{goal} contains a cycle, which is impossible. Thus, we must have $T_s^{goal}\langle y\rangle \in E(\widehat{G/P})$. Furthermore, by construction, h is the only edge incident to y in $\widetilde{T}'$. Thus, we can freely rotate the rotor at y in either direction using single-step or reverse single-step moves as long as we avoid edges in $E(G) \setminus E(\widehat{G/P})$. Our claim follows if we show that the edges of $\chi(y) \cap E(\widehat{G/P})$ must all appear sequentially in $\chi(y)$ on G.

Let $\{y, w\} = \mathfrak{i}(h)$ and let e be any edge of $E(\widehat{G/P}) \setminus h$ that is incident to y. By 2-connectedness of $\widehat{G/P}$, there is a path $\widehat{P}$ in G from y to w that contains e but does not contain h or any edges of P. Thus, $\widehat{P} \cup h$ forms a cycle C. Declare the 'inside' of C to be the region that does not contain c. By planarity, P must be completely outside of C. Since G is 2-connected, the edges inside of C must be a subset of $E(\widehat{G/P})$. This includes all of the edges between h and e with respect to $\chi(y)$. Since e was arbitrary, the claim follows. □

Corollary B.7. *Suppose that $(c - s, T)$ is a single-step pair from g to f such that $\alpha_{(G,\chi)}([c - s], T) = E(G) \setminus T$.*

- *For any $T' \in \mathcal{T}(G)$ such that $(c - s, T')$ is a single-step pair from g to f,*

$$\alpha_{(G,\chi)}([c - s], T') = E(G) \setminus T'.$$

- *For any $T'' \in \mathcal{T}(G)$ such that $(s - c, T'')$ is a reverse single-step pair from f to g,*

$$\alpha_{(G,\chi)}([s - c], T'') = E(G) \setminus T''.$$

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

Proof. For the first claim, apply Lemma B.5 with $T^{start} = T$ and $T^{goal} = T'$. This produces a sequence $\{(c_i - s_i, T^i)\}_{i\in[0,k]}$. By repeatedly applying Lemma B.4, we find that $\alpha_{(G,\chi)}([c-s], T^{goal}) = E(G) \setminus T^{goal}$ as desired.

For the second claim, by definition, $(c - s, T'' \setminus f \cup g)$ is a single-step pair from f to g. Thus, $\alpha_{(G,\chi)}([c - s], T'' \setminus f \cup g) = (E(G) \setminus T'') \setminus g \cup f$. The result follows from Lemma B.3. □

B.2. The counterexample does not exist

Throughout this section, (G, χ) is a 2-connected plane graph and α is a consistent sandpile torsor algorithm such that $\alpha_{(G',\chi')} = r_{(G',\chi')}$ for any proper minor (G', χ') of (G, χ). We also fix $c, s \in V(G)$ and $g, f \in E(G)$ such that for some $T \in \mathcal{T}(G)$,

- ○ the pair $(c - s, T)$ is a source-turn pair from g to f, and
- ○ $\alpha_{(G,\chi)}([c - s], T) = E(G) \setminus T$.

Furthermore, $x \in V(G)$ is defined such that $\mathfrak{i}(g) = \{c, x\}$ (where x and s are allowed to coincide). We also assume that each spanning tree of G contains half of the edges of G. Note that we use T as a variable for an arbitrary tree in this section. We do not fix T to be the specific tree satisfying the conditions of Case 4.

Our general strategy for proving Case 4 makes use of Corollary B.7 and the statement of Case 4 in the following manner:

Step 1: Corollary B.7 implies that for any single-step pair of the form $(c - s, T')$, we have $\alpha_{(G,\chi)}([c - s], T') = E(G) \setminus T'$. If there is any such T' where $E(G) \setminus T' \notin \mathcal{T}(G)$, then this contradicts the fact that $\alpha_{(G,\chi)}$ is a sandpile torsor action (see Lemma B.12).

Step 2: We define and characterize the class of *telescope graphs* which consists of all plane graphs in which the contradiction above does not appear (see Lemma B.11).

Step 3: We prove the result directly when restricted to telescope graphs (See Lemma B.13).

Definition B.8. We introduce a family of plane graphs called *telescope graphs*, which all satisfy the conditions at the beginning of the subsection. These graphs are indexed by $n \in \mathbb{Z}_{\geq 0}$ and a vector $(k_0, k_1, \ldots, k_n) \in \mathbb{Z}_{\geq 0}^{n+1}$. We begin with c, s, x, f and g as defined above. We then relabel s as z_n and x as z_0. Add the vertices $z_1, \ldots, z_{n-1}$ to the graph and then, for each $i \in [n]$, add two parallel edges between z_{i-1} and z_i. Finally, for each $i \in [0, n]$, we add k_i degree 2 vertices to the graph, which are incident to both c and z_i.

Up to ribbon graph isomorphism, there is a unique planar ribbon structure such that f comes directly after g in the cyclic order around c. We will write the graph along with this planar ribbon structure as $\mathcal{T}^{\backslash}_{(\|/,\ldots,\|\backslash)}$.

Figure 7 shows an example of a telescope graph. For each $i \in [0, n]$ where $k_i \neq 0$, we label the degree 2 vertices incident to z_i by $(w_i^1, w_i^2, \ldots, w_i^{k_i})$, ordered based on the ribbon structure at z_i. We label the edge between z_i and w_i^j by h_i^j and the edge between c and w_i^j by $\widehat{h}_i^j$. We also label the two edges between z_{i-1} and z_i by e_i and $\widehat{e}_i$, where e_i comes directly before $\widehat{e}_i$ in the cyclic order around z_{i-1}.

Definition B.9. Let (G, χ) be a plane graph satisfying the conditions discussed at the beginning of the subsection. We say that a spanning tree $T \in \mathcal{T}(G)$ is a *single-step tree* if $(c - s, T)$ is a single-step pair from g to f or $(s - c, T)$ is a reverse single-step pair from f to g.

Lemma B.10. *A spanning tree $T \in \mathcal{T}(G)$ is a single-step tree if and only if it contains exactly one of f and g as well as a path from x to s that does not pass through c.*

Proof. This follows immediately from Definition B.9 as well as Lemma 5.11. □

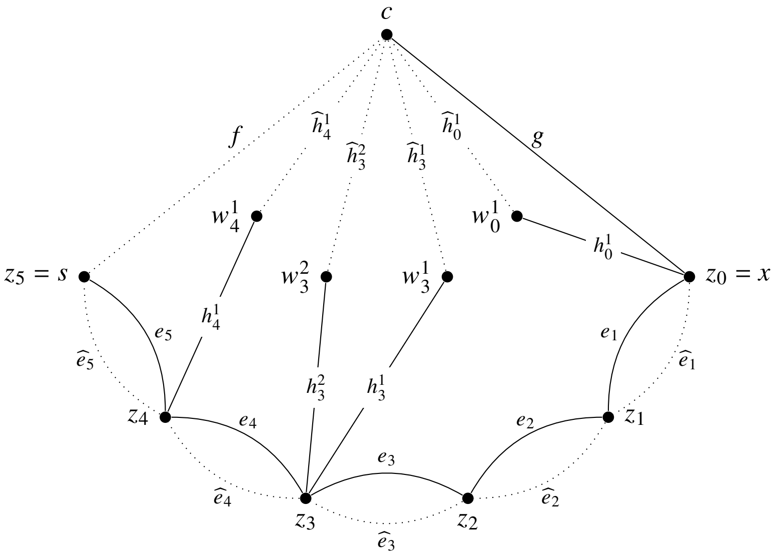

Figure 7. *Above is the telescope graph $\mathcal{T}^{\triangledown}_{(\infty,/,/,\in,\infty,/)}$ that is defined in Definition B.8. The solid lines indicate the spanning tree we consider in the proof of Lemma B.13.*

Lemma B.11. *The following are equivalent (when (G,χ) is a 2-connected plane graph satisfying the conditions from the beginning of the section):*

- ○ $(G,\chi) = \mathcal{T}^{\backslash}_{(\|/,\ldots,\|\backslash)}$ *for some $n \in \mathbb{Z}_{\geq 0}$ and $(k_0, k_1, \ldots, k_n) \in \mathbb{Z}^{n+1}_{\geq 0}$.*
- ○ *For every single-step tree $T \in \mathcal{T}(G)$, we have $E(G) \setminus T \in \mathcal{T}(G)$.*

Proof. The forward direction follows from direct argument. For the reverse direction, we apply induction on the number of edges in the graph.

For the forward direction, consider any single-step tree $T \in \mathcal{T}(G)$. For any $i \in [0,n]$ and $j \in [1,k_i]$, T must contain at least one of h_i^j and $\widehat{h}_i^j$ or else w_i^j is isolated. By Lemma B.10, T contains f or g and an path P from z_0 to z_n that does not pass through c. Notice that $P \cup \{h_i^j, \widehat{h}_i^j\} \cup f$ and $P \cup \{h_i^j, \widehat{h}_i^j\} \cup g$ both contain circuits. Thus, T must contain exactly one of the edges h_i^j and $\widehat{h}_i^j$. Similarly, for $i \in [1,n]$, the path P must contain exactly one of $\{e_i, \widehat{e}_i\}$. To prevent a circuit, T must also contain exactly one of every such pair. It follows that for T to be a single-step tree, it must contain exactly one of $\{f,g\}$, exactly one of each $\{h_i^j, \widehat{h}_i^j\}$ and exactly one of each $\{e_i, \widehat{e}_i\}$. It is straightforward to check that any such choice of edges will give a single-step tree. Furthermore, the complement of a single-step tree makes the opposite choice for each pair of edges listed above. Thus, $E(G) \setminus T$ is also a single-step tree and a spanning tree.

We will prove the reverse direction by induction. The only 2-edge plane graph satisfying the property is the double edge, which is equivalent to $\mathcal{T}'_{(/)}$. Suppose that every plane graph with $2N$ edges satisfying the property is a telescope graph. Let (G,χ) be a 2-connected plane graph with $2N+2$ edges such that for every single-step tree $T \in \mathcal{T}(G)$, we have $E(G) \setminus T \in \mathcal{T}(G)$. On any graph, the size of each spanning tree is $|V(G)| - 1$. For $E(G) \setminus T$ to ever be a spanning tree, we must have $2(|V(G)| - 1) = |E(G)|$ and $|V(G)| = |E(G)|/2 + 1 = N + 2$.

First, suppose that $x = s$ so that the edges f and g are parallel in G. For this case, we do not actually need to use induction. Let v be an arbitrary vertex other than x and s. Suppose that v is incident to a pair of parallel edges other than $\{f,g\}$. By 2-connectedness, removing these parallel edges does not disconnect the graph. Thus, there must be some tree T containing g and neither of these parallel edges. Since $x = s$, it follows immediately from Lemma B.10 that T is a single-step tree. We reach a contradiction because

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

$E(G) \setminus T$ contains a pair of parallel edges and cannot be a tree. Thus, $\{f, g\}$ is the only pair of parallel edges in G.

Furthermore, if there is some spanning tree T that contains g and all of the edges incident to v, then v is isolated in $E(G) \setminus T$ and this set cannot form a spanning tree. Thus, any spanning tree of G containing g must only contain a subset of the edges incident to v. Because every subtree of G is the subset of a spanning tree, this implies that the union of g and the edges incident to v must contain a circuit. If $v = c$, then this circuit is simply the double edge $\{f, g\}$. If $v \neq c$, then the fact that $i(g) = \{c, s\}$ implies that there must be an edge incident to v and c and an edge incident to v and s. We showed earlier that if there are N vertices other than c and s, then there must be $2N + 2$ edges. These edges are accounted for by f, g and the two specified edges incident to each of the N vertices $v \notin \{c, s\}$. We already accounted for all of these edges, so (G, χ) must be equal to $\mathcal{T}'_{(\mathcal{N})}$.

Now, we consider the case where $s \neq x$. By 2-connectedness, there must be some path P along G from x to s that does not pass through c. Then, the union of P with f is also acyclic, so it can be expanded to a tree T, which must not contain g. It follows from Lemma B.10 that T is a single-step tree, which means that $E(G) \setminus T$ must be a spanning tree by assumption. However, $E(G) \setminus T$ contains all edges incident to both c and x. Thus, there cannot be any edges parallel to g.

Consider the set of all edges incident to x. If these edges contain no circuits, then we can take their union with P and then add edges to obtain a spanning tree T. It follows from Lemma B.10 that T is a single-step tree. However, $E(G) \setminus T$ cannot be a spanning tree because x is isolated. This is a contradiction, so the edges incident to x must contain a circuit. In particular, this must be a double edge $\{e, \widehat{e}\}$. By our reasoning in the previous paragraph, these edges must connect x to a vertex $v \neq c$.

By assumption, any single-step tree T must contain exactly one of e and $\widehat{e}$. Otherwise, T or $E(G) \setminus T$ contains a cycle. Furthermore, it follows from Lemma B.10 that T is a single-step tree on (G, χ) if and only if $T \setminus \{e, \widehat{e}\}$ is a single-step tree on $(G/e \setminus \widehat{e}, \chi/e \setminus \widehat{e})$. This implies that for every single-step tree $T \in (G/e \setminus \widehat{e}, \chi/e \setminus \widehat{e})$, we must have $E(G) \setminus T \in \mathcal{T}(G/e \setminus \widehat{e})$. By the induction hypothesis, this means that $(G/e \setminus \widehat{e}, \chi/e \setminus \widehat{e})$ must be a telescope graph.

We have shown that after contracting a double edge incident to x, we obtain a telescope graph. Let z be the vertex formed by contraction. To recover G, we need to determine which edges incident to z in $G/e \setminus \widehat{e}$ are incident to x in G and which are incident to v. If $v = s$, then it is immediate that (G, χ) is a telescope graph with $n = 1$. Otherwise, $G/e \setminus \widehat{e}$ must contain a pair of parallel edges e' and $\widehat{e}'$ which are incident to z. By definition, (G, χ) is a telescope graph if and only if neither e' nor $\widehat{e}'$ are incident to x on (G, χ).

Suppose for the sake of contradiction that e' is incident to x on G. Then, e' is part of a path P from x to s that does not pass through c. In particular, $f \notin P$ and $g \cup P$ contains no cycles. Thus, $P \cup g$ is contained in a spanning tree that does not contain f, e or $\widehat{e}$. This spanning tree is a single-step tree, but the complement contains the cycle $\{e, \widehat{e}\}$.

The argument from the previous paragraph still holds if we replace e' with $\widehat{e}'$. Thus, (G, χ) must be a telescope graph. □

Lemma B.12. *If (G, χ) is not a telescope graph, then for any $T \in \mathcal{T}(G)$ such that $(c - s, T)$ is a single-step pair from g to f, we have $\alpha_{(G,\chi)}([c - s], T) = T \setminus g \cup f$.*

Proof. Suppose that this condition does not hold. By Lemma B.1, this means that $\alpha_{(G,\chi)}([c - s], T) = E(G) \setminus T$. By Lemma B.11, there must be some single-step tree T' such that $E(G) \setminus T'$ is not a spanning tree. We arrive at a contradiction from Corollary B.7 and the definition of single-step trees because the output of a sandpile torsor action must always be a spanning tree. □

Lemma B.13. *If (G, χ) is a telescope graph, then for any $T \in \mathcal{T}(G)$ such that $(c - s, T)$ is a single-step pair from g to f, we have $\alpha_{(G,\chi)}([c - s], T) = T \setminus g \cup f$.*

Proof. Suppose for the sake of contradiction that the lemma does not hold. Then, by Lemma B.1, there exists some $T \in \mathcal{T}(G)$ such that $(c - s, T)$ is a single-step pair from g to f and $\alpha_{(G,\chi)}([c - s], T) = E(G) \setminus T$. By Corollary B.7, this equality holds for any single-step tree.

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press

To obtain a contradiction, we explicitly construct a divisor D_Σ such that $\alpha_{(G,\chi)}([D_\Sigma],T) = \alpha_{(G,\chi)}([c-s],T) = E(G)\setminus T$, but $[D_\Sigma] \neq [c-s]$.

Case 1: $n > 0$.

Let $\mathcal{T}^{\backslash}_{(\|_{\prime},\ldots,\|_{\backslash})} = (G,\chi)$ and let T' be the spanning tree

$$T' := g \cup \{e_i \mid i \in [1,n]\} \cup \{h_i^j \mid i \in [0,n], j \in [1,k_i]\}.$$

See Figure 7 for this tree on $\mathcal{T}^{\bigtriangledown}_{(\infty,\prime,\prime,\in,\infty,\prime)}$.

We consider several sequences of divisors (where we ignore any divisors involving w_i^j for $k_i = 0$ or z_i for $i < 0$):

$$\begin{aligned}
\mathcal{E}_0 &= (w_0^1 - c, \ldots, w_0^{k_0} - c, w_1^1 - c, \ldots, w_1^{k_1} - c, \ldots, w_n^{k_n} - c)\\
\mathcal{E}_1 &= (z_n - w_n^1, \ldots, z_n - w_n^{k_n}, z_n - c)\\
\mathcal{E}_2 &= (z_{n-1} - w_{n-1}^1, \ldots, z_{n-1} - w_{n-1}^{k_{n-1}}, z_{n-1} - z_n, z_{n-1} - z_n)\\
\mathcal{E}_3 &= (z_{n-2} - w_{n-2}^1, \ldots, z_{n-2} - w_{n-2}^{k_{n-2}}, z_{n-2} - z_{n-1}, z_{n-2} - z_{n-1})\\
\ldots &= \ldots\\
\mathcal{E}_{n+1} &= (z_0 - w_0^1, \ldots, z_0 - w_0^{k_0}, z_0 - z_1, z_0 - z_1)\\
\mathcal{E} &= (\mathcal{E}_0, \mathcal{E}_1, \ldots, \mathcal{E}_{n+1}).
\end{aligned}$$

Let D_l be the l^{th} entry in $\mathcal{E}$. Let $T^0 = T'$ and recursively define the sequence

$$(T^1, T^2, T^3, \ldots, T^{|\mathcal{E}|}) \in \mathcal{T}(G)^{|\mathcal{E}|}$$

such that $r_{(G,\chi)}([D_l], T^{l-1}) = T^l$. It is straightforward to see that $T^{|\mathcal{E}|} = E(G)\setminus T'$, and for any $l \in [1,|\mathcal{E}|]$, the pair (D_l, T^{l-1}) is a single-step pair. We claim that for all $l \in [1,|\mathcal{E}|]$, we also have $\alpha_{(G,\chi)}([D_l],T^{l-1}) = T^l$.

For any $l \in [1,|\mathcal{E}|]$, let $c', s' \in V(G)$ such that $D_l = c' - s'$. Notice by the definition of $\mathcal{E}$ that $c' \neq c$. By Lemma B.12, $\alpha_{(G,\chi)}([D_l],T^{l-1}) = r_{(G,\chi)}([D_l],T^{l-1}) = T^l$ unless (G,χ) remains a telescope graph when c' plays the role of c and s' plays the role of s (i.e., when there is a ribbon graph automorphism which maps c to c' and s to s'). It is easy to verify that when $n > 0$, the only ribbon graph automorphism is trivial.

Let D_Σ be the sum of all of the divisors that make up $\mathcal{E}$. We have shown that

$$\alpha_{(G,\chi)}([D_\Sigma],T') = r_{(G,\chi)}([D_\Sigma],T') = T^{|\mathcal{E}|} = E(G)\setminus T'.$$

If $\alpha_{(G\chi)}([c-s],T') = E(G)\setminus T'$, then this implies that $[D_\Sigma] = [c-s]$. This is false because

$$r_{(G,\chi)}([D_\Sigma],T') = E(G)\setminus T' \neq T'\setminus g \cup f = r_{(G,\chi)}([c-s],T').$$

Case 2: $n = 0$.

If $k_0 = 0$, then $E(G) = \{f,g\}$ and the proof is trivial. Otherwise, let $T' = g \cup \{h_0^j \mid j \in [1,k_i]\}$ and $\mathcal{E} = (w_0^1 - c, w_0^2 - c, \ldots, w_0^{k_0} - c, s - w_0^1, s - w_0^2, \ldots, s - w_0^{k_0}, s - c)$. Define D_l, T^l and D_Σ as in Case 1, but with this new choice of $\mathcal{E}$.

The argument that we used in Case 1 is mostly still valid, except that there does exist a nontrivial ribbon graph automorphism on (G,χ). Nevertheless, the only such automorphism must flip c and s. Thus, the only additional equality that we need to check is $\alpha_{(G,\chi)}([s-c],T^{2k_0}) = T^{2k_0+1}$. However, T^{2k_0} contains neither g nor f, so $E(G)\setminus T^{2k_0}$ is not a spanning tree. By Lemma B.1, we know that

$\alpha_{(G,\chi)}([s-c], T^{2k_0}) = r_{(G,\chi)}([s-c], T^{2k_0}) = T^{2k_0+1}$. The remainder of the proof follows using the same reasoning from Case 1. □

Acknowledgements. We would like to thank Matt Baker, Alex Fink, Richard Kenyon, Caroline Klivans, Cyrus Peterpaul, Lilla Tóthmérész, Chi Ho Yuen and the anonymous referees for helpful conversation, comments and suggestions. In particular, we are grateful to Caroline Klivans for calling our attention to the main conjecture, Richard Kenyon for suggesting we look for a consistency condition related to contraction and deletion and Lilla Tóthmérész for engaging with an early version of our work and discovering Example 4.7.

Competing interest. The authors have no competing interest to declare.

Funding statement. The first author was partially supported by the United States Army Research Office under grant number W911NF2010133.

References

[BBY19] S. Backman, M. Baker and C. H. Yuen, ‘Geometric bijections for regular matroids, zonotopes, and Ehrhart theory’, *Forum Math. Sigma* **7** (2017).

[Ber08] O. Bernardi, ‘Tutte polynomial, subgraphs, orientations and sandpile model: new connections via embeddings’, *Electron. J. Combin.* **15**(R109) (2008).

[Big99] N. L. Biggs, ‘Chip-firing and the critical group of a graph’, *J. Algebraic Combin.* **9**(1) (1999), 25–45.

[BLHN97] R. Bacher, P. de La Harpe and T. Nagnibeda, ‘The lattice of integral flows and the lattice of integral cuts on a finite graph’, *Bull. Soc. Math. France* **125**(2) (1997) 167–198.

[BLVS+99] A. Björner, M. Las Vergnas, B. Sturmfels, N. White and G. M. Ziegler, *Oriented Matroids*, vol. 46 (Cambridge University Press, 1999).

[BW17] M. Baker and Y. Wang, ‘The Bernardi process and torsor structures on spanning trees’, Int. Math. Res. Not. *IMRN* (2017).

[CCG14] M. Chan, T. Church and J. A. Grochow, ‘Rotor-routing and spanning trees on planar graphs’, *Int. Math. Res. Not. IMRN* 2015(**11**), 2014.

[CP18] S. Corry and D. Perkinson, *Divisors and Sandpiles*, vol. 114 (American Mathematical Society, 2018).

[Dha90] D. Dhar, ‘Self-organized critical state of sandpile automaton models’, *Phys. Rev. Lett.* **64**(14) (1990), 1613.

[Din21] C. Ding, ‘The rotor-routing torsor and the bernardi torsor disagree for every non-planar ribbon graph’, Preprint, 2021, arXiv:2103.01137.

[Ell11] J. Ellenberg, ‘What is the sandpile torsor?’ MathOverflow, 2011.

[HLM+08] A. E. Holroyd, L. Levine, K. Mészáros, Y. Peres, J. Propp and D. B. Wilson, ‘Chip-firing and rotor-routing on directed graphs’, in *In and Out of Equilibrium 2* (Springer, 2008), 331–364.

[Kli18] C. Klivans, *The Mathematics of Chip Firing* (Chapman & Hall, 2018).

[KLT22] T. Kálmán, S. Lee and L. Tóthmérész, ‘The sandpile group of a trinity and a canonical definition for the planar bernardi action’, *Combinatorica* (2022), 1–34.

[LL09] I. Landau and L. Levine, ‘The rotor–router model on regular trees’, *J. Combin. Theory Ser. A* **116**(2) (2009), 421–433.

[McD21a] A. McDonough, ‘Determining genus from sandpile torsor algorithms’, *Discrete Math. Theor. Comput. Sci.* **23**(1) (2021).

[McD21b] A. McDonough, ‘Higher-dimensional sandpile groups and matrix-tree multijections’, Ph.D. thesis, Brown University, 2021.

[Mer99] C. Merino, ‘Matroids, the Tutte polynomial and the chip firing game’, Ph.D. thesis, University of Oxford, 1999.

[Oxl06] J. G. Oxley, *Matroid Theory*, vol. 3 (Oxford University Press, 2006).

[PDDK96] V. B. Priezzhev, D. Dhar, A. Dhar and S. Krishnamurthy, ‘Eulerian walkers as a model of self-organized criticality’, *Phys. Rev. Lett.* **77**(25) (1996), 5079.

[PPS98] A. M. Povolotsky, V. B. Priezzhev and R. R. Shcherbakov, ‘Dynamics of Eulerian walkers’, *Phys. Rev. E* **58**, (1998), 5449–5454.

[SW21] F. Shokrieh and C. Wright, ‘Torsor structures on spanning trees’, Preprint, 2021, arXiv:2103.10370.

[Tót23] L. Tóthmérész, ‘Consistency of the planar rotor-routing action via the trinity definition’, (2023).

[Wag00] D. G. Wagner, ‘The critical group of a directed graph’, Preprint, 2000, arXiv:math/0010241.

[Yue17] C. H. Yuen, ‘Geometric bijections between spanning trees and break divisors’, *J. Combin. Theory Ser. A* **152** (2017), 159–189.

[Yue18] C. H. Yuen, ‘Geometric bijections of graphs and regular matroids’, Ph.D. thesis, Georgia Tech, 2018.

https://doi.org/10.1017/fms.2023.77 Published online by Cambridge University Press