Combinatorics, Probability and Computing (2011) 20, 11–25. © Cambridge University Press 2010
doi:10.1017/S0963548310000192

A Bijective Proof of a Theorem of Knuth

HODA BIDKHORI and SHAUNAK KISHORE

Massachusetts Institute of Technology, Massachusetts Avenue, Cambridge, MA 02139, USA
(e-mail: bidkhori@mit.edu, skishore@mit.edu)

Received 15 November 2009; revised 30 March 2010; first published online 21 July 2010

The line graph $\mathcal{L}G$ of a directed graph G has a vertex for every edge of G and an edge for every path of length 2 in G. In 1967, Knuth used the Matrix Tree Theorem to prove a formula for the number of spanning trees of $\mathcal{L}G$, and he asked for a bijective proof [6]. In this paper, we give a bijective proof of Knuth's formula. As a result of this proof, we find a bijection between binary de Bruijn sequences of degree n and binary sequences of length 2^{n-1}. Finally, we determine the critical groups of all the Kautz graphs and de Bruijn graphs, generalizing a result of Levine [7].

1. Introduction

In a directed graph $G = (V, E)$, each edge $e \in E$ is directed from its *source* $s(e)$ to its *target* $t(e)$. We denote the indegree and outdegree of a vertex v by $\text{indeg}(v)$ and $\text{outdeg}(v)$, respectively.

Definition 1 (Directed line graph). Let $G = (V, E)$ be a directed graph. The *directed line graph* $\mathcal{L}G$ is a directed graph with vertex set E, and with an edge (e, f) for every pair of edges e and f of G with $t(e) = s(f)$.

At times we may speak of a subset F of E as a subgraph of G: in this case we mean the subgraph (V, F). If H is a subgraph of G and v is in H, we denote the indegree of v in H by $\text{indeg}_H(v)$, and the outdegree by $\text{outdeg}_H(v)$.

Definition 2 (Oriented spanning tree). Let $G = (V, E)$ be a directed graph. An *oriented spanning tree* of G is an acyclic subgraph of G with a distinguished node, the *root*, in which there is a unique path from every vertex $v \in V$ to the root. We refer to these trees as *spanning trees*.

Let T be a spanning tree of G. Every vertex of G has outdegree 1 in T, except the root, which has outdegree 0. We denote the number of spanning trees of G by $\kappa(G)$, and the number of spanning trees rooted at r by $\kappa(G, r)$. Suppose that every vertex of G has indegree greater

https://doi.org/10.1017/S0963548310000192 Published online by Cambridge University Press

than 0. With this notation, Knuth's formula [7] for the number of spanning trees of $\mathcal{L}G$ states that

$$\kappa(\mathcal{L}G, e) = \frac{\kappa(G, s(e))}{\text{outdeg}(t(e))} \prod_{v \in V} \text{outdeg}(v)^{\text{indeg}(v)-1}. \tag{1.1}$$

Knuth's proof relied on the Matrix Tree Theorem. In his paper, he noted that the simple form of this result suggested that a bijective proof was possible, but that it was not at all obvious how to find such a bijection [6].

In fact, there are even stronger relations between $\kappa(\mathcal{L}G)$ and $\kappa(G)$. Let $\{x_v | v \in V\}$ and $\{x_e | e \in E\}$ be variables indexed by the vertices and edges of G. The vertex and edge generating functions of G are defined as follows, where the sums are taken over all vertices v and all spanning trees T of G rooted at v:

$$\kappa^{\text{edge}}(G) = \sum_T \prod_{e \in T} x_e, \quad \kappa^{\text{vertex}}(G) = \sum_T \prod_{e \in T} x_{t(e)}. \tag{1.2}$$

Levine used linear algebraic methods to prove the following generalization of Knuth's result. Our first result in this paper is a bijective proof of Levine's theorem, which yields a bijective proof of Knuth's theorem as a special case.

Theorem 1.1. *Let $G = (V, E)$ be a directed graph in which every vertex has indegree greater than 0. Then:*

$$\kappa^{\text{vertex}}(\mathcal{L}G) = \kappa^{\text{edge}}(G) \prod_{v \in V} \left(\sum_{s(e)=v} x_e \right)^{\text{indeg}(v)-1}. \tag{1.3}$$

Using this bijection, we are able to answer the following open question posed by Stanley in [8].

Exercise 5.73 from [8]. Let $\mathcal{B}(n)$ be the set of binary de Bruijn sequences of degree n, and let $\mathcal{S}(n)$ be the set of all binary sequences of length 2^n. Find an explicit bijection $\mathcal{B}(n) \times \mathcal{B}(n) \to \mathcal{S}(n)$.

The critical group $K(G)$ of a graph G is a finite abelian group whose order is the number of spanning trees of G. Critical groups have applications in statistical physics [4], algebraic combinatorics [7], and arithmetic geometry [1]. We review the definition of this group in Section 2.

The Kautz graphs $\text{Kautz}_n(m)$ and the de Bruijn graphs $\text{DB}_n(m)$ are families of iterated line graphs. $\text{Kautz}_1(m)$ is the complete directed graph on $m+1$ vertices, without self-loops, and $\text{DB}_1(m)$ is the complete graph on m vertices, with self-loops. These families are defined for $n > 1$ as follows:

$$\text{Kautz}_n(m) = \mathcal{L}^{n-1} \text{Kautz}_1(m), \quad \text{DB}_n(m) = \mathcal{L}^{n-1} \text{DB}_1(m). \tag{1.4}$$

Levine recently determined $K(\text{DB}_n(2))$ and $K(\text{Kautz}_n(m))$, where m is prime [7]. We generalize these results, proving the following characterizations of the critical groups of all the Kautz and de Bruijn graphs.

https://doi.org/10.1017/S0963548310000192 Published online by Cambridge University Press

Theorem 1.2. *The critical group of* $\mathrm{DB}_n(m)$ *is*

$$K(\mathrm{DB}_n(m)) = \left(\mathbb{Z}_{m^n}\right)^{m-2} \oplus \bigoplus_{i=1}^{n-1} \left(\mathbb{Z}_{m^i}\right)^{m^{n-1-i}(m-1)^2}. \tag{1.5}$$

Theorem 1.3. *The critical group of* $\mathrm{Kautz}_n(m)$ *is*

$$K(\mathrm{Kautz}_n(m)) = (\mathbb{Z}_{m+1})^{m-1} \oplus \left(\mathbb{Z}_{m^{n-1}}\right)^{m^2-2} \oplus \bigoplus_{i=1}^{n-2} \left(\mathbb{Z}_{m^i}\right)^{m^{n-2-i}(m-1)^2(m+1)}. \tag{1.6}$$

The rest of this paper is organized as follows. In Section 2 we define the critical group and provide further background. In Section 3, we introduce a bijection which proves Theorem 1.1. We apply this bijection in Section 4 to construct a bijection betweeen binary de Bruijn sequences of order n and binary sequences of length 2^{n-1}. Finally, in Section 5, we prove Theorems 1.2 and 1.3, giving a complete description of the critical groups of the Kautz and de Bruijn graphs.

2. Background and definitions

A directed graph $G = (V, E)$ is *strongly connected* if there is a path from v to w in G for every pair of vertices $v, w \in V$. Let $G = (V, E)$ be a strongly connected directed graph, and let $\mathbb{Z}^V$ be the free abelian group generated by vertices of G, *i.e.*, the group of formal linear combinations of vertices of G with integer coefficients. We define $\Delta_v \in \mathbb{Z}^V$, for all $v \in V$, as follows:

$$\Delta_v = \sum_{e \in E \text{ s.t. } s(e) = v} (t(e) - v). \tag{2.1}$$

For any vertex $r \in V$, let $H(r)$ be the subgroup of $\mathbb{Z}^V$ generated by r and by $\{\Delta_v | v \in V \backslash r\}$. The *sandpile group* $K(G, r)$ with *sink* r is the quotient group

$$K(G, r) = \mathbb{Z}^V / H(r). \tag{2.2}$$

It is well known that the order of $K(G, r)$ is $\kappa(G, r)$ [7].

A directed graph G is *Eulerian* if $\mathrm{indeg}(v) = \mathrm{outdeg}(v)$ for all vertices v in V. According to Lemma 4.12 of [4], if G is Eulerian, the sandpile groups $K(G, r_1)$ and $K(G, r_2)$ are isomorphic for any two r_1, r_2 in V. In this case, we call the group the *critical group* $K(G)$.

One of our main techniques for finding the critical group of a graph is row-reduction of the graph's Laplacian matrix.

Definition 3 (The Laplacian). Let $G = (V, E)$ be a finite directed graph, and let its vertices be $v_1, v_2, \ldots, v_{|V|}$. The *adjacency matrix* $A(G)$ of G is the $|V| \times |V|$ matrix in which $A(G)_{ij}$ is the multiplicity of the edge (v_i, v_j) in E. The *degree matrix* $D(G)$ is the $|V| \times |V|$ diagonal matrix in which $D_{ii} = \mathrm{outdeg}(v_i)$. The *Laplacian* $L(G)$ of G is defined as $A(G) - D(G)$.

The following elementary row and column operations on matrices with entries in a ring R are invertible over R.

https://doi.org/10.1017/S0963548310000192 Published online by Cambridge University Press

- Permuting two rows (columns).
- Adding a multiple of a row (column) by an element of R to another row (column).
- Multiplying the entries of a row (column) by a unit.

If L' is obtained from $L(G)$ by invertible row and column operations over $\mathbb{Z}$, then $\mathbb{Z}^V/\text{Im } L' \cong \mathbb{Z}^V/\text{Im } L(G)$.

Suppose that R is a principal ideal domain. Under these operations, any matrix with entries in R is equivalent to a matrix in *Smith normal form*. A matrix in this form is diagonal, and its diagonal entries $x_{11}, x_{22}, \ldots, x_{nn}$ are elements of R such that $x_{(i+1)(i+1)}$ is a multiple of x_{ii} for all $i < n$. These entries are called the *invariant factors* of the original matrix, and they are unique up to multiplication by units [3]. If the invariant factors of $L(G)$ over $\mathbb{Z}$ are $x_{11}, x_{22}, \ldots, x_{nn}$, then

$$\mathbb{Z}^V/\text{Im } L(G) = \bigoplus_{i=1}^{n} \mathbb{Z}_{x_{ii}}. \tag{2.3}$$

Note that the row vectors of $L(G)$ are the elements Δ_v. We consider $L(G)^T$ as a $\mathbb{Z}$-linear operator on $\mathbb{Z}^V$: its image is the subgroup generated by the Δ_v. For a strongly connected Eulerian graph G, the Laplacian has exactly one eigenvalue 0, so for such a graph G, we have

$$\mathbb{Z}^V/\text{im } L(G) \cong \bigoplus_{i=1}^{n} \mathbb{Z}_{x_{ii}} \cong \mathbb{Z}^V/\text{im } L(G)^T \cong \mathbb{Z} \oplus K(G), \tag{2.4}$$

where the x_{ii} are the invariant factors of $L(G)$. Thus, row-reducing the Laplacian yields information about the critical group.

3. Counting spanning trees

Let $G = (V, E)$ be a directed graph, and let $\{x_e\}_{e \in E}$ be variables indexed by the edges of G. In this section, we give a bijective proof of Theorem 1.1. We will show that if every vertex of G has indegree greater than 0, then

$$\sum_{T'} \prod_{(e,f) \in T'} x_f = \left(\sum_{T} \prod_{e \in T} x_e \right) \prod_{v \in V} \left(\sum_{s(e)=v} x_e \right)^{\text{indeg}(v)-1}, \tag{3.1}$$

where the sums are taken over every spanning tree T' of $\mathcal{L}G$ and every spanning tree T of G. In order to find a bijection, we adopt the following strategy.

- We give a bijection between monomial terms on the right-hand side of (3.1) and *tree arrays*, which are arrays of lists, one list for each vertex $v \in V$.
- Then we present a map σ that takes a tree array to a spanning tree of $\mathcal{L}G$ which contributes the same term to the left-hand side of (3.1).
- Finally, we show that σ is bijective by constructing an inverse map π which takes a spanning tree of $\mathcal{L}G$ to a tree array.

We define a *list* to be an ordered tuple of edges. We *append* an element x to a list l by adding x to the end of l. We *pop* list l by removing the first element of l. We denote the number of times an element e appears in a list l by $N(l, e)$.

https://doi.org/10.1017/S0963548310000192 Published online by Cambridge University Press

Definition 4 (Tree array.). Let $G = (V, E)$ be a directed graph. A *tree array* of G is an array of lists $\langle l_v \rangle$, with one list for each vertex $v \in V$, which satisfies the following properties.

(a) The length of l_v is $\text{indeg}(v)$, for all $v \in V$.
(b) Every element of l_v is an edge with source v, for all vertices v except $t(r')$. The last element of $l_{t(r')}$ is the variable Ω, and every other element of $l_{t(r')}$ is an edge with source $t(r')$.
(c) The set T of edges which are the last elements of the lists $\{l_v | v \in V \backslash t(r')\}$ is a spanning tree of G.

Let v be a vertex of G and let l'_v be a list with $\text{indeg}(v) - 1$ elements, all of which are edges with source v. We map l'_v to a monomial term of $(\sum_{s(e)=v} x_e)^{\text{indeg}(v)-1}$, as follows:

$$l'_v = (e_1, e_2, \ldots, e_{\text{indeg}(v)-1}) \rightarrow x_{e_1} x_{e_2} \cdots x_{e_{\text{indeg}(v)-1}}. \tag{3.2}$$

This map provides a bijection between lists l'_v and terms of $(\sum_{s(e)=v} x_e)^{\text{indeg}(v)-1}$. Therefore, a term on the right-hand side of (3.1) corresponds to a choice of spanning tree T of G and a choice of one such list l'_v for each vertex v.

Suppose a monomial term on the right-hand side of (3.1) corresponds to a spanning tree T rooted at r and an array of lists $\langle l'_v \rangle$. For each vertex $v \in V \backslash r$, we obtain l_v by appending the unique edge e in T with source v to the list l'_v. We obtain l_r by appending the variable Ω to l'_r.

By construction, an array of lists $\langle l_v \rangle_{v \in V}$ obtained in this way is always a tree array. Given a tree array, we can reverse this construction: we use the last elements of the lists l_v to form a spanning tree of G, and we use the first $\text{indeg}(v) - 1$ elements of each l_v to choose a monomial term of $(\sum_{s(e)=v} x_e)^{\text{indeg}(v)-1}$. Therefore, terms on the right-hand side of (3.1) are in bijection with tree arrays.

We now fix an arbitrary total order on the edges in E, which we will use to construct the maps σ and π. We define the bijective map σ, which takes a tree array of G to a spanning tree of $\mathcal{L}G$, as follows.

The bijection σ. We start with a tree array $\langle l_v \rangle$ and an empty subgraph T' of $\mathcal{L}G$. Then we run the following algorithm.

Step 1. Let R be the subset of edges e of G for which $N(l_{s(e)}, e) = 0$ and $\text{outdeg}_{T'}(e) = 0$. Let f be the smallest edge in R under the order on E.
Step 2. Pop the first element g from the list $l_{t(f)}$. If g is Ω, then $\sigma(\langle l_v \rangle) = T'$.
Step 3. Otherwise, $g \in E$ and $s(g) = t(f)$. Add the edge (f, g) to T', and then return to step 1.

We also define a map π which takes a spanning tree of $\mathcal{L}G$ to a tree array of G.

The inverse map π. We start with a spanning tree T' of $\mathcal{L}G$, and an empty list l_v at each vertex $v \in V$. This map is given by another algorithm.

Step 1. Let S be the set of leaves of T'. Let f be the smallest edge in S under the order on E.
Step 2. If f is not the root of T', remove f and its outedge (f, g) from T', and append g to $l_{t(f)}$. Go back to step 2.
Step 3. If f is the root of T', append Ω to $l_{t(f)}$, and return the array of lists.

As an example, we apply σ to a tree array in a small directed graph G. We order the edges of G by the lexicographic order.

https://doi.org/10.1017/S0963548310000192 Published online by Cambridge University Press

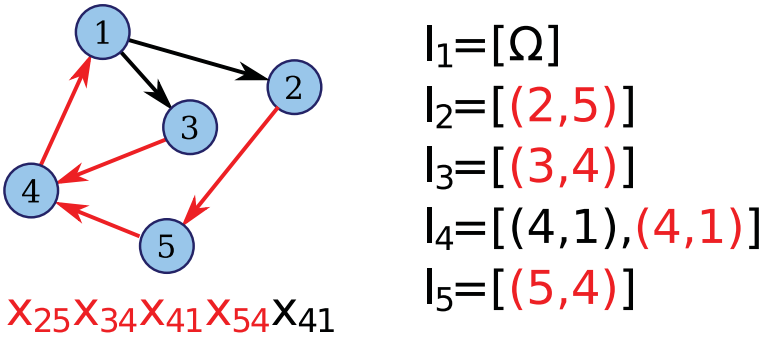


Figure 1. (Colour online) A spanning tree T highlighted in the tinted areas. Below the graph is a monomial term of $\kappa^{\text{vertex}}(G)$, and to the right is the associated tree array.

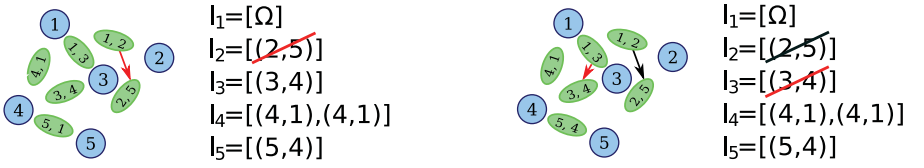


Figure 2. (Colour online) The algorithm for σ first pops $(2,5)$ from l_2 and adds the edge $((1,2),(2,5))$ to T'. Then it pops $(3,4)$ from l_3 and adds $((1,3),(3,4))$ to T'.

In order to prove Theorem 1.1, we first prove three lemmas. In the definition of the algorithm for the map σ, we assumed that the set R is always non-empty at step 1 and that the the list $l_{t(f)}$ is always non-empty at step 2. In Lemma 3.1, we show that both assumptions are valid.

Lemma 3.1. *The algorithm used to define map σ is well-defined: at step 1, the set R is non-empty, and at step 2, the list $l_{t(f)}$ is non-empty.*

Proof. After k edges have been added to T', there are $|E|-k$ elements left in all the lists l_v, where one of the elements is Ω. There are $|E|-k-1$ edges left in the lists, but there are $|E|-k$ edges of G which do not have an outedge in T', so R must be non-empty at step 1.

Every time we pop l_v, we add an edge (f,g) to T', where $t(f)=v$. When we are at step 2, $\text{outdeg}_{T'}(f)=0$, so at most $\text{indeg}(t(f))-1$ of the elements of $l_{t(f)}$ have been popped. Therefore, the list $l_{t(f)}$ is always non-empty at step 2. The algorithm is well-defined. □

The following lemma shows that σ takes a tree array corresponding to a term on the right-hand side of (3.1) to a spanning tree which contributes the same term to the left-hand side.

Lemma 3.2. *Suppose that $\langle l_v\rangle$ is a tree array and that $\sigma(\langle l_v\rangle)=T'$. Then T' is a spanning tree of $\mathcal{L}G$, and* $\text{indeg}_{T'}(e)=N(l_{s(e)},e)$, *for all $e\in E$.*

Proof. Let $I(e)$ be the initial value of $N(l_{s(e)},e)$. By the definition of a tree array, the edges which are the last elements of the lists l_v form a spanning tree T of G.

https://doi.org/10.1017/S0963548310000192 Published online by Cambridge University Press

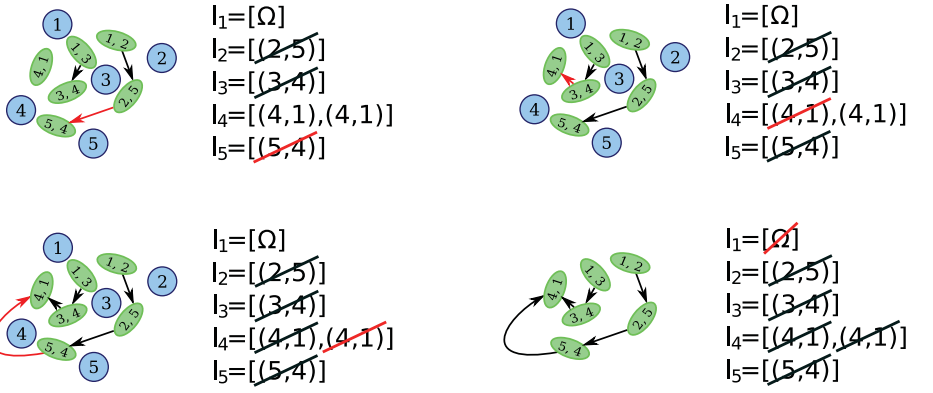


Figure 3. (Colour online) The last three edges added to T', and the final tree. The last element left in the lists of the tree array is Ω.

We claim that T' is acyclic, because the last edge of a cycle is never included in T'. While the algorithm is running, suppose that (e_n, e_1) is not an edge of T', and that it completes a cycle $(e_1, e_2), (e_2, e_3), \ldots, (e_{n-1}, e_n)$ of edges in T'. Since (e_1, e_2) was already added to T', $N(l_{s(e_1)}, e_1)$ must be 0. Therefore, (e_n, e_1) will never be added to T'.

We say a vertex $v \in V$ is *cleared* if all the elements of its list are popped. Suppose that $e = (v, w)$ is an edge in T. The list l_w is cleared when all the edges of G with target w have an outedge in T'. Then w can only be cleared after an outedge (e, f) of e is added to T. The edge (e, f) can only be added to T' when $N(l_v, e) = 0$. Because e is an edge of T, it is the last element of l_v, so v must be cleared before w can be cleared.

The algorithm terminates when Ω is popped from l_r, which occurs when r is cleared. There is a path $(v, v_1, v_2, \ldots, v_k, r)$ in T from any vertex v to r. Therefore r can only be cleared after all the vertices on this path are cleared. Thus, all the vertices of G are cleared when the algorithm finishes, so there are $|E| - 1$ edges in the subgraph T'.

All the vertices of $\mathcal{L}G$ have an outedge in T', except one. Since T' is acyclic, it is a spanning tree of $\mathcal{L}G$. Because $\text{indeg}_{T'}(e) + N(l_{s(e)}, e)$ is constant, when the algorithm returns T', $\text{indeg}_{T'}(e) = I(e)$ for all $e \in E$. □

In our final lemma, we show that π will take a spanning tree T' of $\mathcal{L}G$ and reconstruct a tree array $\langle l_v \rangle$.

Lemma 3.3. *Suppose T' is a spanning tree of $\mathcal{L}G$ with root r', and that π takes T' to the array of lists $\langle l_v \rangle$. Then $\langle l_v \rangle$ is a tree array.*

Proof. We first show that $\langle l_v \rangle$ satisfies parts (a) and (b) of Definition 4. Each time an edge $e \in E$ is removed from T', an element is appended to the list $l_{t(e)}$. Since r' can only be removed after all the other vertices of T', this algorithm adds $\text{indeg}(v)$ elements to l_v for all $v \in V$, so part (a) holds. Every element of the list l_v is an edge with source v, with the exception of Ω, which is the last element of l_r, so part (b) holds.

While the algorithm π is running, say a vertex $v \in V$ is *filled* if l_v has $\text{indeg}(v)$ elements. Every vertex is eventually filled, so the order in which vertices are filled is a total order on V.

https://doi.org/10.1017/S0963548310000192 Published online by Cambridge University Press

We claim that this order is a topological sort of the subgraph T, *i.e.*, for every edge $e \in E$, $s(e)$ is filled before $t(e)$. Suppose that $f = (v, w)$ is the last element of l_v for some vertex v other than r. Vertex v was filled at step 2, right after some leaf e and some edge (e, f) were removed from T' at step 1. However, w cannot be filled until f is removed from T', which happens after e and (e, f) are removed from T'. Therefore v is filled before w, so the filling ordering on V is a topological sort of T, and T is acyclic.

After π terminates, $t(r')$ has no outedge in T and every other vertex of G has one outedge. Then T is a spanning tree of G, and part (c) of Definition 4 holds. □

We now prove the main result.

Proof of Theorem 1.1. Let $\langle l_v \rangle$ be a tree array and let $T' = \sigma(\langle l_v \rangle)$. We show the following claim by induction on n: after n edges have been added to T' by the algorithm for $\sigma(\langle l_v \rangle)$, and n edges have been removed from T' by the algorithm for $\pi(T')$, we have the following.

(a) The set of edges added to T' by σ is the set of edges removed by π.
(b) The elements popped from l_v by σ are exactly the elements added to l_v by π, in the same order.

In the base case $n = 0$, both claims hold trivially. Suppose both results hold for $n = k$. The edge e is a leaf of T' in π if and only if it satisfies $N(l_{s(e)}, e) = 0$ and $\mathrm{outdeg}_{T'}(e) = 0$ in σ.

Therefore, the $(k+1)$st edge (f, g) added to T' by σ is also the $(k+1)$st edge removed from T' by π, and the element g popped from $l_{t(f)}$ in σ is also the element appended to $l_{t(f)}$ by π. Both claims hold for $n = k + 1$. By induction, they hold for all $n \leqslant |E| - 1$.

When $n = |E| - 1$, condition (b) implies that $\pi(T') = \langle l_v \rangle$. Then π is a left inverse of σ, and σ is injective. By similar reasoning, π is a right inverse of σ, and σ is surjective. So σ is a bijection between tree arrays in G and spanning trees of $\mathcal{L}G$. The bijection σ induces a bijection between equal terms on the left- and right-hand sides of (3.1), which proves Theorem 1.1. □

4. The de Bruijn bijection

A *binary de Bruijn sequence of degree n* is a cyclic binary sequence B such that every binary sequence of length n appears as a subsequence of consecutive elements of B exactly once. For example, 0011 is a binary de Bruijn sequence of degree 2, since its cyclic subsequences of length 2 are 00, 01, 11, and 10.

It is well known that there are $2^{2^{n-1}}$ binary de Bruijn sequences of degree n. Stanley posed the following open problem in [8].

Exercise 5.73 of [8]. Let $\mathcal{B}(n)$ be the set of binary de Bruijn sequences of degree n, and let $\mathcal{S}_n$ be the set of all binary sequences of length 2^n. Find an explicit bijection $\mathcal{B}(n) \times \mathcal{B}(n) \to \mathcal{S}_n$.

Our solution to this problem involves the de Bruijn graphs, which are closely related to de Bruijn sequences.

Definition 5 (de Bruijn graph). The *de Bruijn graph* $\mathrm{DB}_n(m)$ has m^n vertices, which are identified with the strings of length n on m symbols. The edges of the graph are labelled with

https://doi.org/10.1017/S0963548310000192 Published online by Cambridge University Press

the strings of length $n+1$ on m symbols. The edge $s_0 s_1 \cdots s_n$ has source $s_0 s_1 \cdots s_{n-1}$ and target $s_1 s_2 \cdots s_n$.

An edge of $\mathrm{DB}_n(m)$ can be identified with the vertex of $\mathrm{DB}_{n+1}(m)$ that is labelled with the same string of length $n+1$. With this identification, we have

$$\mathrm{DB}_n(m) = \mathcal{L}\,\mathrm{DB}_{n-1}(m). \tag{4.1}$$

Each vertex $v = s_0 s_1 \cdots s_{n-1}$ of $DB_n(2)$ has two outedges, $s_1 s_2 \cdots s_{n-1}0$ and $s_1 s_2 \cdots s_{n-1}1$. We call these edges the *zero edge* of v and the *one edge* of v, respectively.

It is well known that binary de Bruijn sequences of degree n are in bijection with Hamiltonian paths in $DB_n(2)$. Let $B = b_0 b_1 \cdots b_{2^n-1}$ be a binary de Bruijn sequence of degree n. Let $v_i = b_i b_{i+1} \cdots b_{i+n-1}$, for $0 \leqslant i \leqslant 2^n - 1$, where indices are taken mod 2^n. The path $(v_0, v_1, \ldots, v_{2^n-1})$ is the corresponding Hamiltonian path in $DB_n(2)$.

Theorem 4.1. *There is an explicit bijection between $\mathcal{B}(n)$ and the set of binary sequences of length 2^{n-1}, for $n > 1$.*

Proof. We describe a bijection between Hamiltonian paths in $\mathrm{DB}_n(2)$ and binary sequences of length 2^{n-1}. By composing this bijection with the map between de Bruijn sequences and Hamiltonian paths, we construct the desired bijection.

We order the vertices in $DB_k(2)$ by the lexicographic order on their associated binary strings, for $1 \leqslant k \leqslant n$. Let $(v_1, \ldots, v_{2^n})$ be a Hamiltonian path in $DB_n(2)$. This path is an oriented spanning tree of $DB_n(2)$, so we can apply the inverse map π defined in Section 3 to it.

Let A_{n-1} be the tree array $A_{n-1} = \pi(v_1, \ldots, v_{2^n})$. We recursively define a sequence of tree arrays A_k, for $1 \leqslant k \leqslant n-1$. Suppose we have a tree array A_{k+1} in $DB_{k+1}(2)$. Let T_{k+1} be the spanning tree consisting of the edges which are the last elements of the lists in A_{k+1}. We define A_k to be $\pi(T_{k+1})$.

We construct a binary sequence $s_1 s_2 \cdots s_{2^{n-1}}$ from these tree arrays. We denote vertex w's list in the tree array A_k by $(A_k)_w$. Let $s_{2^{n-1}}$ be 0 if the first element of $(A_{n-1})_{t(v_{2^n})}$ is the zero edge of $t(v_{2^n})$, and 1 otherwise.

We define s_{2^k} through $s_{2^{k+1}-1}$, for $1 \leqslant k \leqslant n-2$, as follows. Let $w_1, w_2, \ldots, w_{2^k}$ be the vertices of $\mathrm{DB}_k(2)$, in lexicographic order. Let s_{2^k+i-1} be 0 if the first element of $(A_k)_{w_i}$ is the zero edge of w_i, and 1 otherwise. Let s_1 be 0 if T_1 is rooted at vertex 0, and 1 otherwise.

The string $s_1 s_2 \cdots s_{2^{n-1}}$ is the binary sequence that corresponds to the Hamiltonian path we began with.

Now we construct the inverse map, from binary sequences to Hamiltonian paths. Given any binary sequence S of length 2^{n-1}, we use the first $2^{n-1} - 1$ characters of the sequence to invert the previous procedure and construct a sequence of spanning trees $T_1, T_2, \ldots, T_{n-1}$. The tree T_k will be a spanning tree of $DB_k(2)$.

We determine T_k recursively. The tree T_1 in $DB_1(2)$ is rooted at 0 if s_1 is 0, and rooted at 1 otherwise. Assume that the first $2^k - 1$ characters of S determine a spanning tree T_k of $DB_k(2)$, where $k \leqslant n-2$. We choose a tree array A_k of $DB_k(2)$ using this tree and the next 2^k characters of S, as follows.

https://doi.org/10.1017/S0963548310000192 Published online by Cambridge University Press

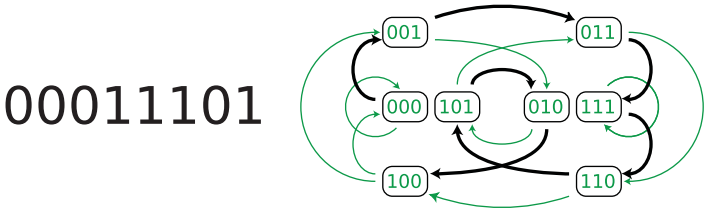


Figure 4. (Colour online) A de Bruijn sequence and the associated Hamiltonian path in $DB_3(2)$.

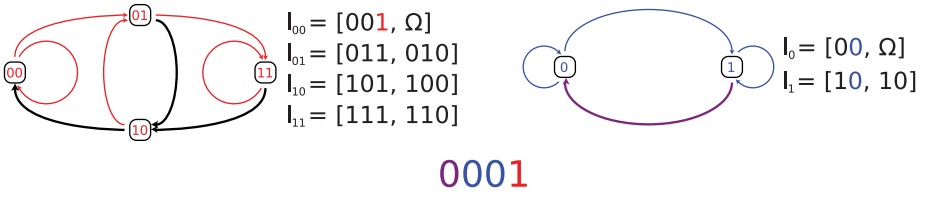


Figure 5. (Colour online) The bijection applied to the sequence above. The trees and tree arrays constructed by the bijection are shown. The first element of the final string is 0 because tree T_1 is rooted at 0. The rest of the string is defined by the tree arrays.

Let the vertices of $DB_k(2)$ be $w_1, w_2, \ldots, w_{2^k}$, in lexicographic order. The first element of $(A_k)_{w_i}$ is the zero edge of w_i if s_{2^k+i-1} is 0, and the one edge of w_i otherwise. The second element of $(A_k)_{w_i}$ comes from T_k. We define T_{k+1} to be $\sigma(A_k)$, using the map defined in Section 3.

We use T_{n-1} to construct a tree array A_{n-1} such that $\sigma(A_{n-1})$ is a Hamiltonian path in $DB_n(2)$. Let r be the root of T_{n-1}, and let v be another arbitrary vertex. The list l_v in the array A_{n-1} must contain two distinct edges, if $\sigma(A_{n-1})$ is a Hamiltonian path. The second edge in l_v must be the unique edge in T_{n-1} with source v, so l_v is determined. Our only remaining choice is which of the two edges of $DB_{n-1}(2)$ with source r to include in l_r, which we determine by $s_{2^{n-1}}$.

Clearly, this map from binary sequences to Hamiltonian paths inverts the map from Hamiltonian paths to binary sequences. Therefore, our first map is the bijection we need. □

This bijection can easily be generalized to count the k-ary de Bruijn sequences, in which the 2-symbol alphabet $\{0, 1\}$ is replaced with the k-symbol alphabet $\{0, 1, \ldots, k-1\}$.

5. The Kautz and de Bruijn graphs

In this section, we determine the critical groups of all the Kautz graphs and the de Bruijn graphs. The critical groups of these graphs have been found in some special cases by Levine [7].

The Kautz graphs are similar to the de Bruijn graphs, except that the vertices are indexed by *Kautz strings*. A Kautz string is a string in which no two adjacent characters are the same.

Definition 6 (Kautz graph). The *Kautz graph* $\mathrm{Kautz}_n(m)$ has $(m+1)m^{n-1}$ vertices, identified with the Kautz strings of length n on $m+1$ symbols. The edges of the graph are labelled with the Kautz strings of length $n+1$ on $m+1$ symbols, such that the edge $s_0 s_1 \cdots s_n$ has source $s_0 s_1 \cdots s_{n-1}$ and target $s_1 s_2 \cdots s_n$.

https://doi.org/10.1017/S0963548310000192 Published online by Cambridge University Press

We also consider the Kautz and de Bruijn graphs as families of iterated line graphs. $\mathrm{Kautz}_1(m)$ is the complete directed graph on $m+1$ vertices, without self-loops, and $\mathrm{DB}_1(m)$ is the complete directed graph on m vertices, with self-loops. Then, for $n > 1$, we have

$$\mathrm{Kautz}_{n+1}(m) = \mathcal{L}\,\mathrm{Kautz}_n(m) = \mathcal{L}^n\,\mathrm{Kautz}_1(m), \tag{5.1}$$

$$\mathrm{DB}_{n+1}(m) = \mathcal{L}\,\mathrm{DB}_n(m) = \mathcal{L}^n\,\mathrm{DB}_1(m). \tag{5.2}$$

We say a directed graph $G = (V, E)$ is *balanced k-regular* if $\mathrm{indeg}(v) = \mathrm{outdeg}(v) = k$ for all $v \in V$. Both $\mathrm{Kautz}_n(m)$ and $\mathrm{DB}_n(m)$ are balanced m-regular, for all $n \in \mathbb{N}$, which implies that they are Eulerian. Since these graphs are also strongly connected, their critical groups are defined.

Levine found the critical groups of the de Bruijn graphs $\mathrm{DB}_n(2)$ and the Kautz graphs $K_n(p)$, where p is prime [7]. In this section we characterize the critical groups of all the Kautz and de Bruijn graphs. We prove the following theorems.

Theorem 1.2. *The critical group of* $\mathrm{DB}_n(m)$ *is*

$$K(\mathrm{DB}_n(m)) = (\mathbb{Z}_{m^n})^{m-2} \oplus \bigoplus_{i=1}^{n-1} (\mathbb{Z}_{m^i})^{m^{n-1-i}(m-1)^2}. \tag{5.3}$$

Theorem 1.3. *The critical group of* $\mathrm{Kautz}_n(m)$ *is*

$$K(\mathrm{Kautz}_n(m)) = (\mathbb{Z}_{m+1})^{m-1} \oplus (\mathbb{Z}_{m^{n-1}})^{m^2-2} \oplus \bigoplus_{i=1}^{n-2} (\mathbb{Z}_{m^i})^{m^{n-2-i}(m-1)^2(m+1)}. \tag{5.4}$$

In order to prove these theorems, we first prove two lemmas about row-reducing the Laplacians $L(\mathrm{Kautz}_n(m))$ and $L(\mathrm{DB}_n(m))$. We refer to the row and column of a vertex v in the Laplacian by $R(v)$ and $C(v)$, respectively. We also use $L(v, w)$ to denote the entry in the row of v and the column of w.

We say two strings of length n are *similar* if their last $n-1$ characters are equal. Similarity is an equivalence relation. We partition the vertices of $\mathrm{Kautz}_n(m)$ and $\mathrm{DB}_n(m)$ into equivalence classes, by grouping vertices labelled with similar strings in the same class. There are m vertices in each class.

Lemma 5.1. *Let $G = (V, E)$ be a Kautz graph $\mathrm{Kautz}_{n+1}(m)$ or a de Bruijn graph $\mathrm{DB}_{n+1}(m)$, where $n \in \mathbb{N}$. Then G contains a cycle $(v_1, v_2, \ldots, v_c)$ of length $c = |V|/m$ which contains one vertex from each class.*

Proof. Let G', the predecessor of G, be $\mathrm{Kautz}_n(m)$ if G is $\mathrm{Kautz}_{n+1}(m)$, and $\mathrm{DB}_n(m)$ if G is $\mathrm{DB}_{n+1}(m)$.

First we show that there is a Hamiltonian cycle in G'. Such a cycle exists in the complete graphs K_m and K_{m+1}, so the case $n = 1$ is done. There is an Eulerian tour of $\mathrm{Kautz}_{n-1}(m)$ and of $\mathrm{DB}_{n-1}(m)$ for $n > 1$, since graphs in both families are Eulerian. Because G' is either $\mathcal{L}\,\mathrm{Kautz}_{n-1}(m)$ or $\mathcal{L}\,\mathrm{DB}_{n-1}(m)$, one of these Eulerian tours induces a Hamiltonian cycle in G', for $n > 1$.

https://doi.org/10.1017/S0963548310000192 Published online by Cambridge University Press

$$
\begin{array}{c} \\ 01 \\ 02 \\ 10 \\ 12 \\ 20 \\ 21 \end{array}
\begin{array}{c}
\begin{array}{cccccc} 01 & 02 & 10 & 12 & 20 & 21 \end{array} \\
\begin{pmatrix} -2 & 0 & 1 & 1 & 0 & 0 \\ 0 & -2 & 0 & 0 & 1 & 1 \\ 1 & 1 & -2 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2 & 1 & 1 \\ 1 & 1 & 0 & 0 & -2 & 0 \\ 0 & 0 & 1 & 1 & 0 & -2 \end{pmatrix}
\end{array}
\rightarrow
\begin{array}{c} \\ 01 \\ 02 \\ 10 \\ 12 \\ 20 \\ 21 \end{array}
\begin{array}{c}
\begin{array}{cccccc} 01 & 02 & 10 & 12 & 20 & 21 \end{array} \\
\begin{pmatrix} -2 & 0 & 1 & 1 & 0 & 0 \\ 0 & -2 & 0 & 2 & 0 & 0 \\ 0 & 0 & -2 & 0 & 2 & 0 \\ 0 & 0 & 0 & -2 & 1 & 1 \\ 1 & 1 & 0 & 0 & -2 & 0 \\ 2 & 0 & 0 & 0 & 0 & -2 \end{pmatrix}
\end{array}
$$

Figure 6. Reducing $L(\mathrm{Kautz}_2(2))$ using the cycle $(01, 12, 20)$. The original Laplacian is on the left. We obtain the reduced Laplacian on the right by subtracting $R(01)$ from $R(21)$, $R(12)$ from $R(02)$, and $R(20)$ from $R(10)$. Every entry of rows $R(02)$, $R(10)$, and $R(21)$ is divisible by 2.

The Hamiltonian cycle in G' can be represented as a string $S = s_1 s_2 \cdots s_{n+c-1}$, where the ith vertex of the cycle is labelled with $s_i s_{i+1} \cdots s_{i+n-1}$.

We use string S to find a cycle in G. Let $v_i = s_i s_{i+1} \cdots s_{i+n}$ for $i < c$, and let $v_c = s_c s_{c+1} \cdots s_{n+c-1} s_1$. By the construction of S, $(v_1, v_2, \ldots, v_c)$ is a cycle which contains one vertex from each class. □

In the next lemma we show that every invariant factor of $L(\mathrm{Kautz}_{n+1}(m))$ and $L(\mathrm{DB}_n(m))$ is either a multiple of m or relatively prime to m. We prove this lemma by row-reducing the Laplacian in an order derived from the cycle in Lemma 5.1.

Lemma 5.2. *Let $G = (V, E)$ be a Kautz graph $\mathrm{Kautz}_{n+1}(m)$ or a de Bruijn graph $\mathrm{DB}_{n+1}(m)$, where $n \in \mathbb{N}$. The first $c = |V|/m$ invariant factors of $L(G)$ are relatively prime to m, and all of the rest are divisible by m.*

Proof. We reduce the Laplacian $L(G)$ over the principal ideal domain $\mathbb{Z}_m$. Let the invariant factors of $L(G)$ over $\mathbb{Z}$ be $x_1, x_2, \ldots, x_{|V|}$. Any invertible row or column operation over $\mathbb{Z}$ descends to an invertible operation over $\mathbb{Z}_m$, so the invariant factors of $L(G)$ over $\mathbb{Z}_m$ are the x_i mod m.

Let $(v_1, v_2, \ldots, v_c)$ be the cycle in G from Lemma 5.1, and let $[v_i]$ be the set of vertices in the class of v_i. We take indices mod c, so v_{c+1} is v_1.

Note that if u and v are vertices in the same class, then (u, w) is an edge if and only if (v, w) is, for all w. Therefore, the rows of u and v in the adjacency matrix $A(G)$ are the same.

Because every vertex of G has outdegree m, $L(G) \equiv A(G)$ mod m. Therefore rows $R(u)$ and $R(v)$ are congruent mod m if u and v are in the same class. We reduce the Laplacian in c stages. In the ith stage, we subtract row $R(v_i)$ from $R(v)$ for all $v \in [v_i] \backslash v_i$. After this operation, every entry of $R(v)$ is divisible by m.

The entry $L(v_i, v_{i+1})$ is 1 before and after these row operations, for $1 \leqslant i \leqslant c$. We claim that in the reduced Laplacian, every entry of $C(v_{i+1})$ is divisible by m except $L(v_i, v_{i+1})$. There are m edges with target v_{i+1} in G. The sources of these edges are the m vertices in $[v_i]$. After the row operations, every entry of $R(v)$ is divisible by m for $v \in [v_i] \backslash v_i$, so $L(v_i, v_{i+1})$ is the only entry in $C(v_{i+1})$ which is non-zero mod m, for $1 \leqslant i \leqslant c$.

https://doi.org/10.1017/S0963548310000192 Published online by Cambridge University Press

By permuting rows and columns, we move $L(v_i, v_{i+1})$ to the ith diagonal entry L_{ii} of the Laplacian. The reduced Laplacian is now in the form

$$\begin{pmatrix} I_c & A \\ 0 & 0 \end{pmatrix} \bmod m, \tag{5.5}$$

where I_c is the $c \times c$ identity matrix. Using column operations, we can make all the entries in A divisible by m, without changing the rest of the matrix mod m. After these column operations, the Laplacian is in Smith normal form.

The first c invariant factors of $L(G)$ over $\mathbb{Z}_m$ are 1, so the first c invariant factors of $L(G)$ over $\mathbb{Z}$ are relatively prime to m. The last $|V| - c$ invariant factors of $L(G)$ over $\mathbb{Z}_m$ are 0, so the last $|V| - c$ invariant factors of $L(G)$ over $\mathbb{Z}$ are divisible by m. The lemma holds. □

We use Lemma 5.2 to characterize the critical group of the Kautz and de Bruijn graphs. The first step is finding the orders of these groups. We apply Theorem 1.1 to $\mathrm{DB}_n(m)$, and we let all the variables x_e equal 1, to find that

$$\kappa(\mathrm{DB}_{n+1}(m)) = \kappa(\mathrm{DB}_n(m))\left(m^{(m-1)m^n}\right). \tag{5.6}$$

The number of spanning trees of the complete graph $\mathrm{DB}_1(m)$ is m^{m-1} [4]. By a simple induction, we have

$$\kappa(\mathrm{DB}_n(m)) = m^{m^n - 1}. \tag{5.7}$$

In an Eulerian graph, the sandpile groups $K(G, v)$ are isomorphic for all vertices v, so $|K(G)| = \kappa(G)/|V|$. Therefore, we have

$$|K(\mathrm{DB}_n(m))| = m^{m^n - n - 1}. \tag{5.8}$$

Similarly, we have

$$\kappa(\mathrm{Kautz}_n(m)) = (m+1)^m m^{(m^n-1)(m+1)}, \tag{5.9}$$

$$|K(\mathrm{Kautz}_n(m))| = (m+1)^{m-1} m^{(m^n + m^{n-1} - m - n)}. \tag{5.10}$$

We are ready to prove Theorems 1.2 and 1.3.

Proof of Theorem 1.2. We proceed by induction on n. The critical group of the complete graph on m vertices is $(\mathbb{Z}_m)^{m-2}$, so the base case holds.

Assume that Theorem 1.2 holds for $n - 1$, where $n > 1$. We prove it for n. As shown by Levine [7], if G is a balanced k-regular graph, then

$$kK(\mathcal{L}G) \cong K(G). \tag{5.11}$$

We will use this fact to determine $\mathrm{Syl}_p(K(\mathrm{DB}_n(m)))$, the Sylow-$p$ subgroup of $K(\mathrm{DB}_n(m))$, for any prime p. We break the proof into two cases: either p does not divide m, or p divides m.

If p does not divide m, then by (5.11), we have

$$\mathrm{Syl}_p(K(\mathrm{DB}_n(m))) \cong \mathrm{Syl}_p(K(\mathrm{DB}_{n-1}(m))). \tag{5.12}$$

By the inductive hypothesis, $\mathrm{Syl}_p(K(\mathrm{DB}_n(m)))$ is the trivial group.

https://doi.org/10.1017/S0963548310000192 Published online by Cambridge University Press

Now let p be a prime that divides m, and suppose p^k is the largest power of p that divides m. Let the Sylow-p subgroup of $\mathrm{DB}_n(m)$ be

$$\mathrm{Syl}_p(K(\mathrm{DB}_n(m))) = \mathbb{Z}_p^{a_1} \oplus \mathbb{Z}_{p^2}^{a_2} \oplus \cdots \oplus \mathbb{Z}_{p^l}^{a_l}. \tag{5.13}$$

By Lemma 5.2, $K(\mathrm{DB}_n(m))$ can be written as a direct sum of cyclic groups, such that the order of each group is either non-zero mod p or divisible by p^k. Thus, $a_i = 0$ for $i < k$. Further, we can derive the order of $\mathrm{Syl}_p(K(\mathrm{DB}_n(m)))$ from (5.8). We find that

$$\sum_{i=n}^{l} i a_i = k(m^n - n - 1), \tag{5.14}$$

because the expression on the right-hand side equals the number of factors of p in m^{m^n-n-1}. By (5.11) and the inductive hypothesis, we know that

$$p^k \,\mathrm{Syl}_p(K(\mathrm{DB}_n(m))) = \mathbb{Z}_p^{a_{k+1}} \oplus \mathbb{Z}_{p^2}^{a_{k+2}} \oplus \cdots \oplus \mathbb{Z}_{p^{l-k}}^{a_l} \cong \mathrm{Syl}_p(K(\mathrm{DB}_{n-1}(m))), \tag{5.15}$$

$$\mathbb{Z}_p^{a_{k+1}} \oplus \mathbb{Z}_{p^2}^{a_{k+2}} \oplus \cdots \oplus \mathbb{Z}_{p^{l-k}}^{a_l} \cong (\mathbb{Z}_{p^{nk}})^{m-2} \oplus \bigoplus_{i=1}^{n-2} (\mathbb{Z}_{p^{ik}})^{m^{n-2-i}(m-1)^2}. \tag{5.16}$$

Equation (5.16) implies that $a_{nk} = m - 2$, that $a_{(i+1)k} = m^{n-2-i}(m-1)^2$ for $1 \leqslant i \leqslant n-2$, and that $a_i = 0$ for $i > nk$ or $k \nmid i$. The only a_i which we have not yet determined is a_k. We solve (5.14) for a_k, by moving all the other a_{ik} to the right-hand side and dividing by k:

$$a_k = (m^n - n - 1) - n(m-2) - \sum_{i=2}^{n-1} i(m^{n-1-i}(m-1)^2). \tag{5.17}$$

By evaluating the geometric series, we find that $a_k = m^{n-2}(m-1)^2$. With these values, we may write

$$\mathrm{Syl}_p(K(\mathrm{DB}_n(m))) = (\mathbb{Z}_{p^{nk}})^{m-2} \oplus \bigoplus_{i=1}^{n-1} (\mathbb{Z}_{p^{in}})^{m^{n-1-i}(m-1)^2}. \tag{5.18}$$

The Sylow-p subgroups of $K(\mathrm{DB}_n(m))$ are trivial for $p \nmid m$. Taking the direct sum of the Sylow-p subgroups over primes p which divide m, we find

$$K(\mathrm{DB}_n(m)) \cong \bigoplus_{p \mid m} \mathrm{Syl}_p(K(\mathrm{DB}_n(m))) \cong (\mathbb{Z}_{m^n})^{m-2} \oplus \bigoplus_{i=1}^{n-1} \left(\mathbb{Z}_{m^i}\right)^{m^{n-1-i}(m-1)^2}. \tag{5.19}$$

With this equation we complete the inductive step, as desired. □

Proof of Theorem 1.3. This proof is similar to the proof of Theorem 1.2. Again, we induct on n. Because the critical group of the complete graph on $m+1$ vertices is $(\mathbb{Z}_{m+1})^{m-1}$, the base case holds.

Assume that Theorem 1.3 holds for $n-1$, where $n > 1$. Using (5.11), we calculate the direct sum of the Sylow-p subgroups of $K(\mathrm{Kautz}_n(m))$ over primes p which do not divide m, as follows:

$$\bigoplus_{p \nmid m} \mathrm{Syl}_p(K(\mathrm{Kautz}_n(m))) = \bigoplus_{p \nmid m} \mathrm{Syl}_p(K(\mathrm{Kautz}_1(m))) = (\mathbb{Z}_{m+1})^{m-1}. \tag{5.20}$$

https://doi.org/10.1017/S0963548310000192 Published online by Cambridge University Press

Now let p be a prime that divides m. Suppose that p^k is the largest power of p that divides m, and that the Sylow-p subgroup of $K(\mathrm{Kautz}_n(m))$ is

$$\mathrm{Syl}_p(K(\mathrm{Kautz}_n(m))) = \mathbb{Z}_p^{a_1} \oplus \mathbb{Z}_{p^2}^{a_2} \oplus \cdots \oplus \mathbb{Z}_{p^l}^{a_l}. \tag{5.21}$$

Lemma 5.2 implies that $a_i = 0$ for $i < k$. Furthermore, we know the order of the Sylow-p subgroup of $K(\mathrm{Kautz}_n(m))$ from (5.10), which implies that

$$\sum_{i=k}^{l} i a_i = k\big(m^n + m^{n-1} - m - n\big), \tag{5.22}$$

because the right-hand side equals the number of factors of p in $m^{(m^n+m^{n-1}-m-n)}$. By (5.11), we have

$$p^k \,\mathrm{Syl}_p(K(\mathrm{Kautz}_n(m))) = \mathbb{Z}_p^{a_{k+1}} \oplus \mathbb{Z}_{p^2}^{a_{k+2}} \oplus \cdots \oplus \mathbb{Z}_{p^{l-k}}^{a_l} \cong \mathrm{Syl}_p(K(\mathrm{Kautz}_{n-1}(m))). \tag{5.23}$$

By the inductive hypothesis, we find that $a_{(i+1)k} = m^{n-3-i}(m-1)^2(m+1)$ for $1 \leqslant i \leqslant n-3$, that $a_{(n-1)k} = m^2 - 2$, and that $a_i = 0$ for $i > (n-1)k$ or $k \nmid i$.

We solve (5.22) for a_k. We find that $a_k = m^2 - 2$ if $n = 2$ and that $a_k = m^n - m^{n-1} - m^{n-2} + 1$ if $n > 2$. Then we have

$$\mathrm{Syl}_p(K(\mathrm{Kautz}_n(m))) = \big(\mathbb{Z}_{p^{(n-1)k}}\big)^{m^2-2} \oplus \bigoplus_{i=1}^{n-2} \big(\mathbb{Z}_{p^{ik}}\big)^{m^{n-2-i}(m-1)^2(m+1)}. \tag{5.24}$$

By taking the direct sum of (5.20) and (5.24) over all primes which divide m, we complete the inductive step and prove Theorem 1.3, as desired. □

References

[1] Berget, A., Manion, A., Maxwell, M., Potechin, A. and Reiner, V. The critical group of a line graph. arXiv:0904.1246.

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

[3] Christianson, H. and Reiner, V. (2002) The critical group of a threshold graph. *Linear Algebra Appl.* **349** 233–244.

[4] Holroyd, A. E., Levine, L., Mészáros, K., Peres, Y., Propp, J. and Wilson, D. B. (2008) Chip-firing and rotor-routing on directed graphs. In *In and Out of Equilibrium 2*, Vol. 60 of *Progress in Probability*, Birkhäuser, pp. 331–364.

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

[6] Knuth, D. E. (1967) Oriented subtrees of an arc digraph. *J. Combin. Theory* **3** 309–314.

[7] Levine, L. (2010) Sandpile groups and spanning trees of directed line graphs. To appear in *Journal of Combinatorial Theory, Series A*. arXiv:0906.2809.

[8] Stanley, R. P. (1999) *Enumerative Combinatorics*, Vol. 1, Cambridge University Press.

https://doi.org/10.1017/S0963548310000192 Published online by Cambridge University Press