Eötvös Loránd University
Institute of Mathematics

Ph.D. thesis

The chip-firing game

Lilla Tóthmérész

Doctoral School: Mathematics
Director: Miklós Laczkovich
Professor, member of the Hungarian Academy of Sciences

Doctoral Program: Applied Mathematics
Director: István Faragó
Professor

Supervisor: Zoltán Király
Associate Professor, Department of Computer Science, Eötvös Loránd University

Department of Computer Science, Eötvös Loránd University

2016

Contents

Acknowledgement

I would like to thank my supervisor Zoltán Király for all his help and guidance. I am also very grateful to the members of the EGRES group for the opportunity of working together with them. I would like to thank Bálint Hujter and Viktor Kiss for all the common work we did. Finally, I would like to thank my family for their constant support.

Table of notations

Graphs

G	graph or digraph
$G[V_0]$	subgraph induced by $V_0 \subseteq V(G)$
$V(G)$	vertex set of G
$E(G)$	edge set of G
$\overrightarrow{uv}$	directed edge with tail u and head v
$\overrightarrow{d}(u,v)$	multiplicity of the directed edge $\overrightarrow{uv}$
$d(v)$	degree of vertex v
$\mathbf{d}$	degree vector, $\mathbf{d} \in \mathbb{Z}^V$, $\mathbf{d}(v) = d(v)$ for each $v \in V$
$d^+(v)$	outdegree of vertex v
$\mathbf{d}^+$	outdegree vector, $\mathbf{d}^+ \in \mathbb{Z}^V$, $\mathbf{d}^+(v) = d^+(v)$ for each $v \in V$
$d^-(v)$	indegree of vertex v
$\mathbf{d}^-$	indegree vector, $\mathbf{d}^- \in \mathbb{Z}^V$, $\mathbf{d}^-(v) = d^-(v)$ for each $v \in V$
$\Delta(G)$	maximal outdegree of digraph G
$\Gamma(v)$	set of neighbors of vertex v ($\{u \in V(G) : uv \in E(G)\}$)
$\Gamma^-(v)$	set of in-neighbors of vertex v ($\{u \in V(G) : \overrightarrow{uv} \in E(G)\}$)
$\Gamma^+(v)$	set of out-neighbors of vertex v ($\{u \in V(G) : \overrightarrow{vu} \in E(G)\}$)
$\mathbb{Z}^V$	the set of integer vectors indexed by V
$\mathbb{Z}^V_+$	$\{z \in \mathbb{Z}^V : z(v) \geq 0$ for each $v \in V\}$
$x \leq y$ for $x, y \in \mathbb{Z}^V$	x is coordinatewise smaller or equal to y
$x < y$ for $x, y \in \mathbb{Z}^V$	$x \leq y$ and $\exists v \in V$ such that $x(v) < y(v)$
$x \wedge y$	coordinatewise minimum of x and y
$\mathbf{1}_S$	characteristic vector of the set S
$\mathbf{1}_v$	characteristic vector of the set $\{v\}$
$\mathrm{Arb}(G, v)$	set of spanning in-arborescences of G rooted at v
$\mathrm{minfas}(G)$	size of the minimum cardinality feedback arc set in G
L	Laplacian matrix (see Definition 1.1.3)
$\mathrm{per}(G)$	period length of G (see Definition 1.1.5)

Chip-firing and rotor-routing

$\mathrm{Chip}(G)$	set of chip-distributions on G $(= \mathbb{Z}^{V(G)})$
$\deg(x)$	sum of coordinates of x
$x \rightsquigarrow y$	there is a legal chip-firing game that leads from x to y
$\sim$	linear equivalence (see Definition 1.2.5)
$\mathrm{dist}(x)$	distance from non-terminating distributions (Definition 1.2.4)
ϱ	rotor configuration

Graph divisor theory

$\mathrm{rank}(f)$	rank of divisor f
$\mathrm{Div}(G)$	group of divisors on G
$\mathrm{Div}^0(G)$	group of degree zero divisors on G
$\sim$	linear equivalence
$\mathrm{Pic}^0(G)$	Picard group

Chapter 1

Introduction

This thesis is about three related topics, chip-firing, graph divisor theory and rotor-routing. Chip-firing and rotor-routing are simple, yet interesting diffusion processes on graphs, that have connections to many parts of mathematics, including the Tutte polynomial, graph orientations and random walks [30, 7, 22]. Graph divisor theory is a discrete analogue of the divisor theory of Riemann surfaces, that has strong connections to chip-firing.

Chip-firing has been introduced independently by many researchers, working in diverse areas. Björner, Lovász, and Shor introduced it as a one-player combinatorial game [8], Dhar, as a model exhibiting self-organized critical behaviour [13], and Engel, as a pedagogical tool (the probabilistic abacus) [15]. In this thesis, we adopt the viewpoint of Björner, Lovász and Shor, and think of chip-firing as a one-player combinatorial game. This game is played on a digraph, where on each vertex, there is an integer number of chips. If a vertex has at least as many chips as its out-degree, it is allowed to "fire", i.e. to pass a chip to its neighbors along each out-edge incident to it. In Chapter 2, we investigate the chip-firing reachability problem: Given two chip-distributions x and y on a digraph G, decide whether there exists a legal game transforming x to y. We show that this problem is in **co**-**NP**, and for digraphs with polynomial period length, it is in **P** (even if there are multiple edges). Moreover, we show that if the target distribution is recurrent (i.e. reachable from itself by a nonempty legal game), then a trivial necessary condition is sufficient for the reachability. These results are joint work with Bálint Hujter and Viktor Kiss.

Graph divisor theory is a discrete analogue of the divisor theory of Riemann surfaces. Divisors on graphs, and the Picard group of a graph have been defined by Bacher, de la Harpe and Nagnibeda in 1997 [3]. In 2007, Baker and Norine defined the rank of a graph divisor, and proved the analogue of the Riemann–Roch theorem for this notion [5]. It remained an intriguing open question whether the rank of a graph divisor can be computed in polynomial time. In Chapter 3, we show that computing the rank of a divisor on a graph is **NP**-hard, even for simple graphs.

The results of this chapter are joint work with Viktor Kiss.

The Riemann–Roch theorem of Baker and Norine inspired the research for Riemann–Roch theorems in similar settings, including tropical curves [18, 31], lattices [1], and directed graphs [2]. In Chapter 4, we prove a Riemann–Roch inequality for Eulerian digraphs, that generalizes the Riemann–Roch theorem for undirected graphs by Baker and Norine. A weaker form of this inequality has been proved earlier by Amini and Manjunath [1]; we obtain a stronger result by a simpler proof. We also investigate the natural Riemann–Roch property introduced by Asadi and Backman, proving that an Eulerian digraph has the natural Riemann–Roch property if and only it corresponds to an undirected graph. The results of Chapter 4 are joint work with Bálint Hujter.

Rotor-routing was introduced by Priezzhev et al. [35] under the name Eulerian walkers, as a model of self-organized criticallity, and later it was rediscovered several times [36, 14]. Rotor-routing is a one-player game on a digraph, that can be thought of as a refined version of chip-firing. An important application of this game is that one can define a group action of the Picard group on the spanning in-arborescences of a digraph through rotor-routing [21]. In Chapter 5, we characterize recurrent elements for the rotor-routing game. Also, we define the linear equivalence of configurations, and for Eulerian digraphs, give an interpretation of the rotor-routing action in terms of linear equivalence.

Let us now give a more detailed introduction for the notations and notions used in the thesis. To help readability, the notions of rotor-routing (that are only used in Chapter 5) are introduced in Chapter 5.

1.1 Digraphs

Throughout this thesis, *digraph* means a weakly connected directed graph that can have multiple edges but no loops. A digraph is usually denoted by G. The vertex set and edge set of a digraph G are denoted by $V(G)$ and $E(G)$ (or simply V and E), respectively. For a vertex v, the indegree and the outdegree of v are denoted by $d^-(v)$ and $d^+(v)$, respectively. The set of in-neighbors (out-neighbors) of v is denoted by $\Gamma^-(v)$ $(\Gamma^+(v))$. We denote a directed edge leading from vertex u to vertex v by $\overrightarrow{uv}$. In this case u is called the *tail*, and v is called the *head* of the edge $\overrightarrow{uv}$. The multiplicity of a directed edge $\overrightarrow{uv}$ is denoted by $\overrightarrow{d}(u,v)$. The maximal outdegree of a digraph G is denoted by $\Delta(G)$.

A digraph is *simple*, if $\overrightarrow{d}(u,v) \leq 1$ and $\overrightarrow{d}(v,u) \leq 1$ for each pair of vertices $u, v \in V$. A digraph is *Eulerian*, if $d^+(v) = d^-(v)$ for each $v \in V$. A digraph is *strongly connected*, if for each pair of vertices u, v, there is a directed path from u to v, and also from v to u. A connected Eulerian digraph is always strongly connected. Each digraph has a unique decomposition into strongly connected components. A component is called a *sink component*, if there is no edge leaving the component. Note that a digraph always has at least one sink-component.

Definition 1.1.1. For a digraph G and vertex $r \in V(G)$ a *spanning in-arborescence* of G rooted at r is a subdigraph T such that $d_T^+(v) = 1$ for each $v \in V(G) - r$, and the underlying undirected graph of T is a tree.

Definition 1.1.2. A *feedback arc set* of a digraph G is a set of edges $F \subseteq E(G)$ such that the digraph $G' = (V(G), E(G) \setminus F)$ is acyclic. We denote

$$\text{minfas}(G) = \min\{|F| : F \subseteq E(G) \text{ is a feedback arc set}\}.$$

We denote by $\mathbb{Z}^V$ the set of integer vectors indexed by the vertices of a digraph G. We identify vectors in $\mathbb{Z}^V$ with integer valued functions on V. According to this, we write $z(v)$ for the coordinate corresponding to vertex v of a $z \in \mathbb{Z}^V$. By $\mathbb{Z}_+^V$ we denote the set of vectors with nonnegative integer coordinates. For two vectors x and y in $\mathbb{Z}^V$, we denote by $x \le y$ if x is coordinatewise smaller or equal to y. By $x < y$, we mean that $x \le y$, an there is a coordinate on which x is strictly smaller than y. The coordinatewise minimum of x and y is denoted by $x \wedge y$. For an integer vector $x \in \mathbb{Z}_+^V$, we denote the sum of its coordinates by $\deg(x)$.

For $S \subseteq V$, we denote the characteristic vector of S by $\mathbf{1}_S$, i.e. $\mathbf{1}_S(v) = 1$ if $v \in S$, and $\mathbf{1}_S(v) = 0$ if $v \notin S$. If $S = \{v\}$, we use the notation $\mathbf{1}_v$. We denote the vector with each coordinate equal to zero by $\mathbf{0}$. If we want to emhasize the (di)graph G whose vertices index the coordinates, we write $\mathbf{0}_G$. For a digraph, $\mathbf{d}^+ \in \mathbb{Z}^V$ (resp. $\mathbf{d}^- \in \mathbb{Z}^V$) is the vector where $\mathbf{d}^+(v) = d^+(v)$ (resp. $\mathbf{d}^-(v) = d^-(v)$) for each vertex v. Again, if we want to emphasize the underlying graph, we put it in subscript.

Definition 1.1.3. The *Laplacian* of a digraph G is the following matrix $L \in \mathbb{Z}^{V \times V}$:

$$L(u,v) = \begin{cases} -d^+(v) & \text{if } u = v, \\ \overrightarrow{d}(v,u) & \text{if } u \neq v. \end{cases}$$

The eigenvectors of the Laplacian matrix corresponding to the eigenvalue zero play an important role in the chip-firing theory. A non-negative vector $p \in \mathbb{Z}_+^V$ is called a *period vector* for G if $Lp = \mathbf{0}$. A non-zero period vector is called *primitive* if its entries have no non-trivial common divisor. The following proposition follows from [7, 3.1 and 4.1].

Proposition 1.1.4. *For a strongly connected digraph G there exists a unique primitive period vector* per_G*, moreover, it is strictly positive. If G is connected Eulerian, then* $\text{per}_G = \mathbf{1}_V$*. For a general digraph G, if $G_1, \dots, G_k$ are the sink components of G and a vector $z \in \mathbb{Z}^V$ satisfies $Lz = \mathbf{0}$ then $z = \sum_{i=1}^k \lambda_i p_i$, where for $i \in \{1, \dots, k\}$, $\lambda_i \in \mathbb{Z}$ and p_i is the primitive period vector of G_i restricted to $V(G_i)$ and zero elsewhere.*

Definition 1.1.5. For a strongly connected digraph G, let us denote by per_G the unique primitive period vector of G. The sum of the coordinates of per_G is denoted by $\mathrm{per}(G)$. For a general digraph G let $\mathrm{per}(G) = \sum_{i=1}^{\ell} \mathrm{per}(G_i)$ where $G_1, \ldots, G_\ell$ are the strongly connected components of G. We call this quantity the *period length* of the graph.

We point out that $\mathrm{per}(G)$ can be exponentially large in the size of the description of the graph. As an example, consider the following sequence of graphs G_2, G_3, $\ldots$, where $V(G_n) = \{v_1, \ldots v_n\}$, $E(G_n) = \{\overrightarrow{v_i v_{i+1}} : 1 \le i \le n-1\} \cup \{\overrightarrow{v_i v_1} : 2 \le i \le n\}$. For an example, see G_4 on Figure 1.1. Note that these are strongly connected graphs. It is easy to check that

$$\mathrm{per}_{G_n}(v_i) = \begin{cases} 2^{n-i-1} & \text{for } i \in \{1, \ldots, n-1\} \\ 1 & \text{for } i = n. \end{cases} ,$$

hence $\mathrm{per}(G_n) = 2^{n-1}$.

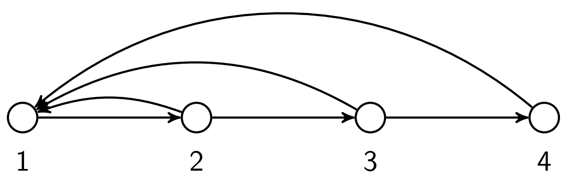

Figure 1.1: G_4

Let us say a few words about our conventions concerning undirected graphs. In most parts of this thesis, we identify undirected graphs with the digraph obtained by replacing each edge with a pair of oppositely directed edges. This way, undirected graphs become special Eulerian digraphs. We use the term *bidirected graph* for those digraphs which correspond to an undirected graph in the above sense, i.e. those digraphs where $\overrightarrow{d}(u,v) = \overrightarrow{d}(v,u)$ for each pair of vertices u, v. Still, in some parts of this thesis, were we talk specially about undirected graphs, it will be more convenient to think about undirected graphs in the ordinary way. If we talk specially about undirected graphs, we use the notation $d(v)$ for the degree of a vertex v, and $\Gamma(v)$ for the set of neighbors of v, while $\mathbf{d} \in \mathbb{Z}^V$ denotes the vector with $\mathbf{d}(v) = d(v)$ for all $v \in V$.

1.1.1 Algorithms

If we give a digraph as an input to an algorithm, we always encode it by its adjacency matrix. Hence the size of the input is not increased by the values of the edge multiplicities, just the logarithms of them. An algorithm runs in *polynomial time* if the number of basic steps it makes is bounded by a polynomial in the size of the input. If the input of an algorithm consists of integer numbers, we can talk about strongly polynomial running time. An algorithm runs in *strongly polynomial time* if the following two conditions are satisfied:

1. in the model, where basic arithmetic operations (addition, subtraction, multiplication, division, and comparison) take a unit time step to perform, its running time is bounded by a polynomial in the number of integers contained in the input;
2. the space used by the algorithm is bounded by a polynomial in the size of the input.

For a more detailed explanation, see [19, Chapter 1.3].

1.2 Chip-firing

In a chip-firing game we consider a digraph G with a pile of chips on each of its nodes. A position of the game, called a *chip-distribution* (or just distribution) is described by a vector $x \in \mathbb{Z}^V$, where $x(v)$ is interpreted as the number of chips on vertex $v \in V$. We denote the set of all chip-distributions on G by $\mathrm{Chip}(G)$. Note that though originally Björner, Lovász and Shor defined chip-firing only for non-negative chip-distributions, in this thesis, we allow vertices to have a negative number of chips. This does not change the main characteristics of the game, but sometimes it will be more convenient to allow negative entries. We use the notation $\deg(x)$ for the number of chips in a chip-distribution, i.e. $\deg(x) = \sum_{v\in V} x(v)$.

The basic move of the chip-firing game is *firing* a vertex. It means that this vertex passes a chip to its neighbors along each outgoing edge, and so its number of chips decreases by its outdegree. In other words, firing a vertex v means taking the new chip-distribution $x + L\mathbf{1}_v$ instead of x.

A vertex $v \in V(G)$ is *active* with respect to a chip-distribution x, if $x(v) \geq d^+(v)$. The firing of a vertex $v \in V(G)$ is *legal*, if v was active before the firing. In other words, the firing of a vertex v is legal, if v has a non-negative amount of chips after the firing. A *legal game* is a sequence of distributions in which every distribution is obtained from the previous one by a legal firing. A legal game terminates if there is no active vertex with respect to the last distribution. The *firing vector* of a game is a vector $f \in \mathbb{Z}_+^V$, where $f(v)$ equals the number of times v was fired during the game.

Chip-firing on an undirected graph is defined as chip-firing on the corresponding bidirected graph. Thinking of an undirected graph in the "ordinary" sense, this means that when firing a vertex, it passes a chip along each edge incident to it.

The following theorem of Björner, Lovász and Shor describes a fundamental "Abelian" property of the chip-firing game.

Theorem 1.2.1. *[8, Remark 2.4] From a given initial chip-distribution on a digraph G, either every legal game can be continued indefinitely, or every legal game terminates after finitely many steps. The firing vector of every maximal legal game is the same.*

We present here a simple proof for this theorem, found by Mikkel Thorup [39].

Proof. [39] By symmetry, it is enough to prove that if a legal game from an initial chip-distribution x terminates with firing vector f, then in any legal game started from x, any vertex v can fire at most $f(v)$ times.

Suppose for contradiction that there is a legal game started from x that fires some vertex more times than its coordinate in f. Take the first moment when a vertex is to be fired more times than its coordinate in f. Let this vertex be v. Let the firing vector of the game until this moment be h. Hence $h(v) = f(v)$, and $h(u) \leq f(u)$ for any $u \neq v$. We see that until this moment v lost the same number of chips as in the game with firing vector f, since it fired the same number of times in both games. On the other hand, it gained at most as many chips as in the game with firing vector f, since its inneighbors fired at most as many times as in the game with firing vector f. Hence now v has at most as many chips as at the end of the game with firing vector f, that is, at most $(x + Lf)(v)$ chips. As that game terminated, $(x + Lf)(v) \leq d^+(v) - 1$, hence v cannot be active at this moment, which is a contradiction with the fact that it is to be fired in the next step. □

Based on Theorem 1.2.1, we call a distribution x *terminating* if a legal game (hence, all legal games) started from x terminates, and we call x *non-terminating* otherwise.

Two very natural questions about the chip-firing game are the followings:

a) *(Chip-firing halting problem)* Given a digraph G, and a chip-distribution $x \in \text{Chip}(G)$, decide whether x is terminating or not.

b) For a digraph G, how long can be the longest terminating game?

It is easy to see, that if we do not require that the initial chip-distribution is non-negative on each vertex, then there is no upper bound on the length of terminating legal games. For an arbitrary digraph G and positive integer k, consider the chip-distribution $\mathbf{0} - k \cdot L\mathbf{1}_v$, where $v \in V(G)$ is an arbitrary vertex. It is easy to see that v can legally fire k times, and after these firings, we arrive at the chip-distribution $\mathbf{0}$, hence the game terminates.

Nevertheless, Question b) is meaningful, if we require that the initial distribution is non-negative on each vertex. Björner and Lovász gives the following upper bound.

(The case of undirected graphs was solved earlier by Tardos [37]). This upper bound will play an important role in an **NP**-hardness reduction in Chapter 3.

Theorem 1.2.2 ([7, Theorem 4.8])**.** *On a directed graph G, every terminating legal game started from an initial distribution that is non-negative on each vertex makes at most*

$$2|V(G)||E(G)|\Delta(G)\mathrm{per}(G)$$

firings.

Lovász and Winkler also shows, that on a strongly connected digraph, there is always a terminating game with a non-negative initial distribution whose length is proportional to the period length:

Theorem 1.2.3 ([27, Theorem 6.9])**.** *On any strongly connected digraph G, there exists a terminating chip-distribution that is non-negative on each vertex, such that the maximal legal game started from it has length* $\mathrm{per}(G) - |V(G)|$.

It is easy to give a family of examples that shows that the length of a terminating game started from a non-negative chip-distribution can also be proportional to the number of edges. By a result of Björner, Lovász and Shor [8, Theorem 2.3], if a chip-distribution on a connected undirected graph G has less than $|E(G)|$ chips, then it is necessarily terminating (here we understand $|E(G)|$ in the "ordinary" undirected sense). Now for any nonnegative integers $k \geq 3$ and $\ell \geq k$, we can take a connected graph with k vertices and ℓ edges such that the graph has at least one vertex of degree one. Place $\ell - 1$ chips at the vertex of degree one, and zero elsewhere. Then the chip-distribution is terminating, since there are less chips than the number of edges. On the other hand, the vertex of degree one can do at least $\ell - 1$ legal firings.

Note that in our model of computation, the terms $|E(G)|, \Delta(G)$ and $\mathrm{per}(G)$ can all be exponentially large in the input size. Hence a terminating game can be exponentially long, even if the initial distribution is non-negative on each vertex. The above example shows that the length of a terminating game started from a non-negative distribution can be exponentially large even on undirected graphs (where $\mathrm{per}(G) = |V(G)|$). Hence the problem of deciding whether a chip-distribution in terminating or not is nontrivial even in these special cases.

It was shown by Farrell and Levine [16], that the chip-firing halting problem is **NP**-hard. In Chapter 2, we show that for Eulerian digraphs, the chip-firing halting problem is in $\mathbf{NP} \cap \mathbf{co} - \mathbf{NP}$.

It is easy to see by the pidgeonhole-principle, that if a chip-distribution on a digraph G has more than $|E(G)| - |V(G)|$ chips, then there is always an active vertex, hence the distribution is non-terminating [7]. Consequently, the following quantity is well defined.

Definition 1.2.4. For a distribution $x \in \mathrm{Chip}(G)$, let

$$\mathrm{dist}(x) = \min\{\deg(y) : y \in \mathrm{Chip}(G), y \geq \mathbf{0}, x + y \text{ is non-terminating}\}.$$

We say that $\mathrm{dist}(x)$ is the *distance* of x from non-terminating distributions.

This quantity will be important in Chapters 3 and 4, since it is a counterpart of the notion of rank from the discrete Riemann–Roch theory.

Let us introduce an equivalence relation on chip-distributions. This equivalence relation comes from the discrete Riemann–Roch theory, but as we will see, it is also very useful for analysing classical chip-firing questions.

Definition 1.2.5. We say that two chip-distributions x and y on a digraph G are *linearly equivalent*, if there exists $z \in \mathbb{Z}^V$ such that $y = x + Lz$.

It is easy to see that linear equivalence is indeed an equivalence relation. The usefulness of this notion is based on the following lemma, which appears first in [9, Lemma 4.3.]. To be self-contained, we give a proof.

Lemma 1.2.6. *Let G be a strongly connected digraph and $x, y \in \mathrm{Chip}(G)$. If $x \sim y$, then x is terminating if and only if y is terminating.*

Proof. By symmetry, it is enough to prove that if x is terminating, then y is also terminating.

Let x be a terminating chip-distribution. Play the chip-firing game starting from x until it terminates. Let the final configuration be x^*. Clearly, $x^* \sim x \sim y$. Let $z \in \mathbb{Z}^{V(G)}$ be a vector with $x^* = y + Lz$. We can suppose that $z \in \mathbb{Z}_+^{V(G)}$, since by Proposition 1.1.4, the Laplacian of a strongly connected digraph has a strictly positive eigenvector with eigenvalue zero. Start a game from y in the following way: If there is an active vertex v that has been fired less than $z(v)$ times, then one such vertex is fired. If there is no such vertex, the game ends. Clearly, after at most $\sum_{v \in V(G)} z(v)$ steps, this modified game ends. We claim that for the final distribution $y' = y + Lz'$ (where $z' \leq z$), $y'(v) < d^+(v)$ for each vertex v. Indeed, as the game stopped, for any vertex v with $y'(v) \geq d^+(v)$, $z'(v) = z(v)$. As x^* is stable, $x^*(v) < d^+(v)$. But then from $x^* = y' + L(z - z')$ and $z(v) = z'(v)$, we get $d^+(v) > x^*(v) \geq y'(v)$, which is a contradiction. □

Another useful property of the notion of linear equivalence is that it can be decided in polynomial time whether $x \sim y$ holds for two chip-distributions x and y. Indeed, we need to decide whether the system of linear equalities $Lf = y - x$ has an integer solution. As L is an integer matrix and $y - x$ is an integer vector, by [19, Theorem 1.4.21], this can be done in polynomial time.

Claim 1.2.7. *For two chip-distributions on a digraph G, whether $x \sim y$ holds can be decided in polynomial time.*

Another basic problem in chip-firing is the reachability problem. This can be defined in the following way: Given two chip-distributions x and y on a digraph G, decide if there exists a legal game that transforms x to y. Previously, the complexity of the reachability problem was only known for simple Eulerian digraphs, where it was shown to be in **P** [7, Theorem 5.1]. In Chapter 2, we show that the reachability problem is in **co-NP** for general digraphs. Moreover, we give a polynomial algorithm for graphs where $\text{per}(G)$ is polynomial in the input size. We also show that if the target distribution is recurrent (i.e. reachable from itself by a nonempty legal game), then a trivial necessary condition is sufficient for the reachability.

1.3 Graph divisor theory

In this section we give the basic definitions of the graph divisor theory. Originally, Baker and Norine introduced graph divisor theory for undirected graphs. For directed graphs, the theory is less well developed, and the Riemann–Roch theorem does not hold in general. Nevertheless, as the basic notions can be defined for strongly connected digraphs as well, in this introduction, we give the definitions for the case of strongly connected digraphs.

The basic objects are called *divisors*. For a strongly connected digraph G, $\text{Div}(G)$ is the free abelian group on the set of vertices of G. An element $f \in \text{Div}(G)$ is called a *divisor*. The *degree* of a divisor is the following:

$$\deg(f) = \sum_{v \in V(G)} f(v).$$

We denote the set of divisors on G of degree k by $\text{Div}^k(G)$. Note that $\text{Div}^0(G)$ is a subgroup of $\text{Div}(G)$ for the coordinatewise addition.

The following equivalence relation on $\text{Div}(G)$ is called *linear equivalence*: For $f, g \in \text{Div}(G)$, $f \sim g$ if there exists a $z \in \mathbb{Z}^V$ such that $g = f + Lz$.

A divisor $f \in \text{Div}(G)$ is *effective*, if $f(v) \geq 0$ for each $v \in V(G)$. A divisor is called *equi-effective*, if it is linearly equivalent to an effective divisor.

A basic quantity associated to a divisor is its rank.

Definition 1.3.1 (The rank of a divisor, [5])**.**

$$\text{rank}(f) = \min\{\deg(g) - 1 : \; g \in \text{Div}(G),\, g \text{ is effective},\, f - g \text{ is not equi-effective } \}.$$

When we wish to emphasize the underlying graph, we write $\text{rank}_G(f)$ instead of $\text{rank}(f)$.

In [5], Baker and Norine proved that for undirected graphs, this notion of rank satisfies a Riemann–Roch theorem.

Theorem 1.3.2 (Riemann–Roch theorem for graphs [5, Theorem 1.12]). *Let G be an undirected graph and let f be a divisor on G. Then*

$$\text{rank}(f) - \text{rank}(K_G - f) = \deg(f) - \mathfrak{g} + 1$$

where $\mathfrak{g} = |E(G)| - |V(G)| + 1$ *and* $K_G(v) = d(v) - 2$ *for each* $v \in V(G)$.

The rank of a divisor on a graph is a purely combinatorial notion. The question whether it can be computed in polynomial time has been posed in several papers [20, 29, 6], originally attributed to H. Lenstra. In Chapter 3, we prove that the computation of the rank of a divisor is **NP**-hard, even on simple undirected graphs.

The discrete Riemann–Roch theorem of Baker and Norine inspired much research about Riemann–Roch theorems in similar settings, including tropical curves, lattices and directed graphs [1, 2, 18]. In Chapter 4, we investigate the Riemann–Roch property on directed graphs, using the connection of chip-firing to divisor theory.

1.3.1 The Picard group

There is a group connected to chip-firing and divisor theory, that is either called the Picard group, the Jacobian group or the Sandpile group. This group can be defined in various ways, see for example [3, 21]. Let us give one of these definitions.

Note that the divisors linearly equivalent to $\mathbf{0}$ form a subgroup of $\text{Div}^0(G)$ which is isomorphic to $Im(L)$, the image of the linear operator on $\mathbb{Z}^V$ corresponding to L. The *Picard-group* is the factor group of $\text{Div}^0(G)$ by linear equivalence:

$$\text{Pic}^0(G) = {\text{Div}^0(G)} \Big/ {Im(L)}.$$

1.4 The connection between chip-firing and graph divisor theory

The notions of chip-firing and graph divisor theory are connected by a simple duality. This phenomenon was discovered by Baker and Norine [5] in their first paper about graph divisor theory. This connection will be crucial in Chapters 3 and 4 of this thesis. Let us describe this duality.

Let G be a strongly connected digraph. For a divisor $f \in \text{Div}(G)$, we call $\mathbf{d}^+ - \mathbf{1}_V - f$ the *dual pair* of f, and think of it as a chip-distribution. Note that each chip-distribution is a dual pair of some divisor. The connection between chip-firing and graph divisor theory is established by the following proposition.

Proposition 1.4.1 ([5, Corollary 5.4]). *A divisor $f \in \mathrm{Div}(G)$ on a strongly connected digraph G is equi-effective if and only if $\mathbf{d}^+ - \mathbf{1}_V - f$ is a terminating chip-distribution.*

In [5], the proposition is stated only for undirected graphs, hence we give a short proof here.

Proof of Proposition 1.4.1 [24]. If f is equi-effective, fix a divisor $f^* \geq 0$ such that $f \sim f^*$. Then the chip-distribution $\mathbf{d}^+ - \mathbf{1}_V - f^*$ has no active vertex, hence $\mathbf{d}^+ - \mathbf{1}_V - f^*$ is necessarily terminating. As $f^* \sim f$, we have $\mathbf{d}^+ - \mathbf{1}_V - f^* \sim \mathbf{d}^+ - \mathbf{1}_V - f$. Since $\mathbf{d}^+ - \mathbf{1}_V - f^*$ is terminating, by Lemma 1.2.6, $\mathbf{d}^+ - \mathbf{1}_V - f$ is also terminating.

On the other hand, if $x := \mathbf{d}^+ - \mathbf{1}_V - f$ is a terminating chip-distribution, then we play the game until it terminates at some chip distribution x^*. Clearly, $x^* \sim x$. Since the game terminated, $x^*(v) \leq d^+(v) - 1$ on each vertex. Hence $f^* := \mathbf{d}^+ - \mathbf{1}_V - x^* \geq \mathbf{0}$. Moreover, $f^* = \mathbf{d}^+ - \mathbf{1}_V - x^* \sim \mathbf{d}^+ - \mathbf{1}_V - x = f$. □

The following is a straightforward consequence of Proposition 1.4.1.

Corollary 1.4.2. *For any $f \in \mathrm{Div}(G)$ on a strongly connected digraph G, the following holds:*

$$\mathrm{rank}(f) = \mathrm{dist}(\mathbf{d}^+ - \mathbf{1}_V - f) - 1$$

Chapter 2

Reachability

This chapter is based on [23], which is joint work with Bálint Hujter and Viktor Kiss.

2.1 Introduction

In this chapter, we analyze the complexity of the chip-firing reachability problem: given two chip-distributions x and y, decide whether y can be reached from x by playing a legal game. This question is a special case of the reachability problem for integral vector addition systems [7]. It was first considered by Björner and Lovász, who gave an algorithm that decides the reachability problem and runs in polynomial time for simple digraphs with polynomial period length [7]. The complexity of the reachability problem was left open both for Eulerian digraphs with multiple edges and for digraphs with large period length. The question whether the reachability problem is in **NP** or in **co-NP** was also left open.

In this chapter, we show that the chip-firing reachability problem is in **co-NP**. Also, we give an algorithm for the chip-firing reachability problem that runs in polynomial time for digraphs with polynomial period length (even if they have multiple edges). This case includes for example Eulerian digraphs with multiple edges. In addition, for Eulerian digraphs, our algorithm is strongly polynomial. The main ingredient of the algorithm is a lemma stating that if one chip-distribution is reachable from another, then it can be reached by a game of nice structure.

Also, we show that for a special class of target chip-distributions, the chip-firing reachability problem is polynomial time solvable on general digraphs.

Finally, in Section 2.5, we collect some open problems related to the reachability problem. In this last section, we show that the chip-firing halting problem is in $\mathbf{NP} \cap \mathbf{co} - \mathbf{NP}$ for Eulerian digraphs, which makes it a good candidate for the search of a polynomial algorithm.

2.2 Preliminaries

Let us sum up the previous results about the chip-firing reachability problem.

It turns out, that the following bounded variant of the chip-firing game plays an important role in the reachability problem:

Definition 2.2.1. For a given vector $b \in \mathbb{Z}_+^V$, let us call the following game *chip-firing game with upper bound b*: We are only allowed to make legal firings, and each vertex v can be fired at most $b(v)$ times during the whole game.

Björner and Lovász show the "Abelian" property for the bounded chip-firing game as well.

Lemma 2.2.2. *[7] For a given bound $b \in \mathbb{Z}_+^V$ and initial distribution x, each maximal bounded game with upper bound b and initial distribution x has the same firing vector.*

Proof. Follows from Lemmas 1.2, 1.3 and 1.4 of [7]. □

The following lemma of Björner and Lovász will also be an important tool.

Lemma 2.2.3. *[7, Lemma 4.3] Let p be a period vector of a digraph G, and suppose that $\alpha = (v_{i_1}, v_{i_2}, \dots, v_{i_s})$ is a legal sequence of firings on G from some initial distribution. Let α' be the sequence obtained from α by deleting the first $p(v)$ occurrence of each vertex v (if v occurs less than $p(v)$ times in α, then we delete all of its occurrences). Then α' is also a legal sequence of firings from the same initial distribution.*

For completeness, we give a proof.

Proof ([7]). Let $\alpha' = (v_{j_1}, \dots v_{j_r})$. Suppose that $(v_{j_1}, \dots v_{j_k})$ is a legal sequence of firings for some k. We show that then $(v_{j_1}, \dots v_{j_k}, v_{j_{k+1}})$ is also a legal sequence of firings.

Let $v_{j_{k+1}} = w$. Take the firing of w in α corresponding to the $k+1^{th}$ firing in α'. As α is a legal sequence of firings, at that moment, w has at least $d^+(w)$ chips. If now we delete the first $p(v)$ occurrence of each vertex in α, we delete $p(w)$ occurrences of w before the considered moment. On the other hand, we delete at most $p(u)$ occurrences of each in-neighbor u of w. Hence in α', before the $k+1^{th}$ firing, w gives out $p(w)d^+(w)$ less chips than in α until the corresponding moment, and the in-neighbors of w give at most $\sum_{u \in \Gamma^-(v)} p(u)d^+(u) = p(w)d^+(w)$ less chips to w. Hence in α', before the $k+1^{th}$ firing, w has at least as many chips as in α in the corresponding moment, thus w can be legally fired at the next step. □

A non-negative vector $f \in \mathbb{Z}_+^V$ is called *reduced* if $f \not\geq p$ for every non-zero period vector p, or, equivalently, if for any non-zero period vector p, $f \wedge p < p$. The following phenomenon is a direct consequence of the previous lemma:

Lemma 2.2.4. *[7, Lemma 5.2] If $x \rightsquigarrow y$, then there exists a legal game transforming x to y with a reduced firing vector.*

Note that if $x \rightsquigarrow y$ then for the firing vector f of a legal game transforming x to y, $y = x + Lf$. Note also that among the vectors $g \in \mathbb{Z}_+^V$ satisfying $y = x + Lg$, there is a unique one that is reduced.

Corollary 2.2.5. *Let G be a digraph, and $x, y \in \mathrm{Chip}(G)$. $x \rightsquigarrow y$ if and only if there exists a reduced vector f such that $y = x + Lf$ and there exists a legal game from initial distribution x with firing vector f.*

In particular, the existence of a reduced vector f such that $y = x + Lf$ is a necessary condition for $x \rightsquigarrow y$. The following claim tells us that this necessary condition can be decided in polynomial time.

Claim 2.2.6. *There is a polynomial algorithm that for a given digraph G and $x, y \in Chip(G)$ decides whether there exists a reduced vector f such that $y = x + Lf$, and if such a vector exists, it computes one.*
In the case of Eulerian digraphs, this can be done in strongly polynomial time.

Proof. First, let G be Eulerian. If $\deg(x) \neq \deg(y)$, then there cannot be such an f. Now suppose that $\deg(x) = \deg(y)$. As a connected Eulerian digraph is strongly connected, the Laplacian matrix L of G has a one-dimensional kernel, and for an arbitrary vertex $v \in V(G)$, the matrix L_v obtained from L by deleting the row and column corresponding to v is nonsingular. We can compute L_v^{-1} in strongly polynomial time [19, Corollary 1.4.9]. Let x_v and y_v be the vectors we get from x and y by deleting the coordinate corresponding to v, respectively. Let $g \in \mathbb{R}^V$ be the vector with

$$g(u) = \begin{cases} (L_v^{-1}(y_v - x_v))(u) & \text{if } u \neq v \\ 0 & \text{if } u = v. \end{cases}$$

It is easy to see that $y(u) = (x + Lg)(u)$ for each $u \neq v$, and since $\sum_{u \in V} x(u) = \sum_{u \in V} y(u)$, we have $y = x + Lg$. The coordinates of g are not necessarily integer. All the vectors f such that $y = x + Lf$ are of the form $g - c \cdot \mathrm{per}_G$, therefore we need to decide if there is a reduced vector of this form. As now G is Eulerian, $\mathrm{per}_G = \mathbf{1}_G$. Since $g(v) = 0$, c needs to be an integer, hence g also needs to be an integer vector. If g is an integer vector, choosing $c = \min\{g(u) : u \in V\}$, $f := g - c \cdot \mathrm{per}_G$ is a reduced vector such that $y = x + Lf$.

If G is not Eulerian, we proceed with the following polynomial, although not strongly polynomial algorithm. By [19, Theorem 1.4.21], we can decide in polynomial time if the equation $Lg = y - x$ has an integer solution, and if it does, compute one. By Proposition 1.1.4, a nonnegative solution exists if and only if the g we got from solving $Lg = y - x$ has nonnegative coordinates on the non-sink components. If g is nonnegative on the non-sink components, we can make it reduced by adding (subtracting) appropriate period vectors. □

Corollary 2.2.5, Lemma 2.2.2 and Claim 2.2.6 imply that the reachability question can be decided “greedily”: For given $x, y \in \text{Chip}(G)$ one can decide if there exists a reduced vector f with $y = x + Lf$. If no such vector exists then $x \not\rightsquigarrow y$. If such a vector f exists, it can be computed. By Corollary 2.2.5, $x \rightsquigarrow y$ if and only if there is a legal game from x to y with firing vector f. By Lemma 2.2.2, we can find greedily a maximal chip-firing game from x with upper bound f. There exists a legal game from x with firing vector f if and only if this maximal bounded chip-firing game has firing vector f.

This reasoning gives an algorithm for deciding the reachability problem, however, this algorithm is in general not polynomial, as the firing vector f may have exponentially large elements. Björner and Lovász improve this greedy algorithm by a scaling-like technique, and obtain the following:

Theorem 2.2.7. *[7, Theorem 5.1] There is an algorithm that for given $x, y \in \text{Chip}(G)$ on a digraph G decides whether $x \rightsquigarrow y$ holds, and runs in*

$$O(|V|^2 \Delta(G)^2 \text{per}(G) \log(|V| \cdot \Delta(G) \cdot \deg(x) \cdot \text{per}(G)))$$

time.

This algorithm is not polynomial in general, as $\text{per}(G)$ and $\Delta(G)$ may be exponentially large. However, as for simple Eulerian digraphs, $\text{per}(G) = |V|$ and $\Delta(G) \leq |V|$, the algorithm is weakly polynomial for simple Eulerian digraphs.

In this chapter, we show that the reachability problem can be decided in polynomial time if $\text{per}(G)$ is a polynomial of the input size, even if the graph has multiple edges. This case includes for example Eulerian digraphs with multiple edges. In addition, for Eulerian digraphs, our algorithm is strongly polynomial. For general digraphs, we show that the reachability problem is in **co-NP**. We also show that in the special case if y is recurrent restricted to each strongly connected component, whether $x \rightsquigarrow y$ holds can be decided in polynomial time for general digraphs.

2.3 An algorithm for digraphs with polynomial period length

In this section, we describe our algorithm for deciding the chip-firing reachability problem, that runs in polynomial time for multigraphs if $\text{per}(G)$ is polynomial. We first give an algorithm for strongly connected digraphs, then show how to solve the question on general digraphs by applying the algorithm for strongly connected digraphs to the strongly connected components.

The heart of our algorithm is Lemma 2.3.3, that ensures that if $x \rightsquigarrow y$, then there is a legal game from x to y with a certain nice structure. Before we state Lemma 2.3.3, we need a couple of definitions.

Definition 2.3.1. For a strongly connected digraph G and a non-negative vector $f \in \mathbb{Z}_+^V$, we call $\lceil \max_{v\in V} \frac{f(v)}{\mathrm{per}_G(v)} \rceil$ the *level* of f, and denote it by $\mathrm{lvl}(f)$. We call a sequence $f_1, \dots, f_{\mathrm{lvl}(f)}$ of vectors the *level vectors* of f if $\sum_{i=1}^{\mathrm{lvl}(f)} f_i = f$ and $f_i = (\sum_{j=1}^{i} f_j) \wedge \mathrm{per}_G$ for each $i \in \{1, \dots, \mathrm{lvl}(f)\}$.

We call a level vector f_i *trivial* if $f_i(v) \in \{0, \mathrm{per}_G(v)\}$ for each $v \in V(G)$. Otherwise we call it *nontrivial*.

Let us give some intuition to the notion of level vectors. In the case if G is Eulerian, $\mathrm{per}_G = \mathbf{1}_G$. Hence in this case f_i is a vector that is 1 on the vertices that occur at least $\mathrm{lvl}(f) - i + 1$ times in f, and 0 otherwise.

Claim 2.3.2. *For any strongly connected digraph G and non-negative vector $f \in \mathbb{Z}_+^V$, the sequence of level vectors of f exists and is unique.*

Proof. We use induction on $\mathrm{lvl}(f)$. If $\mathrm{lvl}(f) = 0$, that means that $f \equiv 0$. Then the empty sequence is the unique solution for the conditions of Definition 2.3.1.

If $\mathrm{lvl}(f) \geq 1$, then the last level vector needs to be $f \wedge \mathrm{per}_G$. Take $f' = f - (f \wedge \mathrm{per}_G)$. It is easy to check that $\mathrm{lvl}(f') = \mathrm{lvl}(f) - 1$, hence by the induction hypothesis, f' has a unique sequence of level vectors $f'_1, \dots f'_{\mathrm{lvl}(f)-1}$. Taking $f_{\mathrm{lvl}(f)} = f \wedge \mathrm{per}_G$, the sequence $f'_1, \dots f'_{\mathrm{lvl}(f)-1}, f_{\mathrm{lvl}(f)}$ satisfies the conditions of level vectors. Also, since we need to have $f_{\mathrm{lvl}(f)} = f \wedge \mathrm{per}_G$, and the sequence of vectors $f_1, \dots f_{\mathrm{lvl}(f)-1}$ needs to be a sequence of level vectors of $f - f_{\mathrm{lvl}(f)}$ by definition, this is the unique solution for the conditions of Definition 2.3.1. □

Lemma 2.3.3. *Let G be a strongly connected digraph. Let x be a chip-distribution on G and $f \in \mathbb{Z}_+^V$ a non-negative vector such that there exists a legal game from x with firing vector f. Let $f_1, \dots, f_{\mathrm{lvl}(f)}$ be the level vectors of f. Then there exists a sequence of legal firings $(v_1, v_2, \dots, v_s)$ from x with firing vector f, such that there exist indices $i_0 = 0, i_1, i_2, \dots i_{\mathrm{lvl}(f)} = s$ such that for each $j = 1, \dots, \mathrm{lvl}(f)$, the firing vector of the sequence $v_{i_{j-1}+1}, \dots, v_{i_j}$ is f_j.*

Proof. Lemma 2.2.3 plays a key role in this proof.

We use induction on $\mathrm{lvl}(f)$. If $\mathrm{lvl}(f) = 0$, then $f \equiv 0$, hence the empty firing sequence satisfies the conditions of the lemma.

Suppose that $\mathrm{lvl}(f) \geq 1$. By our assumption, from initial distribution x there exists a legal sequence of firings $\alpha = (w_1, \dots w_s)$ with firing vector f. From the definition of level vectors, it follows that $f_{\mathrm{lvl}(f)} = f \wedge \mathrm{per}_G$. We prove that from initial distribution x, there exists a legal sequence of firings α' with firing vector $f' = f - f_{\mathrm{lvl}(f)}$ that can be extended legally by a sequence β of firings with firing vector $f_{\mathrm{lvl}(f)}$. Indeed, by Lemma 2.2.3, the sequence of firings α' that we get from α by deleting the first $f_{\mathrm{lvl}(f)}(v)$ occurrence of each vertex v, is still legal. The firing vector of this sequence is $f - f_{\mathrm{lvl}(f)}$. Play the bounded chip-firing game with upper bound f from initial distribution x. Then α' is a valid beginning. As α is a legal chip-firing game with upper bound f, and its firing vector is f, by Lemma 2.2.2, each maximal bounded chip-firing game with upper bound f has firing vector f. Hence α' can be extended to a legal game with firing vector f. Let us call the sequence of the last $\deg(f_{\mathrm{lvl}(f)})$ firings β. The firing vector of β is necessarily $f_{\mathrm{lvl}(f)}$.

From the proof of Claim 2.3.2, we know that $\mathrm{lvl}(f') = \mathrm{lvl}(f) - 1$ and the level vectors of f' are $f_1, \dots f_{\mathrm{lvl}(f)-1}$. Hence by the induction hypothesis, there is a legal sequence $\gamma = (v_1, \dots, v_{s'})$ of firings with firing vector f' such that there exist indices $i_0 = 0, i_1, i_2, \dots i_{\mathrm{lvl}(f)-1} = s'$ such that for each $j = 1, \dots, \mathrm{lvl}(f) - 1$, the firing vector of the sequence $v_{i_{j-1}+1}, \dots v_{i_j}$ is f_j.

As α' can be legally extended by β, γ can also be legally extended by β, since the chip-distribution after a sequence of firings only depends on the firing vector, which is the same for γ and for α'. Hence setting $i_{\mathrm{lvl}(f)} = s' + \deg(f_{\mathrm{lvl}(f)})$, the sequence of firings γ followed by the sequence of firings β satisfies the conditions of the lemma. □

Remark 2.3.4. It is worth noting that the condition $f_i = (\sum_{j=1}^{i} f_j) \wedge \mathrm{per}_G$ in the definition of level vectors ensures $f_1 \leq \dots \leq f_{\mathrm{lvl}(f)}$. Indeed, the f_i have nonnegative elements, thus $f_i = (\sum_{j=1}^{i} f_j) \wedge \mathrm{per}_G \leq (\sum_{j=1}^{i+1} f_j) \wedge \mathrm{per}_G$. In particular, for Eulerian digraphs, where $\mathrm{per}_G = \mathbf{1}_G$, the f_i are zero-one vectors, hence Lemma 2.3.3 implies that if $x \rightsquigarrow y$, then there is a legal game transforming x to y that fires “an ascending chain of sets of vertices”.

There are some lemmas of similar flavor, using 'ascending chains' in the related field of graph divisor theory, see for example [41, Lemma 1.3.] or the notion of 'level sets' in [42].

Note that even for a reduced vector f, $\mathrm{lvl}(f)$ can be exponentially large. For being able to use Lemma 2.3.3 in our algorithm deciding the reachability problem, we need to be able to manipulate the level vectors for a non-negative vector in polynomial time. We claim that even though there can be more than polynomially many level vectors, there are only polynomially many different ones, which enables us to compute the i^{th} level vector for given i in polynomial time.

Claim 2.3.5. *There is an algorithm that runs in $O(|V(G)|^2)$ time, and for a given vector $f \in \mathbb{Z}_+^V$, primitive period vector per_G and index $1 \leq a \leq \mathrm{lvl}(f)$ outputs the a^{th} level vector of f.*

Proof. Consider first Algorithm 1, which is a naive (and potentially exponential time) procedure. It is clear from the definition of level vectors, that the vectors f_i computed by Algorithm 1 are the level vectors of f. However, the running time of this algorithm is proportional to $\mathrm{lvl}(f)$, hence not polynomial in general. Algorithm 2 is an improved version of Algorithm 1, where we only compute the level vectors

Data: primitive period vector per_G and a non-negative vector $f \in \mathbb{Z}_+^V$
$i := \lceil \max_{v \in V} \frac{f(v)}{\mathrm{per}_G(v)} \rceil$;
$h := f$;
while $i \neq 0$ **do**
  $f_i := h \wedge \mathrm{per}_G$;
  $h := h - f_i$;
  $i := i - 1$;
end

Algorithm 1: Naive algorithm

for some "important" indices, and store these indices in an array I. To be able to output the level vector f_a for any given a, we need another query algorithm that, based on the data computed by Algorithm 2, gives us the requested f_a. This is done by Algorithm 3.

Claim 2.3.6. *Algorithm 3 produces correct answer.*

Proof. We claim that for those indices i that are stored in the array I, the vector f_i computed by Algorithm 2 is the i^{th} level vector of f. Also, we claim that for $I[k] \geq i > I[k+1]$, the i^{th} level vector of f is equal to $f_{I[k]}$. These two claims imply that Algorithm 3 produces correct answer.

Suppose that we are at some execution of the while loop of Algorithm 2, and so far the two claims are true for the computed vectors. Then for the present h and i, h is the sum of the first i level vectors of f. Hence indeed, the i^{th} level vector of f is equal to $h \wedge \mathrm{per}_G$. If there exists $v \in V(G)$ such that $0 < h(v) < \mathrm{per}_G(v)$, then $j = 0$, hence the second claim is meaningless. If the ratio $\frac{h(v)}{\mathrm{per}_G(v)}$ is either 0 or at least j on each vertex, then it is easy to see that the $i^{th}, i+1^{th}, \ldots, i+j-1^{th}$ level vectors of f are all equal to $h \wedge \mathrm{per}_G$, hence the two claims also hold in this case. □

Claim 2.3.7. *The while loop of Algorithm 2 is executed at most* $2|V(G)|$ *times.*

Proof. If in an execution there is a vertex v such that $0 < h(v) < \mathrm{per}_G(v)$, then we subtract $h \wedge \mathrm{per}_G$ from h, hence from the next iteration, $h(v) = 0$. If there is no such vertex, then let v be a vertex that minimizes $\lfloor \frac{h(v)}{\mathrm{per}_G(v)} \rfloor$. Then in the next iteration, $h(v) < \mathrm{per}_G(v)$, hence after one more iteration, $h(v) = 0$. Hence each vertex can be minimizer of the value $\lfloor \min_{v \in S} \frac{h(v)}{\mathrm{per}_G(v)} \rfloor$ at most twice. Note that after each execution of the while loop, $i = \mathrm{lvl}(h)$, hence while $i > 0$, $h \neq \mathbf{0}$. Hence there is a minimizer vertex in each execution of the while loop. Thus there can be at most $2|V(G)|$ executions of the while loop. □

Data: primitive period vector per_G and a non-negative vector $f \in \mathbb{Z}_+^V$
$i := \lceil \max_{v \in V} \frac{f(v)}{\mathrm{per}_G(v)} \rceil$;
$h := f$;
$k := 0$;
while $i \neq 0$ **do**
  $k := k + 1$;
  $I[k] := i$;
  $S := \{v \in V(G) : h(v) \neq 0\}$;
  $j := \lfloor \min_{v \in S} \frac{h(v)}{\mathrm{per}_G(v)} \rfloor$;
  $f_i := h \wedge \mathrm{per}_G$;
  if $j \neq 0$ **then**
    $h := h - j \cdot f_i$;
    $i := i - j$;
  else
    $h := h - f_i$;
    $i := i - 1$;
  end
end
$\ell := k$;

Algorithm 2: Preprocessing algorithm

An execution of the while loop of Algorithm 2 takes $O(|V(G)|)$ time, hence altogether Algorithm 2 runs in $O(|V(G)|^2)$ time. The fact that the while loop of Algorithm 2 is executed at most $2|V(G)|$ times implies that the array I stores at most $2|V(G)|$ elements. In other words, $\ell \leq 2|V(G)|$. Hence Algorithm 3 runs in $O(|V(G)|)$ time. □

From the proof of Claim 2.3.7 we can also deduce the following corollary.

Corollary 2.3.8. *For a strongly connected digraph G, among the level vectors of a non-negative vector $f \in \mathbb{Z}_+^V$, there are at most $|V(G)|$ types of different trivial level vectors, and at most $|V(G)|$ different nontrivial level vectors.*

The following theorem gives the main part of our algorithm. We analyze its complexity in the arithmetic model, i.e. we count the elementary arithmetic operations (addition, subtraction, multiplication, division, comparison) as one step.

Theorem 2.3.9. *Given a strongly connected digraph G, its primitive period vector per_G, a chip-distribution $x \in \mathrm{Chip}(G)$ and a non-negative vector $f \in \mathbb{Z}_+^V$, it can be decided in $O(|V(G)|^2(|V(G)| + \mathrm{per}(G)))$ steps (in the arithmetic model) whether*

Data: an index $1 \leq a \leq \text{lvl}(f)$ and the data computed by Algorithm 2
$k := 1$;
while $k \leq \ell$ **do**
  if $I[k] < a$ **then**
    BREAK;
  end
  $k := k + 1$;
end
Result: $f_{I[k-1]}$

Algorithm 3: Query algorithm

there exists a legal chip-firing game with firing vector f from initial distribution x. The algorithm uses polynomial space in the size of the input.

Proof. The idea of the proof is the following: By Lemma 2.3.3, f can be legally fired from initial distribution x if and only if there is a legal game from x that fires the sequence of level vectors of f. The main idea is that though there might be exponentially many level vectors, it is enough to check for each type of level vector, whether it can be fired at its last occurrence.

Let us write this formally. The algorithm is the following:

Run Algorithm 2 with input f. Let $a_i = I[\ell - i + 1]$ for $i \in \{1, \ldots, \ell\}$, and let $a_0 = 0$. This means that if $a_{i-1} < j \leq a_i$, then $f_j = f_{a_i}$.

Now let $x_0 = x$ and define $x_j = x + \sum_{k=1}^{j} L f_k$ for $j = 1, \ldots, \text{lvl}(f)$. We do not compute all of these chip-distributions (as there can be exponentially many), but note that for a fixed j, x_j can be computed in polynomial time: If $a_{i-1} < j \leq a_i$ then

$$x_j = x + L \left((j - a_{i-1}) f_{a_i} + \sum_{k=1}^{i-1} (a_k - a_{k-1}) f_{a_k} \right).$$

Now the algorithm proceeds as follows: For each $1 \leq i \leq \ell$ compute $x_{a_i - 1}$ and check whether the firing vector f_{a_i} can be fired from initial distribution x_{a_i-1}. By Lemma 2.2.2, we can check this greedily. If the firing vector f_{a_i} can be fired from initial distribution x_{a_i-1} for each $1 \leq i \leq \ell$, then the algorithm returns YES, otherwise the algorithm returns NO.

Let us compute the running time of the algorithm. We get the firing vectors $f_{a_1}, \ldots, f_{a_\ell}$ in $O(|V(G)|^2)$ time by running Algorithm 2. As $\ell \leq 2|V(G)|$, for a given i, x_{a_i-1} can be computed in $O(|V(G)|^2)$ time. We compute x_{a_i-1} for ℓ different values of i. This means altogether $O(|V(G)|^3)$ steps. By Lemma 2.2.2, we can check greedily whether the firing vector f_{a_i} can be fired from initial distribution x_{a_i-1}. At any point, we can check in $O(|V(G)|)$ time whether there exists a vertex that can be fired. If we find a vertex that can be fired, the effect of a firing can be computed in $O(|V(G)|)$ time. We need to do at most $\deg(f_{a_i})$ firings. As $f_{a_i} \le \mathrm{per}_G$, we have $\deg(f_{a_i}) \le \mathrm{per}(G)$. Hence for a given i, we can check in $O(|V(G)|\mathrm{per}(G))$ time whether the firing vector f_{a_i} can be fired from initial distribution x_{a_i-1}. We need to do this for ℓ values of i, which means altogether $O(|V(G)|^2\mathrm{per}(G))$ time. It is also clear that the algorithm uses polynomial space in the size of the input.

Now we prove the correctness of the algorithm. First we prove that if the algorithm returns YES, then indeed there is a legal chip-firing game from initial distribution x with firing vector f.

Note that $f = \sum_{j=1}^{\mathrm{lvl}(f)} f_j$. Thus for proving that f can be fired from initial distribution x, it is enough to prove for each $1 \le j \le \mathrm{lvl}(f)$ that f_j can be fired from initial distribution x_{j-1}. Let $1 \le j \le \mathrm{lvl}(f)$. Then for some $i \le \ell$, $a_{i-1} < j \le a_i$. Hence $f_j = f_{a_i}$.

Since the algorithm returned YES, f_{a_i} can be fired from initial distribution x_{a_i-1}. If f_{a_i} is a nontrivial level vector, then necessarily $j = a_i$, hence indeed f_j can be fired from x_{j-1}.

If f_{a_i} is a trivial level vector, let β be a legal game from initial distribution x_{a_i-1} with firing vector f_{a_i}. We prove that β is also a legal game starting from the distribution x_{j-1}. For this, it is enough to show that $x_{j-1}(v) \ge x_{a_i-1}(v)$ for each $v \in V(G)$ such that $f_{a_i}(v) > 0$.

We have $x_{a_i-1} = x_{j-1} + (a_i - j) \cdot Lf_{a_i}$. From the fact that f_{a_i} is a trivial level vector, each vertex v has either $f_{a_i}(v) = 0$ or $f_{a_i}(v) = \mathrm{per}_G(v)$. For those vetices, where $f_{a_i}(v) = \mathrm{per}_G(v)$, $(Lf_{a_i})(v) \le 0$, since

$$(Lf_{a_i})(v) = \sum_{u\in\Gamma^-(v)} f_{a_i}(u)\overrightarrow{d}(u,v) - d^+(v)f_{a_i}(v) =$$

$$\sum_{u\in\Gamma^-(v)} f_{a_i}(u)\overrightarrow{d}(u,v) - d^+(v)\mathrm{per}_G(v) \le$$

$$\sum_{u\in\Gamma^-(v)} \mathrm{per}_G(u)\overrightarrow{d}(u,v) - d^+(v)\mathrm{per}_G(v) = 0.$$

Hence for any vertex v where $f_{a_i}(v) > 0$, $x_{a_i-1}(v) \le x_{j-1}(v)$. Thus indeed f_j can be fired from initial distribution x_{j-1} for every $j \le \mathrm{lvl}(f)$. Hence f can be fired from initial distribution x.

Now it remains to show that if there exists a legal game with firing vector f from initial distribution x, then the algorithm returns YES. Suppose that f can be fired from initial distribution x. Take the legal game $(v_1, v_2, \dots, v_s)$ from initial distribution x with firing vector f provided by Lemma 2.3.3. By definition, after firing $(v_1, \dots v_{i_{j-1}})$ from initial distribution x, we arrive at x_{j-1}. The firing vector of the sequence $(v_{i_{j-1}+1}, \dots v_{i_j})$ is f_j by definition, and also by definition, this part of the game is also legal. Hence f_j can be fired from initial distribution x_{j-1} for any

$1 \le j \le \mathrm{lvl}(f)$. In particular, f_{a_i} can be fired from x_{a_i-1} for each $1 \le i \le \ell$, hence the algorithm returns YES.

□

Theorem 2.3.10. *Let G be an Eulerian digraph, and $x, y \in \mathrm{Chip}(G)$. Then it can be decided in strongly polynomial time whether $x \rightsquigarrow y$.*

Proof. By Claim 2.2.6, we can check in strongly polynomial time whether a reduced vector f exists such that $y = x + Lf$, and if it exists, compute it. By Corollary 2.2.5, $x \rightsquigarrow y$ if and only if such an f exists, and there exists a legal game from initial distribution x with firing vector f. If G is Eulerian, it is necessarily strongly connected. Thus by Theorem 2.3.9, given f, whether there exists a legal game from initial distribution x with firing vector f, can be decided in $O(|V(G)|^2(|V(G)| + \mathrm{per}(G))) = O(|V(G)|^3)$ time. As f can be computed in strongly polynomial time, its size is necessarily polynomial in the size of the description of x, y and G. As the algorithm of Theorem 2.3.9 uses polynomial space in the size of x, f and G, it also uses polynomial space in the size of x, y and G. □

Now we generalize Theorem 2.3.9 to general digraphs.

Theorem 2.3.11. *Given a digraph G, the primitive period vectors of its strongly connected components, a chip-distribution $x \in \mathrm{Chip}(G)$ and a non-negative vector $f \in \mathbb{Z}_+^V$, it can be decided in $O(|V(G)|^2(|V(G)| + \mathrm{per}(G)))$ steps (in the arithmetic model) whether there exists a legal chip-firing game with firing vector f from initial distribution x. The algorithm uses polynomial space in the size of the input.*

Proof. We proceed by induction on the number of strongly connected components of G. If G has one strongly connected component, i.e. it is strongly connected, then we are ready by Theorem 2.3.9.

Now suppose that G is not strongly connected. If G is not weakly connected, let $G[V_1]$ and $G[V_2]$ be two weakly connected components such that $V = V_1 \cup V_2$. f can be legally fired from initial distribution x on G if and only if $f|_{V_i}$ can be legally fired from initial distribution $x|_{V_i}$ on $G[V_i]$ for $i = 1, 2$. Moreover, both $G[V_1]$ and $G[V_2]$ has less number of strongly connected components than G, hence by induction hypothesis, we can decide in $O(|V_1|^2(|V_1| + \mathrm{per}(G[V_1])) + |V_2|^2(|V_2| + \mathrm{per}(G[V_2]))) = O(|V(G)|^2(|V(G)| + \mathrm{per}(G)))$ time whether both legal games exist, and the algorithm uses polynomial space in the input size.

If G is weakly connected, take a source component of G (strongly connected component that has no ingoing edge from any other strongly connected component). Let the vertex set of this source component be V_0. Contract the vertices in $V - V_0$ to a point, and call the contracted point v_0. Let the obtained graph be G'. Let x' be the following chip-distribution on G'.

$$x'(u) = \begin{cases} x(u) & \text{if } u \in V_0 \\ 0 & \text{if } u = v_0, \end{cases}$$

Let

$$f'(u) = \begin{cases} f(u) & \text{if } u \in V_0 \\ 0 & \text{if } u = v_0. \end{cases}$$

Claim 2.3.12. *If f can be legally fired from initial distribution x on G, then f' can be legally fired from initial distribution x' on G'.*

Proof. If f can be legally fired from initial distribution x on G, then we can suppose that all the firings of the vertices in V_0 happen before the firings of the vertices in $V - V_0$, since the firings of vertices from $V - V_0$ do not modify the number of chips on V_0, and the firings of vertices from V_0 can only increase the number of chips on vertices from $V - V_0$.

Hence from initial distribution x, there exists a legal game that fires only vertices in V_0, and fires each $v \in V_0$ exactly $f(v)$ times. If we play the same game on G' from initial distribution x', it remains legal, since the number of chips will be the same on each vertex of V_0 after each step. Moreover, this is a game with firing vector f'. □

Now for each vertex $u \in V_0$, let us draw an edge of multiplicity $\mathrm{per}_{G[V_0]}(u) \cdot \overrightarrow{d}_{G'}(u, v_0)$ from v_0 to u. Here we use the convention that non-edges have multiplicity zero. Call the obtained graph G''. Since $G'[V_0]$ is strongly connected and v_0 has at least one in-edge in G', now G'' is strongly connected. We claim that $\mathrm{per}_{G''}$ equals the following vector

$$p(u) = \begin{cases} \mathrm{per}_{G[V_0]}(u) & \text{if } u \in V_0 \\ 1 & \text{if } u = v_0. \end{cases}$$

This is a vector with nonnegative integer coordinates, and the largest common divisor of its elements is one, hence it is enough to check that $L_{G''}p = \mathbf{0}$, which is a straightforward calculation. Hence $\mathrm{per}(G'') = \mathrm{per}(G[V_0]) + 1$.

There exists a legal game on G'' from initial distribution x' with firing vector f' if and only if it exists on G', since $f'(v_0) = 0$.

By Theorem 2.3.9, we can decide in $O(|V_0|^2(|V_0| + \mathrm{per}(G''))) = O(|V_0|^2(|V_0| + \mathrm{per}(G[V_0])))$ time if there exists a legal game on G'' from initial distribution x' with firing vector f'. If there exists no such legal game, then there is no legal game with firing vector f from initial distribution x on G.

If there exists such a game, let f'' be equal to f on V_0 and zero on $V - V_0$. Let $\tilde{x} = (x + Lf'')|_{V-V_0}$ and $\tilde{f} = f|_{V-V_0}$. Since in the case if the firing vector f can be legally fired from initial distribution x, we can suppose that all the firings of the vertices in V_0 happen before the firings of the vertices in $V - V_0$, we conclude that in this case there is a legal game with firing vector $\tilde{f}$ from initial distribution $\tilde{x}$ on $G[V - V_0]$.

Moreover, we claim that if f' can be legally fired from x' on G' and $\tilde{f}$ can be legally fired from $\tilde{x}$ on $G[V - V_0]$, then f can be legally fired from x on G. Let α be the legal game with firing vector f' on G' (i.e. α is a sequence of vertices from

V_0), and β be the legal game with firing vector $\tilde{f}$ on $G[V - V_0]$ (i.e. β is a sequence of vertices from $V - V_0$). α is also a legal game if it is played on G with initial distribution x, and it leads to a chip-distribution that agrees with $\tilde{x}$ on $V - V_0$. If we continue with β, this is still a legal game, since in β only vertices of $V - V_0$ are fired, and on these vertices, our chip-distribution agrees with $\tilde{x}$. The firing vector of this game is f.

Hence it is enough to decide if $\tilde{f}$ can be legally fired from $\tilde{x}$ on $G[V-V_0]$. $G[V-V_0]$ has one less strongly connected components than G, hence by induction hypothesis, we can decide this in $O(|V - V_0|^2(|V - V_0| + \mathrm{per}(G[V - V_0])))$ time and polynomial space. Hence altogether, we can give an answer in $O(|V(G)|^2(|V(G)| + \mathrm{per}(G)))$ time, and we need polynomial space. □

Theorem 2.3.13. *Let G be a digraph, and $x, y \in \mathrm{Chip}(G)$. There is an algorithm that decides whether $x \rightsquigarrow y$, and has a running time which is a polynomial of the input size and the period length of G.*

Proof. The strongly connected components of G can be found in polynomial time [38]. By [19, Theorem 1.4.21], the primitive period vectors of the strongly connected components of G can be computed in polynomial time in the size of the description of G.

By Claim 2.2.6, we can decide in polynomial time in the input size whether there exists a reduced vector f such that $y = x + Lf$, and if the answer is yes, compute it. Again, if there exists no such f, then $x \not\rightsquigarrow y$. Now suppose that f exists. Then by Corollary 2.2.5, $x \rightsquigarrow y$ if and only if there exists a legal game from initial distribution x with firing vector f. By Theorem 2.3.11, this can be decided in a running time that is a polynomial of the size of the input and the period length of G. □

2.4 General digraphs

The algorithm of Section 2.3 is not polynomial for digraphs with exponentially large period length. It is conjectured by Björner and Lovász in [7] that the reachability problem is **NP**-hard for general digraphs. In this section, we give two positive results: We show that the reachability problem is in **co-NP**, and we show a special case when it is decidable in polynomial time for general digraphs.

2.4.1 The reachability problem is in co-NP

Theorem 2.4.1. *Let G be a digraph (with possibly multiple edges) and $x, y \in$ $\mathrm{Chip}(G)$. Then deciding whether $x \rightsquigarrow y$ is in* **co-NP**.

Proof. As we noted in Section 2.2, the existence of a reduced $f \in \mathbb{Z}_+^V$ such that $y = x + Lf$ is a necessary condition for $x \rightsquigarrow y$, that can be checked in polynomial time. Hence in case there exists no reduced $f \in \mathbb{Z}_+^V$ such that $y = x + Lf$, our certificate for $x \not\rightsquigarrow y$ is simply the statement that there exists no reduced $f \in \mathbb{Z}_+^V$ such that $y = x + Lf$.

In case there exists a reduced $f \in \mathbb{Z}_+^V$ such that $y = x + Lf$, our certificate is a pair of vectors $f, g \in \mathbb{Z}^V$ satisfying the following properties.

1. $y = x + Lf$, and f is reduced;

2. $\mathbf{0} \le g \le f$, and there exists $v \in V$ such that $g(v) < f(v)$;

3. For any $v \in V$, $g(v) = f(v)$ or $x_g(v) < d^+(v)$, where $x_g = x + Lg$.

All three conditions can be checked in polynomial time. Also, f has polynomially large coordinates in the input size since it could be computed in polynomial time. Hence g also has polynomially large coordinates.

We claim that if $x \not\rightsquigarrow y$ and there exists a reduced $f \in \mathbb{Z}_+^V$ such that $y = x + Lf$ then such f and g exist. Indeed, let f be the reduced firing vector that exists by asumption. Let g be the firing vector of a maximal bounded chip-firing game from initial distribution x with upper bound f. By Lemma 2.2.2, g is well defined. By the definition of the bounded game, $\mathbf{0} \le g \le f$. If $g = f$ then $x \rightsquigarrow y$, hence if $x \not\rightsquigarrow y$, then necessarily there exists $v \in V$ such that $g(v) < f(v)$. The third condition follows because g is the firing vector of a maximal game with upper bound f.

Now we prove that if such an f and g exist then $x \not\rightsquigarrow y$. Suppose for a contradiction that $x \rightsquigarrow y$. By Lemma 2.2.4, there exists a legal sequence $(v_1, v_2, \ldots, v_t)$ of firings with firing vector f that leads from x to y. Let j be the largest index such that $\sum_{i=1}^{j} \mathbf{1}_{v_i} \le g$. Let h be the firing vector of the sequence $(v_1, v_2, \ldots, v_j)$ and let $x_h = x + Lh$. By the choice of j, $g \ge h$ and $g(v_{j+1}) = h(v_{j+1}) < f(v_{j+1})$. Hence

$$x_g(v_{j+1}) - x_h(v_{j+1}) = L(g - h)(v_{j+1}) \ge 0.$$

Since $(v_1, v_2, \ldots, v_j, v_{j+1})$ is a legal sequence of firings, we get

$$d^+(v_{j+1}) \le x_h(v_{j+1}) \le x_g(v_{j+1}),$$

contradicting Condition 3. □

2.4.2 Reachability of recurrent distributions

In this section, we show a case when the reachability problem can be decided in polynomial time also for general digraphs. More exactly, we give a case where the necessary condition of the existence of a reduced vector f such that $y = x+Lf$ is also sufficient for $x \rightsquigarrow y$. Our theorem uses the notion of recurrent chip-distributions.

Definition 2.4.2. We call a chip-distribution $x \in \mathrm{Chip}(G)$ *recurrent* if there exists a non-empty sequence of legal firings that transforms x to itself.

Theorem 2.4.3. *Let G be a strongly connected digraph and $x, y \in \mathrm{Chip}(G)$. If y is recurrent and there exists a reduced f such that $y = x + Lf$, then $x \rightsquigarrow y$.*

Remark 2.4.4. For a strongly connected digraph, the existence of a reduced vector f such that $y = x + Lf$ is equivalent to $x \sim y$. This is true because for strongly connected digraphs, the primitive period vector is strictly positive on every coordinate, therefore the linear equivalence of x and y implies also the existence of a non-negative g such that $y = x + Lg$.

Proof of Theorem 2.4.3. First we claim that if there exists a reduced $f \in \mathbb{Z}_+^V$ such that $y = x + Lf$ then there exists a reduced $g \in \mathbb{Z}_+^V$ such that $x = y + Lg$. Indeed, since G is strongly connected, by Proposition 1.1.4 per_G is strictly positive on each coordinate. Hence we get such a g as $-f + c \cdot \mathrm{per}_G$ for an appropriate choice of c.

We proceed by induction on $\sum_{v \in V(G)} g(v)$. If $\sum_{v \in V(G)} g(v) = 0$, then $x = y$, thus $x \rightsquigarrow y$. Now suppose $\sum_{v \in V(G)} g(v) > 0$. As y is recurrent, there exists a sequence $(v_1, v_2, \ldots, v_k)$ of legal firings from initial distribution y (a vertex may occur multiple times), that leads back to y. Fix such a sequence. We claim that in this sequence, each vertex occurs at least once. Indeed, for the firing vector h of the game, $y = y + Lh$ thus h is a multiple of per_G, which is positive on each coordinate.

Let i be the smallest index such that $g(v_i) > 0$. Such an index exists because each vertex is listed at least once in $v_1, v_2, \ldots, v_k$. From initial distribution y, fire the vertices $v_1, \ldots, v_{i-1}$. This is a legal game by definition. Let the resulting distribution be y'. We claim that the sequence of firings $v_1, \ldots, v_{i-1}$ is also legal from initial distribution x. To prove this, it is enough to show that $x(v_j) \geq y(v_j)$ for all $1 \leq j \leq i-1$. This is true, because $x(v_j) = y(v_j) + (Lg)(v_j)$, where $(Lg)(v_j) \geq 0$, since the only negative element in the row corresponding to v_j is $L(v_j, v_j)$, but $g(v_j) = 0$. Hence the firing of the vertices $v_1, \ldots, v_{i-1}$ from initial distribution x is legal. Let the distribution obtained by this game be x'. Thus $x \rightsquigarrow x'$.

For x' and y', we also have $x' = y' + Lg$. At position y', firing v_i is legal, by definition of the sequence $v_1, \ldots, v_k$. Denote by y'' the distribution we get by firing v_i at y'. The distribution y'' is recurrent, since firing $v_{i+1} \ldots, v_k, v_1, \ldots, v_i$ is a legal game that leads back to y''. Now for x' and y'' we have $x' = y'' + Lg'$, where $g' = g - \mathbf{1}_{v_i}$. This way $\sum_{v \in V(G)} g'(v) = \sum_{v \in V(G)} g(v) - 1$, hence by the induction hypothesis, $x' \rightsquigarrow y''$.

We claim that $y'' \rightsquigarrow y$. Indeed, firing $v_{i+1}, \ldots, v_k$ starting from y'' is a legal game that leads to y. We also have $x \rightsquigarrow x'$. By transitivity, we have $x \rightsquigarrow y$. □

This theorem raises the question of the complexity of deciding whether a given chip-distribution is recurrent. By results of Björner and Lovász (Lemmas 2.2.2 and

2.2.3), a chip-distribution x is recurrent if and only if there exists a primitive period vector p, such that started from x, the maximal chip-firing game with upper bound p has firing vector p. For Eulerian digraphs, this can be checked in polynomial time (even if the digraph has multiple edges). However, for general digraphs, the complexity of deciding recurrence is open.

Our aim is now to generalize Theorem 2.4.3 for weakly connected digraphs. Here, the condition of y being recurrent is not enough to make the necessary condition sufficient.We show this by an example at the end of this section (Example 2.4.7). With a somewhat stronger condition, however, we can generalize Theorem 2.4.3 to weakly connected digraphs.

Theorem 2.4.5. *Let G be a weakly connected digraph, and $x, y \in \mathrm{Chip}(G)$ be two chip-distributions such that there exists a reduced $f \in \mathbb{Z}_+^V$ with $y = x + Lf$. Suppose that for each strongly connected component $G' = (V', E')$ of G, $f|_{V'} = \mathbf{0}$ or $y|_{V'} \in \mathrm{Chip}(G')$ is recurrent. Then $x \rightsquigarrow y$.*

Proof. Let $V_1, V_2, \ldots, V_k$ be a topological ordering of the strongly connected components of G, i.e., $V = V_1 \cup \cdots \cup V_k$, for each i the digraph $G_i = (V_i, E|_{V_i \times V_i})$ is strongly connected, and there is no directed edge from $v_i \in V_i$ to $v_j \in V_j$ if $i > j$.

Let x' be the chip-distribution obtained from x by passing $f(u) \cdot \overrightarrow{d}(u, v)$ chips from u to v for each pair of vertices $u, v \in V$ where u and v are in different strongly connected components. Note that $x \not\rightsquigarrow x'$ is possible. The proof of the theorem is based on the following lemma.

Lemma 2.4.6. *For each i, $x'|_{V_i} \sim y|_{V_i}$ on the digraph G_i. Moreover, if $y|_{V_i}$ is recurrent on G_i, then there exists a legal game on G_i with firing vector $f|_{V_i}$ that transforms $x'|_{V_i}$ to $y|_{V_i}$.*

Proof. Let L_i be the Laplacian matrix of G_i. We first prove that $x'|_{V_i} \sim y|_{V_i}$ (as chip-distributions on G_i) by showing that $x'|_{V_i} + L_i f|_{V_i} = y|_{V_i}$. For this, let $v \in V_i$. Then

$$\begin{aligned} x'(v) + (L_i f|_{V_i})(v) &= \\ x(v) + \sum_{v' \in V \setminus V_i} \left(\overrightarrow{d}(v', v) \cdot f(v') - \overrightarrow{d}(v, v') \cdot f(v) \right) + (L_i f|_{V_i})(v) &= \\ x(v) + (Lf)(v) = y(v). \end{aligned}$$

Now, if $y|_{V_i}$ is recurrent, then by Remark 2.4.4 we can apply Theorem 2.4.3, hence $x'|_{V_i} \rightsquigarrow y|_{V_i}$ on G_i. Let $g_i \in \mathbb{Z}^{V_i}$ be the firing vector of a legal game transforming $x'|_{V_i}$ to $y|_{V_i}$. Then $L_i(f|_{V_i} - g_i) = 0$, hence by Proposition 1.1.4, $g_i - f|_{V_i} = c \cdot p_{G_i}$ with $c \in \mathbb{Z}$. If $c = 0$ then $f|_{V_i}$ is the firing vector of a legal game, proving the lemma. In the followings, we treat separately the cases $c < 0$ and $c > 0$.

Suppose that $c < 0$. Since $y|_{V_i}$ is recurrent, there is a legal game on G_i that transforms $y|_{V_i}$ back to itself. For the firing vector g of this game, $L_i g = 0$, hence $g = \lambda \cdot p_{G_i}$ with $\lambda \in \mathbb{Z}$, $\lambda > 0$. By Lemma 2.2.3, we can suppose that $\lambda = 1$. Now starting from distribution $x'|_{V_i}$ on G_i, after playing the legal game with firing vector g_i, we get to the distribution $y|_{V_i}$. Then iterate $-c$ times the legal game with firing vector p_{G_i}. This gives us a legal game with firing vector $f|_{V_i}$, finishing the proof for the $c < 0$ case.

Now suppose that $c > 0$. Then Lemma 2.2.3 guarantees that there is a legal game from $x'|_{V_i}$ with firing vector $g_i - c \cdot p_{G_i} = f|_{V_i}$. This finishes the proof of the lemma. □

For each $1 \leq i \leq k$ let f_i be the vector with $f_i(v) = f(v)$ if $v \in V_i$, and $f_i(v) = 0$ otherwise. Let $s_i = \sum_{j \leq i} f_j$, i.e., $s_i(v) = f(v)$ if $v \in \bigcup_{j \leq i} V_j$, and $s_i(v) = 0$ otherwise. Let $x_i = x + L s_i$ and $x_0 = x$. We show that for $i = 1, \ldots, k$, starting from the distribution x_{i-1}, there is a legal game on G with firing vector f_i. Since $x_{i-1} + L f_i = x_i$, and $x_k = y$, this is enough to finish the proof of the theorem.

So let i be fixed. It is easy to see that for each $v \in V_i$

$$x'(v) = x_{i-1}(v) - f(v) \cdot \sum_{v' \in V \setminus V_i} \overrightarrow{d}(v, v'). \tag{2.4.1}$$

If $f|_{V_i} = \mathbf{0}_{V_i}$, then $f_i = \mathbf{0}$, hence we have nothing to prove. If this is not the case, then $y|_{V_i}$ is recurrent by the assumptions. Using the lemma, from initial distribution $x'|_{V_i}$ there exists a legal game on G_i with firing vector $f|_{V_i}$. We claim that the same sequence of firings on G, with initial distribution x_{i-1} remains a legal game. Indeed, we can see from (2.4.1) that by playing the game on G from initial distribution x_{i-1}, at any moment we have a distribution that is greater or equal on V_i than the distribution we get by playing the game on G_i with initial distribution $x'|_{V_i}$. Hence there exists a legal game on G with initial distribution x_{i-1} and firing vector f_i. This finishes the proof of the theorem. □

Example 2.4.7. Now we give an example showing that Theorem 2.4.3 does not remain true for general digraphs, i.e. for general digraphs, the existence of a reduced vector f such that $y = x + Lf$ and y being recurrent is not sufficient for $x \rightsquigarrow y$.

Let G be the following digraph:

$$V(G) = \{v_1, v_2, v_3, v_4, v_5, v_6\}$$

$$E(G) = \{\overrightarrow{v_1v_2}, \overrightarrow{v_2v_1}, \overrightarrow{v_2v_3}, \overrightarrow{v_3v_2}, \overrightarrow{v_3v_4}, \overrightarrow{v_4v_3}, \overrightarrow{v_4v_1}, \overrightarrow{v_1v_4}, \overrightarrow{v_3v_5}, \overrightarrow{v_4v_6}, \overrightarrow{v_5v_6}, \overrightarrow{v_6v_5}\}$$

Let $x = (1, 1, 0, 0, 1, 0)$ and $y = (0, 0, 1, 1, 1, 0)$.

It is easy to see that y is recurrent, since firing v_5 then firing v_6 transforms it back to itself. Also, for the reduced $f = (1, 1, 0, 0, 0, 0)$, $y = x + Lf$. However, $x \not\rightsquigarrow y$, as for $x \rightsquigarrow y$ we need to be able to fire the firing vector f. However, neither v_1 nor v_2 can fire in x.

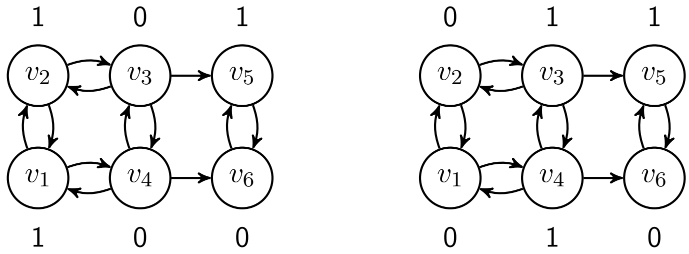

Figure 2.1: The chip-distributions x and y on G

2.5 Open questions and related problems

The most intriguing open question concerning the reachability problem is the complexity of the reachability problem on general digraphs. An interesting special case of this problem is deciding whether a chip-distribution on a general digraph is recurrent.

Problem 2.5.1. Let G be a digraph and $x, y \in \mathrm{Chip}(G)$. What is the complexity of deciding whether $x \rightsquigarrow y$?

Problem 2.5.2. Let G be a digraph. What is the complexity of deciding whether a chip-distribution $x \in \mathrm{Chip}(G)$ is recurrent?

We conjecture that both of these questions are **co**-**NP**-hard.

A related problem to the chip-firing reachability problem is the so-called chip-firing halting problem.

Chip-firing halting problem Given a digraph G and a chip-distribution $x \in \mathrm{Chip}(G)$, decide if x is terminating.

Informally, the halting problem and the reachability problem are both about determining the firing vector of a maximal game, only this game is a chip-firing game for the halting problem, and a bounded chip-firing game for the reachability problem.

The halting problem is known to be in **P** for simple Eulerian digraphs [7], and it is known to be **NP**-complete for general digraphs [16]. The complexity of the problem is open both for simple digraphs, and for Eulerian digraphs. We point out the following:

Proposition 2.5.3. *The chip-firing halting problem is in* **co**-**NP** *for Eulerian digraphs.*

Proof. Our certificate for "x is non-terminating" is a recurrent chip-distribution y such that $x \sim y$. Since the graph is Eulerian, both the fact that y is recurrent, and that $x \sim y$, can be checked in polynomial time.

We show that x is non-terminating if and only if such a y exists. In a chip-firing game a vertex can only loose chips if it is fired, but if it is fired, it is not allowed to go negative. Hence in a game with initial distribution x, on any vertex v, the number of chips is always at least $\min\{0, x(v)\}$. Hence the number of chips on any vertex is at most $\sum_{v \in V} \max\{x(v), 0\}$. Hence the number of chip-distributions reachable from x by a legal game is finite. Thus if x is non-terminating, starting a legal chip-firing game from x, we will eventually visit some chip-distribution y twice. This y is therefore recurrent. Moreover, $y \sim x$, since $y = x + Lf$ for the firing vector of the game leading from x to y.

For the other direction, we use Lemma 1.2.6, that states that for a strongly connected digraph G and $x, y \in \text{Chip}(G)$, if $x \sim y$, then x is terminating if and only if y is terminating. Note that now our graph is strongly connected since it is connected and Eulerian. Note also that a recurrent chip-distribution is always non-terminating, since we can repeat the nonempty legal game transforming it back to itself indefinitely. Hence y is non-terminating, consequently, x is non-terminating. □

This means, that for Eulerian digraphs, the chip-firing halting problem is in $\mathbf{NP} \cap \mathbf{co} - \mathbf{NP}$.

Problem 2.5.4. Is there a polynomial time algorithm that decides the chip-firing halting problem for Eulerian digraphs (with multiple edges possible)?

Chapter 3

The NP-hardness of computing the rank of a divisor on a graph

This chapter is based on the paper [26], which is joint work with Viktor Kiss. Proposition 3.2.2 and Theorem 3.2.8 are from [24], which is joint work with Bálint Hujter.

3.1 Introduction

In this chapter, we prove that computing the rank of a divisor on a graph is **NP**-hard, even for simple undirected graphs. The rank of a divisor is a central notion in the Riemann-Roch theory of graphs. The question whether the rank can be computed in polynomial time has been posed in several papers [20, 29, 6], originally attributed to H. Lenstra. Our result implies also the **NP**-hardness of computing the rank of a divisor on a tropical curve by [28, Theorem 1.6].

Let us say a few words about previous work concerning the computation of the rank. Hladký, Král' and Norine [20] gave a finite algorithm for computing the rank of a divisor on a metric graph. Manjunath [29] gave an algorithm for computing the rank of a divisor on a graph (possibly with multiple edges), that runs in polynomial time if the number of vertices of the graph is a constant. For simple graphs, it can be decided in polynomial time, whether the rank of a divisor is at least c, where c is a constant [6]. Computing the rank of a divisor on a complete graph can be done in polynomial time [11]. For divisors of degree greater than $2\mathfrak{g} - 2$ (where $\mathfrak{g}$ is the genus of the graph), the rank can be computed in polynomial time [29].

Our method for proving the **NP**-hardness of the computation of the rank is the following: Using the duality between chip-firing and graph divisor theory, we translate the question of computing the rank of a divisor on an undirected graph to the question of computing the distance from non-terminating distributions of a chip-distribution on an undirected graph. Based on ideas of Perrot and Pham [33], we first show that computing $\mathrm{dist}(\mathbf{0}_D)$ for an Eulerian digraph D is **NP**-hard. Then we show the **NP**-hardness of computing dist for a chip-distribution on an undirected graph by reducing to it the computation of $\mathrm{dist}(\mathbf{0}_D)$ for an Eulerian digraph D. In the reduction, we imitate the chip-firing game of an Eulerian digraph on an undirected graph.

In this chapter, we work both on directed and undirected graphs, and it will be important to distiguish them by notation. Hence in this chapter, G always means an undirected graph. On the other hand, directed graphs are denoted by D. We emphasise that by the term "graph", we mean undirected graph. Also, in this chapter, we think of undirected graphs in the ordinary way, not as special Eulerian digraphs. Hence for example $\sum_{v\in V(G)} d(v) = 2|E(G)|$ for a graph G.

3.2 Minimal non-terminating distributions on Eulerian digraphs

In this section, based on recent results of Perrot and Pham [33], we prove the following theorem.

Theorem 3.2.1. *Given a digraph D, computing $\mathrm{dist}(\mathbf{0}_D)$ is **NP**-hard, even for simple Eulerian digraphs.*

We use the method of Perrot and Pham. In the paper [33], they proved the **NP**-hardness of an analogous question in the Abelian Sandpile Model, which is a closely related variant of the chip-firing game.

Before proceeding, we need a technical lemma. This lemma appeared in [7], but since in [7] it is only proved for non-negative chip-distributions and we need it for integer valued distributions, we give a proof.

Proposition 3.2.2. *On a strongly connected digraph D, in any infinite legal game every vertex is fired infinitely often.*

Proof. In a chip-firing game a vertex can only loose chips if it is fired, but if it is fired, it is not allowed to go negative. Hence in a game with initial distribution x, on any vertex v, the number of chips is always at least $\min\{0, x(v)\}$. Hence the number of chips on any vertex is at most $\sum_{v\in V} \max\{x(v), 0\}$ at any time.

If a legal game is infinitely long, then there is a vertex that fires infinitely often. If a vertex is fired infinitely often, then it passes infinitely many chips to its out-neighbors, hence the out-neighbors also need to be fired infinitely often, otherwise they would have more chips than possible. By induction, every vertex reachable on directed path from an infinitely often fired vertex is also fired infinitely often. As the graph is strongly connected, each vertex is reachable on directed path from each vertex, thus every vertex fires infinitely often. □

From Proposition 3.2.2 it follows that if $x \in \mathrm{Chip}(D)$ is non-terminating, then playing a legal game, after finitely many steps, each vertex has already fired. At this time, we are at a distribution which is nowhere negative. Hence there exists a non-negative chip-distribution among the non-terminating distributions of minimum degree. This has the following corollary:

Claim 3.2.3. *For a strongly connected digraph D,* $\mathrm{dist}(\mathbf{0}_D)$ *equals to the minimum degree of a non-terminating distribution on D.*

Now we turn back to proving that the computation of $\mathrm{dist}(\mathbf{0}_D)$ is **NP**-hard on a simple Eulerian digraph D. Using the ideas of [33], we first give a formula for the minimum number of chips in a non-terminating distribution on an Eulerian digraph. As a motivation, let us have a look at the analogous question on undirected graphs, which was solved by Björner, Lovász and Shor.

Theorem 3.2.4 ([8, Theorem 2.3])**.** *Let G be an undirected graph. Then* $\mathrm{dist}(\mathbf{0}_G) = |E(G)|$.

We sketch the proof as a motivation for the directed case.

Proof. First we prove the following useful lemma.

Lemma 3.2.5 ([8])**.** *Let D be an acyclic orientation of G and let $x \in \mathrm{Chip}(G)$ be a distribution with $x(v) \geq d^-_D(v)$ for each $v \in V(G)$. Then x is non-terminating.*

Proof. Since the orientation is acyclic, there is a sink, i.e., a vertex $v_0 \in V(G)$ with $d(v_0) = d^-_D(v_0) \leq x(v_0)$. Hence v_0 is active with respect to x. Fire v_0 and denote the resulting distribution by x'. Reverse the direction of the edges incident to v_0 and denote the resulting directed graph by D'. It is easy to see that D' is acyclic and $d^-_{D'}(v) \leq x'(v)$ for each $v \in V(G)$. Hence we can repeat the above argument. This shows that the distribution x is indeed non-terminating. □

Now taking an acyclic orientation D of G and setting $x(v) = d^-_D(v)$ for each $v \in V(G)$ we have a distribution with $\deg(x) = |E(G)|$ that is non-terminating from the lemma. This shows that $\mathrm{dist}(\mathbf{0}_G) \leq |E(G)|$.

For proving $\mathrm{dist}(\mathbf{0}_G) \geq |E(G)|$, take a non-terminating distribution $x \in \mathrm{Chip}(G)$. It is enough to show that $\deg(x) \geq |E(G)|$. Since in a non-terminating game on a connected undirected graph every vertex is fired infinitely often (by Proposition 3.2.2), after finitely many firings, every vertex of G has been fired at least once. Let x' be the distribution at such a moment. Then $\deg(x) = \deg(x')$. Let D be the orientation of G that we get by directing each edge toward the vertex whose last firing occurred earlier. It is straightforward to check that $x'(v) \geq d_D^-(v)$ for each $v \in V(G)$. This fact implies that $\deg(x) = \deg(x') \geq |E(G)|$, completing the proof. $\square$

Now let us consider Eulerian digraphs.

Theorem 3.2.6. *Let D be an Eulerian digraph. Then* $\mathrm{dist}(\mathbf{0}_D) = \mathrm{minfas}(D)$.

This theorem is already stated in a note added in proof of [7], but there only the direction $\mathrm{dist}(\mathbf{0}_D) \geq \mathrm{minfas}(D)$ is proved. We give a proof following ideas of Perrot and Pham [33], who gave an analogous theorem for the Abelian Sandpile Model (which is a variant of the chip-firing game). The idea of the proof can be thought of as the generalization of the idea of the proof of Theorem 3.2.4. In Chapter 4, we will need a structure theorem on non-terminating chip-distributions on Eulerian digraphs, hence instead of Theorem 3.2.6, we prove a stronger structure theorem (Theorem 3.2.8). For this, we need the notion of a turnback arc set.

Definition 3.2.7. A *turnback arc set* of a digraph D is a set of edges $T \subseteq E(G)$ such that the digraph D' we get by reversing the edges in T is acyclic.

Note that any turnback arc set is also a feedback arc set. On the other hand, by a theorem of Gallai [17], each minimal feedback arc set is also a turnback arc set. Hence $\mathrm{minfas}(D)$ equals also to the cardinality of a minimum cardinality turnback arc set.

Theorem 3.2.8. *Let D be an Eulerian digraph. A chip-distribution $x \in \mathrm{Chip}(D)$ is non-terminating if and only if there exists a turnback arc set T of D, and a chip-distribution $a \in \mathrm{Chip}(D)$ with $a \geq 0$, such that if ϱ is the indegree distribution of T, then $x \sim \varrho + a$.*

For the proof of this theorem we need a lemma, which is a variant of a result of Perrot and Pham [33, Lemma 2.4.].

Lemma 3.2.9. *Let $T \subseteq E(D)$ be a turnback arc set. Denote by $d_T^+(v)$ and $d_T^-(v)$ the outdegree and indegree of a vertex v in the digraph $D_T = (V(D), T)$. Then a distribution $x \in \mathrm{Chip}(D)$ satisfying*

$$x(v) \geq d_T^-(v) \text{ for every } v \in V(D) \tag{3.2.1}$$

is non-terminating.

Proof. Let D' be the graph we get from D by reversing the edges of T. From the definition of turnback arc set, D' is an acyclic digraph. Therefore, it has a source v_0.

Hence no out-edge of v_0 in D is from T, but all the in-edges of v_0 in D are from T. From (3.2.1), the choice of v_0 and the fact that D is Eulerian, we have that $x(v_0) \geq d_T^-(v_0) = d^-(v_0) = d^+(v_0)$, therefore v_0 is active with respect to x. Fire v_0. Let x' be the resulting distribution. Let T' be the set of arcs obtained from T by removing the in-edges of v_0 and adding the out-edges of v_0. Then the graph D'' that we get by reversing the edges in T' is acyclic, since compared to D', we only transformed a source to be a sink. Hence T' is a turnback arc set. It is straightforward to check that $x'(v) \geq d_{T'}^-(v)$ for every $v \in V(D)$. Thus, we are again in the starting situation, which shows that x is indeed non-terminating. □

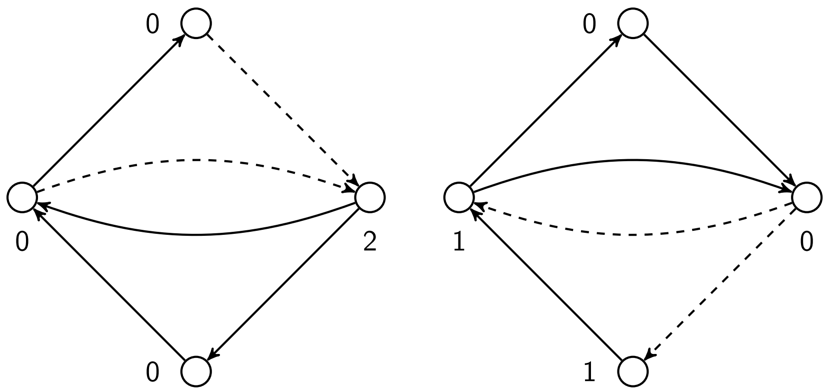

Figure 3.1: An example for simultaneously firing a vertex and changing the turnback arc set. The arcs of the turnback arc sets are drawn by dashed lines.

Proof of Theorem 3.2.8. Suppose that there is a chip-distribution $x \in \mathrm{Chip}(D)$ such that $x \sim \varrho + a$ where $\varrho \in \mathrm{Chip}(D)$ is the indegree sequence of a turnback arc set T of D, and $a \in \mathrm{Chip}(D)$ with $a \geq 0$. Then from the lemma, $\varrho + a$ is non-terminating. From Lemma 1.2.6, $x \sim \varrho + a$ is also non-terminating.

For the other direction, take a non-terminating distribution x. Let us play a chip-firing game with initial distribution x. Proposition 3.2.2 says that after finitely many steps, every vertex has fired. Play until such a moment, and let the distribution at that moment be x'. Then $x \sim x'$.

Let A be the following set of edges:

$$A = \{\overrightarrow{uv} \in E(D) : \text{ the last firing of } u \text{ precedes the last firing of } v\}.$$

As every vertex has fired, A is well defined. Let $v_1, v_2, \ldots v_{|V(D)|}$ be the ordering of the vertices by the time of their last firing.

Then $v_1, v_2, \dots v_{|V(D)|}$ is a topological order of the graph D' that we get by reversing the edges in $E(D) \setminus A$, hence $T = E(D) \setminus A$ is a turnback arc set. We show that $x'(v) \geq d_T^-(v)$ for every $v \in V(D)$. For $1 \leq i \leq |V(D)|$, the vertex v_i has $d_T^-(v_i) = \sum_{j>i} \overrightarrow{d}(v_j, v_i)$. After its last firing, v_i had a nonnegative number of chips. Since then, it kept all chips it received. And as $v_{i+1}, \dots, v_{|V(D)|}$ all fired since the last firing of v_i, it received at least $\sum_{j>i} \overrightarrow{d}(v_j, v_i) = d_T^-(v_i)$ chips. So indeed, we have $x'(v_i) \geq d_T^-(v_i)$.

Therefore $x' = \varrho + a$, where $\varrho(v) = d_T^-(v_i)$, and $a \geq 0$. Since $x \sim x'$, we are ready. □

Note that in the above setting, starting from x', then firing the vertices in the order $v_1, v_2, \dots, v_{|V(D)|}$ (once each) is a legal game. Indeed, we proved that $x'(v_i) \geq d_T^-(v_i) = \sum_{j>i} \overrightarrow{d}(v_j, v_i)$. After firing $v_1, \dots v_{i-1}$, the vertex v_i receives $\sum_{j<i} \overrightarrow{d}(v_j, v_i)$ more chips, so it indeed becomes active ($d^+(v_i) = d^-(v_i) = \sum_{j \neq i} \overrightarrow{d}(v_j, v_i)$) as we did not allow loops).

We need this observation in the next section, so we state it as a proposition:

Proposition 3.2.10. *In a chip-firing game on an Eulerian digraph D, if at some moment every vertex has already fired then there is an order of the vertices in which they can be legally fired once each, starting from that moment.* □

It is worth noting that on an Eulerian digraph, if starting from an initial distribution x we fired each vertex exactly once, then we get back to distribution x: each vertex v gave and received $d^-(v) = d^+(v)$ chips.

Proof of Theorem 3.2.6. From Claim 3.2.3, $\mathrm{dist}(\mathbf{0}_D)$ equals to the minimum degree of a non-terminating chip-distribution on D. By Theorem 3.2.8, this minimum degree equals to the cardinality of a minimum cardinality turnback arc set, which equals to $\mathrm{minfas}(D)$ since minimal feedback arc sets are turnback arc sets [17]. □

Finally, we prove Theorem 3.2.1.

Proof of Theorem 3.2.1. Perrot and Pham proved that computing $\mathrm{minfas}(D)$ for a simple Eulerian digraph D is **NP**-hard [33, Theorem 2], by reducing it to the **NP**-hardness of computing $\mathrm{minfas}(D)$ for general digraphs. From this, and from Theorem 3.2.6, the statement follows. □

3.3 NP-hardness of computing dist and rank

In this section we prove that computing the distance from non-terminating distributions of a chip-distribution is **NP**-hard, and deduce the **NP**-hardness of computing the rank of a divisor on a graph. In our proof of the **NP**-hardness, we rely on the fact that a terminating chip-firing game on an Eulerian digraph D, that is started from a chip-distribution which is non-negative on each vertex terminates after at most $2|V(D)|^2|E(D)|\Delta(D)$ steps (see [7, Corollary 4.9]). With this in mind, we define the following transformation:

Definition 3.3.1. Let φ be the following transformation, assigning an undirected graph $G = \varphi(D)$ to any digraph D:

Split each directed edge by an inner point, and substitute the tail segment by $M = 8|V(D)|^2|E(D)|\Delta(D)$ parallel edges. Then forget the orientations.

We maintain the effect of the transformation by a bijective function $\psi : (V(D) \cup E(D)) \to V(\varphi(D))$:

For a vertex $v \in V(D)$ let $\psi(v)$ be the corresponding vertex of $\varphi(D)$. For an edge $e \in E(D)$, let $\psi(e)$ be the vertex with which we have split e.

Then the degrees in $\varphi(D)$ are the following:

$$d(v) = \begin{cases} d^+(\psi^{-1}(v)) \cdot M + d^-(\psi^{-1}(v)) & \text{if } \psi^{-1}(v) \in V(D) \\ M+1 & \text{if } \psi^{-1}(v) \in E(D). \end{cases}$$

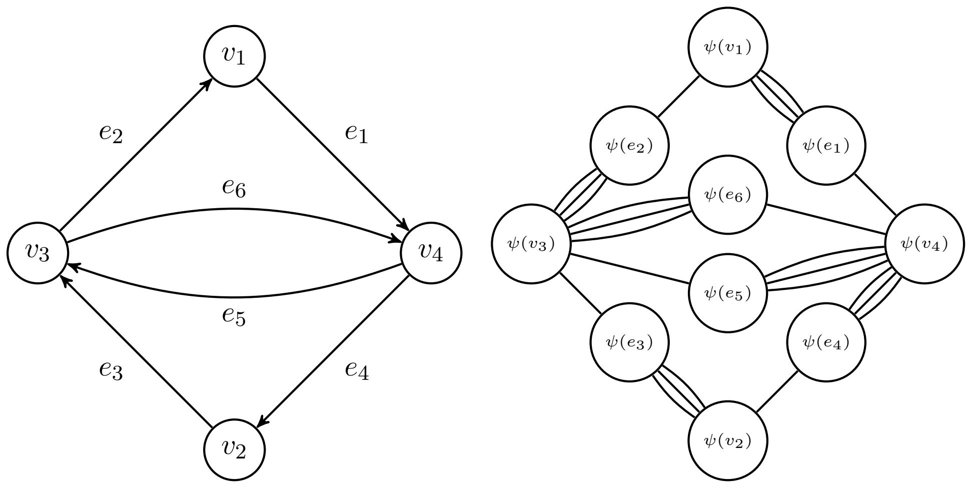

Figure 3.2: A schematic picture for a digraph D and the corresponding $\varphi(D)$. In the reality the multiple edges should be 1536-fold.

Let us define a certain chip-distribution on the graph $\varphi(D)$:

Definition 3.3.2 (base-distribution)**.** Let $base_D \in \text{Chip}(\varphi(D))$ on a vertex $v \in V(\varphi(D))$ be the following:

$$base_D(v) = \begin{cases} d^+(\psi^{-1}(v)) \cdot M & \text{if } \psi^{-1}(v) \in V(D) \\ M/2 & \text{if } \psi^{-1}(v) \in E(D). \end{cases}$$

The key lemma in our proof of Theorem 3.3.5 is the following:

Lemma 3.3.3. *For an Eulerian digraph D,* $\mathrm{dist}_D(\mathbf{0}_D) = \mathrm{dist}_{\varphi(D)}(base_D)$.

Proof. Let $G = \varphi(D)$. First we show that $\mathrm{dist}_D(\mathbf{0}_D) \geq \mathrm{dist}_G(base_D)$.

Let $x \in \mathrm{Chip}(D)$ be a non-terminating chip-distribution such that $\deg(x)$ is minimal. We can assume that there is an order of the vertices of D such that from initial distribution x we can fire the vertices in that order (once each). Otherwise, from Proposition 3.2.2 we can play a chip-firing game from x until each vertex has fired. Denoting the distribution at that moment by x', from Proposition 3.2.10 for x' there is such an order. As firing does not change the number of chips in the game, $\deg(x')$ is still minimal, so we can substitute x with x'. As x' is non-negative on each vertex, we can also suppose that x is non-negative on each vertex.

Let $y \in \mathrm{Chip}(G)$ be the distribution "$x + base_D$", i.e., for a vertex $v \in V(D)$ let $y(\psi(v)) = x(v) + base_D(\psi(v))$ and for an edge $e \in E(D)$ let $y(\psi(e)) = base_D(\psi(e))$. Since $y(w) \geq base_D(w)$ for each $w \in V(G)$ and $\deg(y - base_D) = \deg(x) = \mathrm{dist}_D(\mathbf{0}_D)$, it is enough to show that y is non-terminating.

For that, it is enough to show that we can fire each vertex of G exactly once in some order. Then each vertex $w \in V(G)$ gives and receives $d(w)$ chips, so we get back to the distribution y and can repeat this period indefinitely.

To get such an order of the vertices of G, we will play the chip-firing game simultaneously on D and G.

To firing a vertex v in D, let the corresponding firings in G be: Fire $\psi(v)$, then fire $\psi(e)$ for every out-edge e of v (in some order).

Claim 3.3.4. *If a sequence of firings of length $k \leq M/2$ on D with initial distribution x is legal then the sequence of the corresponding firings on G with initial distribution y is also legal. Moreover, if we denote the resulting distribution on D by $\tilde{x}$ and on G by $\tilde{y}$ then*

$$\tilde{y}(\psi(v)) = \tilde{x}(v) + d^+(v) \cdot M \text{ for each } v \in V(D) \tag{3.3.1}$$

and

$$M/2 - k \leq \tilde{y}(\psi(e)) \leq M/2 + k \text{ for each } e \in E(D). \tag{3.3.2}$$

Proof. We show this by induction on k. For $k = 0$ this is trivial. Take a sequence of firings of length $k \leq M/2$ and assume that the claim holds for $k-1$. Denote the distribution on D after the first $k-1$ firings by x' and the corresponding distribution on G by y'. Assume that the vertex v is the last to be fired on D. Hence v is active with respect to x'. Denote the distribution after firing v by x''. Vertex $\psi(v)$ is active with respect to y', since using (3.3.1) of the induction hypothesis, the fact that v is active with respect to x' and that D is Eulerian, we get that $y'(\psi(v)) = x'(v) + d^+(v) \cdot M \geq d^+(v) + d^+(v) \cdot M = d^-(v) + d^+(v) \cdot M = d(\psi(v))$. Fire $\psi(v)$. Now for each out-edge e of v the vertex $\psi(e)$ is active, since using (3.3.2) of the induction hypothesis, it has at least $M + y'(\psi(e)) \geq M + M/2 - (k-1) \geq M + 1 = d(\psi(e))$ chips. Fire these vertices in an arbitrary order. (Firing one leaves the others active.) Denote by y'' the resulting distribution. It is easy to check that the distributions x'' and y'' satisfy conditions (3.3.1) and (3.3.2). □

We have chosen the distribution x such that we can fire the vertices of D in some order (once each) with initial distribution x. This is a legal sequence of firings of length $|V(D)| < M/2$. According to the previous claim, the sequence of the corresponding firings on G is also legal. Moreover, on G we also fire each vertex exactly once. This finishes the proof of the direction $\mathrm{dist}_D(\mathbf{0}_D) \geq \mathrm{dist}_G(base_D)$.

Now we prove that $\mathrm{dist}_D(\mathbf{0}_D) \leq \mathrm{dist}_G(base_D)$. For this, let $y \in \mathrm{Chip}(G)$ be a minimal non-terminating chip-distribution with $base_D(w) \leq y(w)$ for each $w \in V(G)$. Let $x(v) = y(\psi(v)) - base_D(\psi(v))$ on each $v \in V(D)$. It is enough to show that x is non-terminating.

First note that $\mathrm{dist}_D(\mathbf{0}_D) \leq |E(D)| - |V(D)| + 1$, since having a chip-distribution with at least $|E(D)| - |V(D)| + 1$ chips, at every stage of the game at least one of the vertices has the sufficient number of chips to fire. Consequently, using also the first part of the lemma, we have that $\deg(y - base_D) = \mathrm{dist}_G(base_D) \leq \mathrm{dist}_D(\mathbf{0}_D) \leq |E(D)| - |V(D)| + 1 \leq \frac{1}{8}M$.

Now we play the game on G and D simultaneously from initial distributions y and x, respectively, in the following way. Let a step be the following: Choose a vertex $v \in V(D)$ for which $\psi(v)$ can fire. On G fire $\psi(v)$, then for every out-edge e of v, fire $\psi(e)$. On D fire v.

We show that for $\frac{3}{8}M \geq 2|V(D)|^2|E(D)|\Delta(D) + 1$ steps we can play this legally on both graphs. Note first that for an edge e of D, the change of the number of chips on $\psi(e)$ is at most one after each step. Hence at the beginning of a step a vertex of G of the form $\psi(e)$ can have at most $M/2 + \deg(y - base_D) + \frac{3}{8}M \leq M/2 + \frac{1}{8}M + \frac{3}{8}M < M + 1 = d(\psi(e))$ chips, so it cannot be fired. It also follows from this that on every such vertex the number of chips is positive, since it is at least $M/2 - \frac{3}{8}M > 0$. But y is a non-terminating distribution, hence at the beginning of a step we can find an active vertex, which therefore must be of the form $\psi(v)$ with $v \in V(D)$. After firing $\psi(v)$, $\psi(e)$ becomes active for every out-edge e of v, since $\psi(e)$ had a positive number of chips at the beginning of the step, and received M chips. Hence on G we can play in the desired way for $\frac{3}{8}M$ steps.

For the initial distributions, we have $y(\psi(v)) = d^+(v) \cdot M + x(v)$ for each $v \in V(D)$, so a vertex $v \in V(D)$ is active with respect to x if and only if $\psi(v)$ is active with respect to y. Let x' be the distribution on D and y' the distribution on G at the end of an arbitrary (but at most $\frac{3}{8}M^{\text{th}}$) step. Then it can be shown by induction that $y'(\psi(v)) = d^+(v) \cdot M + x'(v)$ for each $v \in V(D)$. So in each step we have that a vertex $v \in V(D)$ is active if and only if $\psi(v)$ is active.

Hence for $\frac{3}{8}M$ steps, the corresponding game on D is also legal. This means that there is a chip-firing game of length at least

$$\frac{3}{8}M \geq 2|V(D)|^2|E(D)|\Delta(D) + 1$$

on D with initial distribution x, and since x is non-negative on each vertex, [7, Corollary 4.9] implies that the distribution x is non-terminating. This finishes the proof. □

For a general digraph, the construction of the proof imitates the following game: If a vertex v fires, each of its out-neighbors u receives $\overrightarrow{d}(vu)$ chips, but the number of chips on v decreases by the in-degree of v. This modification of the chip-firing game has been studied by Asadi and Backman [2].

Theorem 3.3.5. *For a distribution $x \in \mathrm{Chip}(G)$ on an undirected graph G, computing $\mathrm{dist}(x)$ is **NP**-hard.*

Proof. The theorem follows from Theorem 3.2.1 and the previous lemma. □

Remark 3.3.6. For a simple Eulerian digraph D, one has

$$|E(\varphi(D))| \leq |E(D)| \cdot 9|V(D)|^3|E(D)| \leq 9|V(\varphi(D))|^5,$$

therefore the computation of dist is **NP**-hard even for graphs with $|E(G)| \leq 9|V(G)|^5$.

As a corollary of Theorem 3.3.5 and Corollary 1.4.2, we get the following.

Corollary 3.3.7. *For a divisor $f \in \mathrm{Div}(G)$ on a graph G, computing $\mathrm{rank}(f)$ is **NP**-hard.*

Now let us prove that the computation of dist and rank are also **NP**-hard on simple graphs. In [20], Hladký, Král' and Norine proved the following statement:

Proposition 3.3.8 ([20, Corollary 22.])**.** *Let f be a divisor on a graph G. Let G' be the simple graph obtained from G by subdividing each edge of G by an inner point and let f' be the divisor on G' that agrees with f on the vertices of G and has value 0 on new points. Then $\mathrm{rank}_G(f) = \mathrm{rank}_{G'}(f')$.*

By dualizing this statement, we get the following: For a distribution $x \in \mathrm{Chip}(G)$, if we get $x' \in \mathrm{Chip}(G')$ from x so that we put $d(v) - 1 - 0 = 1$ chip on each new vertex, and on the vertices of G, x' agrees with x, then $\mathrm{dist}_G(x) = \mathrm{rank}_G(\mathbf{d}_G - \mathbf{1}_{V(G)} - x) + 1 = \mathrm{rank}_{G'}(\mathbf{d}_{G'} - \mathbf{1}_{V(G')} - x') + 1 = \mathrm{dist}_{G'}(x')$.

When proving Theorem 3.3.5, we show a somewhat stronger statement: By Remark 3.3.6, computing dist is **NP**-hard even for graphs with $|E(G)| \leq 9|V(G)|^5$. For such a G, $|V(G')| = |V(G)| + |E(G)| \leq 10|V(G)|^5$. Hence G' and x' can be computed in polynomial time for such a graph G and $x \in \mathrm{Chip}(G)$, giving the following corollary.

Theorem 3.3.9. *For a distribution* $x \in \mathrm{Chip}(G)$ *on a simple graph* G*, computing* $\mathrm{dist}(x)$ *is* **NP**-*hard.*

Using Corollary 1.4.2 again, we have the following.

Theorem 3.3.10. *For a divisor* $f \in \mathrm{Div}(G)$ *on a simple graph* G*, computing* $\mathrm{rank}(f)$ *is* **NP**-*hard.*

Using a result of [28], we get that computing the rank of divisors is also **NP**-hard for so called tropical curves. Informally, a metric graph is a graph, where each edge has a positive length, and we consider our graph to be a metric space (the inner points of the edges are also points of this metric space). Tropical curve is more general in that we also allow some edges incident with vertices of degree one to have infinite length. A divisor on a tropical curve is an integer-valued function on the curve with only finitely many nonzero values. The notions of the degree of a divisor, linear equivalence, effective divisor and the rank can be defined as well, see [20].

A metric graph Γ corresponds to the graph G, if Γ is obtained from G by assigning some positive length to each edge.

Theorem 3.3.11 ([28, Theorem 1.6])**.** *Let* f *be a divisor on a graph* G*, and* Γ *be a metric graph corresponding to* G*. Then* $\mathrm{rank}_G(f) = \mathrm{rank}_\Gamma(f)$*.*

As a metric graph is a special tropical curve, we get the following corollary:

Corollary 3.3.12. *For a tropical curve* Γ*,* $f \in \mathrm{Div}(\Gamma)$*, computing* $\mathrm{rank}(f)$ *is* **NP**-*hard.*

From the positive side, we show the following:

Proposition 3.3.13. *For a simple undirected graph* G*, deciding whether for a given divisor* $f \in \mathrm{Div}(G)$*, and integer* k*,* $\mathrm{rank}(f) \leq k$ *is in* **NP***.*

Proof. For an input (f, k) with $\mathrm{rank}(f) \leq k$, our witness is the divisor $g \geq 0$ such that $\deg(g) \leq k+1$, and $\mathrm{rank}(f-g) = -1$ (such a g exists because $\mathrm{rank}(f) \leq k$).

First, we need to check that g can be given so that it has size polynomial in the size of (f, k). As $\deg(g) \leq k+1$, and $g \geq 0$, we have $g(v) \leq k+1$ for each vertex v. Therefore, the size of g is at most $O(|V(G)| \cdot \log k)$.

On the other hand, for simple graphs, it can be checked in polynomial time if $\mathrm{rank}(f-g) = -1$ [6], and also whether $\deg(g) \leq k+1$. □

By applying Corollary 1.4.2, for a simple graph, deciding whether for a given chip-distribution x, and integer k, $\mathrm{dist}(x) \leq k$ is also in **NP**.

3.4 Polynomial time computability in a special case

In this section we consider undirected graphs, and observe that for chip-distributions that are in a sense "small", computing the distance from non-terminating distributions can be done in polynomial time. Moreover, for these distributions, the distance from non-terminating distributions only depends on the number of edges of the graph and the number of chips in the distribution.

The corollaries of this observation for the case of divisors give a special case of the Riemann-Roch theorem.

Recall that Theorem 3.2.4 stated that $\mathrm{dist}(\mathbf{0}_G) = |E(G)|$ for any undirected graph G. We would like to generalize this statement for "small enough" distributions. We say that a distribution $x \in \mathrm{Chip}(G)$ is *under an acyclic orientation*, if there exists an acyclic orientation D of G such that $x(v) \le d_D^-(v)$ for each $v \in V(G)$.

Proposition 3.4.1. *Let G be a graph and let $x \in \mathrm{Chip}(G)$ be a distribution. If x is under an acyclic orientation then* $\mathrm{dist}(x) = |E(G)| - \deg(x)$.

Proof. From Theorem 3.2.4, a non-terminating distribution has at least $|E(G)|$ chips, therefore $\mathrm{dist}(x) \ge |E(G)| - \deg(x)$.

For the other direction, let D be an acyclic orientation of G with $x(v) \le d_D^-(v)$ for each $v \in V(G)$. Let y be the distribution on G corresponding to the indegrees of the orientation, i.e., $y(v) = d_D^-(v)$ for each $v \in V(G)$. Then, using Lemma 3.2.5, y is non-terminating, moreover $\deg(y) = |E(G)|$ and $y(v) \ge x(v)$ for each $v \in V(G)$. Hence $\mathrm{dist}(x) \le \deg(y - x) = |E(G)| - \deg(x)$. This completes the proof of the proposition. □

Remark 3.4.2. It can also be decided in polynomial time whether a distribution $x \in \mathrm{Chip}(G)$ is under an acyclic orientation. A greedy algorithm solves the problem.

From the previous proposition, using the duality between chip-distributions and divisors, we get a special case of the Riemann-Roch theorem for graphs.

Let us denote by K the canonical divisor on a graph G, that is, $K(v) = d(v) - 2$ for each vertex $v \in V(G)$.

Theorem 3.4.3 (Riemann-Roch for graphs, [5])**.** *Let G be a graph, and let f be a divisor on G. Then*

$$\mathrm{rank}(f) - \mathrm{rank}(K - f) = \deg(f) - |E(G)| + |V(G)|.$$

Now, from Corollary 1.4.2 and Proposition 3.4.1 we have for $f = \mathbf{d} - \mathbf{1}_V - x$ that

$$\mathrm{rank}(f) = \mathrm{dist}(x) - 1 = |E(G)| - \deg(x) - 1 = \deg(f) - |E(G)| + |V(G)| - 1,$$

if x is under an acyclic orientation.

We claim that in this case, $\operatorname{rank}(K-f) = -1$. Indeed, $K-f = K-(\mathbf{d}-\mathbf{1}_V-x) = x-\mathbf{1}$, so the dual of $K-f$ is $\mathbf{d}-\mathbf{1}_V-x+\mathbf{1}_V = \mathbf{d}-x$. The distribution x is under an acyclic orientation, let D be an orientation witnessing this, i.e., $x(v) \leq d^-_D(v)$ for each vertex $v \in V(G)$. Then $d(v) - x(v) \geq d^+_D(v)$ for each vertex $v \in V(G)$, hence we can use Lemma 3.2.5 for $\mathbf{d}-x$ and the directed graph obtained from D by reversing every edge. It follows that $\mathbf{d}-x$ is non-terminating, hence for its dual, $\operatorname{rank}(K-f) = -1$ by Corollary 1.4.2.

Therefore, we have $\operatorname{rank}(f) - \operatorname{rank}(K-f) = \deg(f) - |E(G)| + |V(G)|$, showing the Riemann-Roch theorem in this special case.

Chapter 4

Riemann–Roch theorems on digraphs

This chapter is based on [24], which is joint work with Bálint Hujter.

4.1 Introduction

Our aim in this chapter is to show, that the chip-firing framework can be used very effectively to prove Riemann–Roch type results both for graphs and digraphs. First, we give a short proof of the Riemann–Roch theorem on undirected graphs using the chip-firing framework. This proof is just a rephrasing of the proof of Cori and le Borgne [11] of the Riemann–Roch theorem on graphs. Then, we show that in the chip-firing language, this proof can be generalized to the case of Eulerian digraphs. This way we obtain a Riemann–Roch inequality for Eulerian digraphs. This inequality has been proved earlier in a weaker form by Amini and Manjunath [1, Section 6.2]. We obtain a stronger (and sharp) result, with a considerably simpler proof. We heavily build on connections between the chip-firing game and some well-known concepts of combinatorial optimization such as graph orientations and feedback arc sets.

In the second half of the chapter, we investigate the Riemann–Roch property for general strongly connected digraphs. Using the language of chip-firing, we give a necessary and sufficient condition for the Riemann–Roch theorem to hold on a directed graph. At first sight, this setting seems to be somewhat restricted, but it follows from a result of Perkinson, Perlman and Wilmes [32, Theorem 4.11], that divisor theory on strongly connected digraphs is equivalent to divisor theory on lattices (in the sense of Amini and Manjunath [1]), and to the setting of the abstract Riemann–Roch theorem of Baker and Norine [5, Section 2]. Hence our graphical Riemann–Roch condition has the same power as the abstract Riemann–

Roch condition of Baker and Norine, and the condition of Amini and Manjunath. We point out that these three theorems are almost equivalent.

We also investigate the natural Riemann–Roch property defined by Asadi and Backman [2, Definition 3.12], proving that an Eulerian digraph has the natural Riemann–Roch property if and only if it is bidirected i.e. corresponds to an undirected graph. On the other hand, in Section 4.6, we show that there exist non-Eulerian digraphs with the natural Riemann–Roch property. We also give examples of Eulerian digraphs with non-natural Riemann–Roch property, and of digraphs with no Riemann–Roch property.

4.2 Non-terminating chip-distributions and turnback arc sets

Recall from Chapter 3 the following characterization of non-terminating chip-distributions on Eulerian digraphs:

Theorem 3.2.8. *Let G be an Eulerian digraph. A chip-distribution $x \in \mathrm{Chip}(G)$ is non-terminating if and only if there exists a turnback arc set T of G, and a chip-distribution $a \in \mathrm{Chip}(G)$ with $a \geq 0$, such that if ϱ is the indegree distribution of T, then $x \sim \varrho + a$.*

This connection between turnback arc sets and non-terminating chip-distributions will be crucial in our proofs of Riemann–Roch-type theorems.

Let us also state the above theorem for the special case of undirected, i.e. bidirected graphs. In a bidirected graph, a turnback arc set must include exactly one version of each bidirected edge, i.e for each turnback arc set T, for each $uv \in E$, either $\overrightarrow{uv} \in T$ or $\overrightarrow{vu} \in T$. Hence a turnback arc set corresponds to an orientation of G. This orientation also needs to be acyclic, therefore turnback arc sets in undirected graphs correspond to acyclic orientations. Hence one can deduce the following corollary (whose statement is equivalent to [8, Theorem 2.3])

Proposition 4.2.1. *Let G be an undirected graph. A chip-distribution $x \in \mathrm{Chip}(G)$ is non-terminating if and only if there exists an acyclic orientation $\overrightarrow{G}$ of G, and a chip-distribution $a \in \mathrm{Chip}(G)$ with $a \geq \mathbf{0}_G$, such that if ϱ is the indegree distribution of $\overrightarrow{G}$, then $x \sim \varrho + a$.*

4.3 The Riemann–Roch theorem for undirected graphs

Recall the Riemann–Roch theorem for graphs:

Theorem 1.3.2 (Riemann–Roch theorem for graphs [5, Theorem 1.12])**.** *Let G be an undirected graph and let f be a divisor on G. Then*

$$\mathrm{rank}(f) - \mathrm{rank}(K_G - f) = \deg(f) - \mathfrak{g} + 1$$

where $\mathfrak{g} = |E(G)| - |V(G)| + 1$ *and* $K_G(v) = d(v) - 2$ *for each* $v \in V(G)$.

Using the connection between dist and rank (Corollary 1.4.2), we can state the following equivalent form of this theorem.

Theorem 4.3.1 (Riemann–Roch)**.** *Let G be an undirected graph. Then for any* $x \in \mathrm{Chip}(G)$*:*

$$\mathrm{dist}(x) - \mathrm{dist}(\mathbf{d}_G - x) = |E(G)| - \deg(x).$$

The equivalence of the two forms can be seen by choosing $x = \mathbf{d}_G - \mathbf{1}_G - f$.

Several proofs of the Riemann–Roch theorem for graphs have been published (see for example [5, 11, 1, 4, 43]). Here we give the proof of Cori and le Borgne rephrased in the chip-firing language. The key idea of each proof is to understand the relationship of the so-called non-special divisors to each other and to other divisors. A divisor is called non-special, if its degree is equal to $\mathfrak{g} - 1$ and it is is not equi-effective. (Here, $\mathfrak{g}$ again means the genus of the graph.) In the chip-firing setting, these non-special divisors correspond to minimal non-terminating chip-distributions, which are characterized by Proposition 4.2.1. The fact that Proposition 4.2.1 has a version for Eulerian digraphs (Theorem 3.2.8) enables us to generalize the proof of Cori and le Borgne to Eulerian digraphs.

Before giving the proof, let us remark the following:

Claim 4.3.2. *Let G be a strongly connected digraph and $x, y \in \mathrm{Chip}(G)$. If $x \sim y$, then* $\mathrm{dist}(x) = \mathrm{dist}(y)$.

Proof. This is a straightforward consequence of Lemma 1.2.6. □

Proof of Theorem 4.3.1. [11] Let $|E(G)| = m$. First, we prove that $\mathrm{dist}(\mathbf{d} - x) \leq \mathrm{dist}(x) - m + \deg(x)$.

From the definition of dist, there exists a chip-distribution $a \geq \mathbf{0}$, with $\deg(a) = \mathrm{dist}(x)$ such that $x + a$ is non-terminating.

Then, from Proposition 4.2.1, there exists an acyclic orientation $\overrightarrow{G}$ of G with indegree distribution ϱ, and a chip-distribution $b \geq 0$, such that $\varrho + b \sim x + a$

Let $x' = \varrho + b - a$. Clearly, $x' \sim x$, and $\mathbf{d} - x' \sim \mathbf{d} - x$. Therefore, since the dist values of two linearly equivalent chip-distributions are equal (Claim 4.3.2), it is enough to show that $\mathrm{dist}(\mathbf{d} - x') \leq \mathrm{dist}(x) - m + \deg(x') = \mathrm{dist}(x) - m + \deg(x)$.

As $x' + a = \varrho + b$,

$$(\mathbf{d} - x') + b = (\mathbf{d} - \varrho) + a.$$

Note that $\mathbf{d}-\varrho$ is the indegree vector of the reverse of $\overrightarrow{G}$, which is also an acyclic orientation. Therefore, again by Proposition 4.2.1, $(\mathbf{d}-x')+b$ is non-terminating, showing that

$$\mathrm{dist}(\mathbf{d}-x') \le \deg(b) = \deg(x') + \deg(a) - \deg(\varrho) = \deg(x) + \mathrm{dist}(x) - m.$$

From this, we have $\mathrm{dist}(x) - \mathrm{dist}(\mathbf{d}-x) \ge m - \deg(x)$.

Now let $y = \mathbf{d} - x$. Then $x = \mathbf{d} - y$. From the above argument, we have

$$\mathrm{dist}(x) = \mathrm{dist}(\mathbf{d}-y) \le \mathrm{dist}(y) - m - \deg(y) = \mathrm{dist}(\mathbf{d}-x) - m - (2m - \deg(x)),$$

giving $\mathrm{dist}(x) - \mathrm{dist}(\mathbf{d}-x) \le m - \deg(x)$. □

4.4 A weak Riemann–Roch theorem for Eulerian digraphs

The proof of the Riemann–Roch theorem for undirected graphs from Section 4.3 can be naturally generalized to Eulerian digraphs to give the following weak Riemann–Roch theorem:

Theorem 4.4.1. *Let G be an Eulerian digraph. For each $x \in \mathrm{Chip}(G)$,*

$$\mathrm{minfas}(G) - \deg(x) \le \mathrm{dist}(x) - \mathrm{dist}(\mathbf{d}_G^- - x) \le |E(G)| - \mathrm{minfas}(G) - \deg(x).$$

Amini and Manjunath [1, Section 6.2.] proved this theorem for Eulerian digraphs where each edge has multiplicity at least one by using a limiting argument. For the general case they only obtained a lower bound of $\mathrm{minfas}(G) - \deg(x) - 2$ and an upper bound of $|E(G)| - \mathrm{minfas}(G) - \deg(x) + 2$. Here we give a proof of the tighter bound in the general case.

Proof. First, we prove that $\mathrm{dist}(\mathbf{d}^- - x) \le \mathrm{dist}(x) - \mathrm{minfas}(G) + \deg(x)$.

From the definition of $\mathrm{dist}(x)$, there exists a chip-distribution $a \ge 0$ with $\deg(a) = \mathrm{dist}(x)$ such that $x + a$ is non-terminating.

Then, from Theorem 3.2.8, there exists a turnback arc set T of G with indegree vector ϱ, and a chip-distribution $b \ge 0$ such that $x + a \sim \varrho + b$. Let $x' = \varrho + b - a$. Clearly, $x' \sim x$, and $\mathbf{d}^- - x' \sim \mathbf{d}^- - x$. Therefore, by Claim 4.3.2, it is enough to show that $\mathrm{dist}(\mathbf{d}^- - x') \le \mathrm{dist}(x) - \mathrm{minfas}(G) + \deg(x') = \mathrm{dist}(x) - \mathrm{minfas}(G) + \deg(x)$.

As $x' + a = \varrho + b$,

$$(\mathbf{d}^- - x') + b = (\mathbf{d}^- - \varrho) + a.$$

Note that $\mathbf{d}^- - \varrho$ is the indegree vector of the complement edge-set of T, which is also a turnback arc set. Therefore, again by Theorem 3.2.8, $(\mathbf{d}^- - x') + b$ is non-terminating showing that

$$\mathrm{dist}(\mathbf{d}^- - x') \le \deg(b) = \deg(x') + \deg(a) - \deg(\varrho) \le \deg(x) + \mathrm{dist}(x) - \mathrm{minfas}(G),$$

since $\deg(a) = \text{dist}(x)$, and $\deg(\varrho)$ equals the cardinality of the turnback arc set T which is at least $\text{minfas}(G)$ as a turnback arc set is also a feedback arc set.

From this, we have $\text{dist}(x) - \text{dist}(\mathbf{d}^- - x) \geq \text{minfas}(G) - \deg(x)$.

Now let $y = \mathbf{d}^- - x$. Then $x = \mathbf{d}^- - y$. From the above argument, we have $\text{dist}(x) = \text{dist}(\mathbf{d}^- - y) \leq \text{dist}(y) - \text{minfas}(G) + \deg(y) = \text{dist}(\mathbf{d}^- - x) - \text{minfas}(G) + (|E(G)| - \deg(x)) = \text{dist}(\mathbf{d}^- - x) + |E(G)| - \text{minfas}(G) - \deg(x)$, giving $\text{dist}(x) - \text{dist}(\mathbf{d}^- - x) \leq |E(G)| - \text{minfas}(G) - \deg(x)$. □

Remark 4.4.2. The theorem is sharp in the following sense: for any digraph G, taking x to be the indegree-distribution of a minimum cardinality turnback arc set, $\text{dist}(x) = \text{dist}(\mathbf{d}^- - x) = 0$, hence $\text{dist}(x) - \text{dist}(\mathbf{d}^- - x) = \text{minfas}(G) - \deg(x)$. On the other hand, for $y = \mathbf{d}^- - x$, $\text{dist}(y) - \text{dist}(\mathbf{d}^- - y) = |E(G)| - \text{minfas}(G) - \deg(y)$.

For a bidirected graph G, $\text{minfas}(G) = \frac{1}{2}|E(G)|$, hence in this case, we get back the Riemann–Roch theorem of Baker and Norine. The formulas are seemingly different because we define the number of edges in a different way for an undirected graph and for a bidirected graph. In other words, if G is a bidirected graph, and G' is an undirected graph corresponding to it, then $|E(G)| = 2|E(G')|$.

4.5 The Riemann–Roch property for digraphs

A strongly connected digraph G is said to have the Riemann–Roch property if there exists some $K \in \text{Chip}(G)$ and integer t, such that for each $x \in Chip(G)$,

$$\text{dist}(x) - \text{dist}(K - x) = t - \deg(x).$$

In this case we say that K is a canonical distribution for G. Examples of Section 4.6 show that such K and t does not always exist for a digraph.

From the divisor-theoretic point of view, the existence of such K and t implies, that for the divisor $\tilde{K} = 2 \cdot \mathbf{d}_G^+ - 2 \cdot \mathbf{1}_G - K$, each divisor $f \in \text{Div}(G)$ has

$$\text{rank}(f) - \text{rank}(\tilde{K} - f) = t - |E(G)| + |V(G)| + \deg(f).$$

In this section, we investigate the properties of such K and t and give a necessary, and a necessary and sufficient condition for a strongly connected digraph to have the Riemann–Roch property.

Proposition 4.5.1. *If for a strongly connected digraph G, the Riemann–Roch formula*

$$\text{dist}(x) - \text{dist}(K - x) = t - \deg(x)$$

holds for all $x \in Chip(G)$ for some $K \in$ Chip(G) *and value t, then* $\deg(K) = 2t$.

Proof. Take an arbitrary $x \in \mathrm{Chip}(G)$, and write up the Riemann–Roch formula for x and for $K - x$. Note that $K - (K - x) = x$.

$$\begin{aligned} \mathrm{dist}(x) - \mathrm{dist}(K - x) &= t - \deg(x) \\ \mathrm{dist}(K - x) - \mathrm{dist}(x) &= t - \deg(K - x) \end{aligned}$$

Summing these two equalities, we get $2t = \deg(K - x) + \deg(x) = \deg(K)$. □

Proposition 4.5.2. *If for a strongly connected digraph G, the Riemann–Roch formula*

$$\mathrm{dist}(x) - \mathrm{dist}(K - x) = t - \deg(x)$$

holds for all $x \in Chip(G)$ for some $K \in \mathrm{Chip}(G)$ and value t, then $t = \mathrm{dist}(\mathbf{0}_G)$.

Proof. First we show that $t \leq \mathrm{dist}(\mathbf{0}_G)$. Indeed, the Riemann–Roch formula for $x = \mathbf{0}_G$ says that $\mathrm{dist}(\mathbf{0}_G) - \mathrm{dist}(K) = t - 0$. As $\mathrm{dist}(K) \geq 0$, we have $t \leq \mathrm{dist}(\mathbf{0}_G)$.

Now suppose that $t = \mathrm{dist}(\mathbf{0}_G) - k$. We know that k is non-negative, we need to show that k is non-positive. Take a chip-distribution $x \in \mathrm{Chip}(G)$ such that x is non-terminating, and $\deg(x) = \mathrm{dist}(\mathbf{0}_G)$. Such an x exists by definition of $\mathrm{dist}(\mathbf{0}_G)$.

The Riemann–Roch formula says for this x, that $\mathrm{dist}(x) - \mathrm{dist}(K - x) = t - \deg(x) = \mathrm{dist}(\mathbf{0}_G) - k - \mathrm{dist}(\mathbf{0}_G) = -k$. Since x is non-terminating, $\mathrm{dist}(x) = 0$. Hence we have $\mathrm{dist}(K - x) = k$. But from Proposition 4.5.1, we have $\deg(K - x) = \deg(K) - \deg(x) = 2\mathrm{dist}(\mathbf{0}_G) - 2k - \mathrm{dist}(\mathbf{0}_G) = \mathrm{dist}(\mathbf{0}_G) - 2k$. As a non-terminating distribution has degree at least $\mathrm{dist}(\mathbf{0}_G)$, we necessarily have $\mathrm{dist}(K - x) \geq 2k$, which means $k \geq 2k$. This implies $k = 0$. □

Now we give a necessary, and a necessary and sufficient condition for a strongly connected digraph to have the Riemann–Roch property.

Definition 4.5.3. Let us call a chip-distribution $x \in \mathrm{Chip}(G)$ *minimally non-terminating*, if it is non-terminating, but for each $v \in V(G)$, $x - \mathbf{1}_v$ is terminating.

Proposition 4.5.4. *If the Riemann–Roch formula holds for a strongly connected digraph G, then all minimally non-terminating distributions have degree* $\mathrm{dist}(\mathbf{0}_G)$.

Proof. Suppose that the Riemann–Roch formula holds for G, $x \in \mathrm{Chip}(G)$ is non-terminating, and $\deg(x) = \mathrm{dist}(\mathbf{0}_G) + k$ with $k > 0$. We show that x is not minimally non-terminating.

From the Riemann–Roch formula, $\mathrm{dist}(x) - \mathrm{dist}(K - x) = \mathrm{dist}(\mathbf{0}_G) - \deg(x)$, thus $\mathrm{dist}(K - x) = k$. Since $\deg(K - x) = \mathrm{dist}(\mathbf{0}_G) - k$, this means that there exists a chip-distribution $a \in \mathrm{Chip}(G)$, $a \geq 0$, $\deg(a) = k$, such that $K - x + a$ is non-terminating, and is of degree $\mathrm{dist}(\mathbf{0}_G)$.

By the Riemann–Roch formula, $\mathrm{dist}(K - x + a) - \mathrm{dist}(x - a) = \mathrm{dist}(\mathbf{0}_G) - \deg(K - x + a)$. As $\mathrm{dist}(K - x + a) = 0$ and $\deg(K - x + a) = \mathrm{dist}(\mathbf{0}_G)$, we have $\mathrm{dist}(x - a) = 0$, hence $x - a$ is non-terminating. Since $a \geq 0$, we conclude that x is not minimally non-terminating. □

Theorem 4.5.5. *Let G be a strongly connected digraph. G has the Riemann–Roch property if and only if each minimally non-terminating distribution has degree* $\mathrm{dist}(\mathbf{0}_G)$*, and there exists a distribution $K \in \mathrm{Chip}(G)$ such that for any minimally non-terminating distribution $x \in \mathrm{Chip}(G)$, $K-x$ is also minimally non-terminating. If the above condition holds, then the Riemann–Roch formula holds for G with K as canonical distribution.*

Note that the above condition implies also that $\deg(K) = 2 \cdot \mathrm{dist}(\mathbf{0}_G)$.

Proof. First we show the "only if" direction. Suppose that G has the Riemann–Roch property with canonical distribution K. Then by Proposition 4.5.4, each minimally non-terminating distribution has degree $\mathrm{dist}(\mathbf{0}_G)$. Suppose that x is minimally non-terminating. Then $\mathrm{dist}(x) = 0$ and $\deg(x) = \mathrm{dist}(\mathbf{0}_G)$. Since $\mathrm{dist}(x) - \mathrm{dist}(K-x) = \mathrm{dist}(\mathbf{0}_G) - \deg(x) = 0$, we have $\mathrm{dist}(K-x) = 0$, thus $K-x$ is non-terminating. Also, since $0 = \mathrm{dist}(K-x) - \mathrm{dist}(x) = \mathrm{dist}(\mathbf{0}_G) - \deg(K-x)$, we have $\deg(K-x) = \mathrm{dist}(\mathbf{0}_G)$, thus $K-x$ is minimally non-terminating.

Now, we show the "if" direction. Take a distribution $x \in \mathrm{Chip}(G)$. It is enough to show that $\mathrm{dist}(x) - \mathrm{dist}(K-x) \geq \mathrm{dist}(\mathbf{0}_G) - \deg(x)$, as then for $K-x$, $\mathrm{dist}(K-x) - \mathrm{dist}(x) \geq \mathrm{dist}(\mathbf{0}_G) - \deg(K-x) \geq \mathrm{dist}(\mathbf{0}_G) - (2\mathrm{dist}(\mathbf{0}_G) - \deg(x)) = \deg(x) - \mathrm{dist}(\mathbf{0}_G)$ which implies the equality.

x is either terminating or non-terminating.

First suppose that x is terminating. Then $\mathrm{dist}(x) = k > 0$. This means that there exists $a \in \mathrm{Chip}(G)$, $a \geq 0$, $\deg(a) = k$ such that $x + a$ is non-terminating.

There exists at least one minimally non-terminating distribution y such that $x + a = y + b$ where $b \geq 0$. (We can take off chips until our distribution gets minimally non-terminating.) Then by our assumption $\deg(y) = \mathrm{dist}(\mathbf{0}_G)$, and $K-y$ is also a minimally non-terminating distribution. We have $K-(x+a) = K-(y+b)$, thus $(K-x)+b = (K-y)+a$. $(K-y)+a$ is non-terminating, as $K-y$ is non-terminating and $a \geq 0$. Thus $\mathrm{dist}(K-x) \leq \deg(b) = \deg(x) + \deg(a) - \deg(y) = \deg(x) + \mathrm{dist}(x) - \mathrm{dist}(\mathbf{0}_G)$.

Hence

$$\mathrm{dist}(x) - \mathrm{dist}(K-x) \geq \mathrm{dist}(x) - (\deg(x) + \mathrm{dist}(x) - \mathrm{dist}(\mathbf{0}_G)) = \mathrm{dist}(\mathbf{0}_G) - \deg(x).$$

Now suppose that x is non-terminating. Then there exists a minimally non-terminating distribution $y \in \mathrm{Chip}(G)$ such that $x = y + b$ where $b \geq 0$. By our assumptions, $\deg(y) = \mathrm{dist}(\mathbf{0}_G)$ and $K-y$ is also minimally non-terminating.

$K - x + b = K - y$. As $K-y$ is non-terminating, $\mathrm{dist}(K-x) \leq \deg(b)$. On the other hand, $\deg(x) = \deg(y) + \deg(b) = \mathrm{dist}(\mathbf{0}_G) + \deg(b)$. Thus,

$$\mathrm{dist}(x) - \mathrm{dist}(K-x) \geq 0 - \deg(b) = 0 - (\deg(x) - \mathrm{dist}(\mathbf{0}_G)) = \mathrm{dist}(\mathbf{0}_G) - \deg(x).$$

□

We note that Theorem 4.5.5 is nearly equivalent to two previous Riemann–Roch conditions, the abstract Riemann–Roch condition of Baker and Norine [5, Theorem 2.2], and the condition of Amini and Manjunath [1, Theorem 1.4], which uses the language of lattices.

Let us first point out that the three theorems describe the same situation. In the theorem of Baker and Norine, $\mathrm{Div}(X)$ is a free Abelian group on a finite set X, which is equipped with an equivalence relation satisfying two given properties, (E1) and (E2). These two properties hold if and only if the differences of equivalent divisors form a lattice $\Gamma \subset \mathbb{Z}^n_0$ (here $\mathbb{Z}^n_0$ denotes the set $\{(x_1, x_2, \dots, x_n) \in \mathbb{Z}^n : \sum_{i=1}^n x_i = 0\}$). Amini and Manjunath consider this later situation, i.e., for them, divisors are elements of $\mathbb{Z}^n$ and for a fixed lattice $\Gamma \subset \mathbb{Z}^n_0$ they call two divisors equivalent if their difference is in Γ. The case of divisor theory on strongly connected digraphs corresponds to the case if the lattice Γ is generated by the Laplacian matrix of a strongly connected digraph. Since by a theorem of Perkinson, Perlman and Wilmes [32, Theorem 4.11], each lattice $\Gamma \subset \mathbb{Z}^n_0$ can be generated by the Laplacian matrix of a strongly connected digraph, the graphical case is indeed equivalent to the other two.

Hence in the setting of Baker and Norine we can also suppose that the set X is the vertex set of a strongly connected digraph, and the linear equivalence is defined with the Laplacian matrix of this digraph. Using the duality between chip-firing and graph divisor theory (Proposition 1.4.1), one can show that the $(RR1)$ condition of Baker and Norine is equivalent to the condition "each minimally non-terminating distribution has degree $\mathrm{dist}(\mathbf{0}_G)$". $(RR2)$ is equivalent to the condition "For each non-terminating chip-distribution x of degree $\mathrm{dist}(\mathbf{0}_G)$, $K - x$ is also non-terminating." This condition is equivalent to our second condition if $(RR1)$ holds.

In the criterion of Amini and Manjunath, the set *Ext* corresponds to the set of minimally non-terminating chip-distributions in our setting. If they defined the set *Ext* with ℓ_1-norm, their conditions would be equivalent to ours. However, they define *Ext* using ℓ_∞-norm, which makes their conditions different from ours.

4.5.1 The natural Riemann–Roch property in Eulerian digraphs

Asadi and Backman introduced the following variant of the Riemann–Roch property [2, Definition 3.12]: A digraph G has the *natural Riemann–Roch property*, if it satisfies a Riemann–Roch formula with canonical divisor $K(v) = d^+(v) - 2$ for each $v \in V$. This definition translates to the language of chip-firing in the following way:

Definition 4.5.6. A digraph G has the natural Riemann–Roch property, if for each

$x \in \mathrm{Chip}(G)$

$$\mathrm{dist}(x) - \mathrm{dist}(\mathbf{d}_G^+ - x) = \frac{1}{2}|E(G)| - \deg(x)$$

From the Riemann–Roch theorem for undirected graphs, it follows that each undirected (that is, bidirected) graph has the natural Riemann–Roch property. However it is left open in [2] whether there are any other such graphs.

In Section 4.6.3 we show an example that a non-bidirected graph can also have the natural Riemann–Roch property. However, the following theorem shows that this is not possible if the digraph is Eulerian.

Theorem 4.5.7. *Let G be an Eulerian digraph. Then G has the natural Riemann–Roch property if and only if it is a bidirected graph corresponding to an undirected graph (i.e. $\overrightarrow{d}(u,v) = \overrightarrow{d}(v,u)$ for any pair of vertices u, v).*

Proof. Suppose that G has the natural Riemann–Roch property. Then we have $K(v) = d^+(v)$ $\forall v \in V(G)$, thus $\deg(K) = |E(G)|$. Propositions 4.5.1 and 4.5.2 imply that $\mathrm{dist}(\mathbf{0}_G) = \frac{1}{2}\deg(K) = \frac{1}{2}|E(G)|$. Theorem 3.2.8 says that for Eulerian digraphs, $\mathrm{dist}(\mathbf{0}_G) = \mathrm{minfas}(G)$. As a consequence, we have $\mathrm{minfas}(G) = \frac{1}{2}|E(G)|$.

For an ordering $v_1, v_2, \ldots, v_n$ of $V(G)$, we call an arc $\overrightarrow{v_iv_j}$ with $i < j$ a *forward arc*, and an arc $\overrightarrow{v_iv_j}$ with $j < i$ a *backward arc.*

For any ordering, the set of forward arcs forms a feedback arc set. The same is true for the set of backward arcs. As $E(G)$ is the disjoint union of these two sets and $\mathrm{minfas}(G) = \frac{1}{2}|E(G)|$, it follows that for any ordering of vertices there are exactly $\frac{1}{2}|E(G)|$ forward arcs and exactly $\frac{1}{2}|E(G)|$ backward arcs.

Suppose that there is a pair of vertices (u, v) with $\overrightarrow{d}(u,v) \neq \overrightarrow{d}(v,u)$. Then consider orderings $u, v, v_3, \ldots, v_n$ and $v, u, v_3, \ldots, v_n$. The set of forward arcs has different cardinality for these two orderings, therefore they cannot both have size $\frac{1}{2}|E(G)|$, which is a contradiction. □

4.6 Examples

In this section we provide examples showing that a digraph may not have the Riemann–Roch property, but for certain digraphs such a theorem can still hold.

4.6.1 A digraph without Riemann–Roch property

Consider the following graph G_1 (see also Figure 4.1):

$$\begin{aligned} V(G_1) &= \{v_1, v_2, v_3, v_4, v_5, v_6\}, \\ E(G_1) &= \{\overrightarrow{v_1v_2}; \overrightarrow{v_2v_3}, \overrightarrow{v_3v_4}, \overrightarrow{v_4v_5}, \overrightarrow{v_5v_6}, \overrightarrow{v_6v_1}, \overrightarrow{v_1v_5}, \overrightarrow{v_2v_6}, \overrightarrow{v_3v_1}, \overrightarrow{v_4v_2}, \overrightarrow{v_5v_3}, \overrightarrow{v_6v_4}\}. \end{aligned}$$

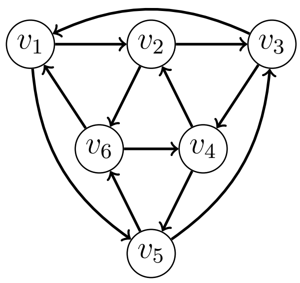

Figure 4.1: G_1, a graph with no Riemann–Roch property

It is easy to check that for G_1, $x_1 = (1, 0, 0, 1, 0, 2)$ and $x_2 = (2, 1, 1, 1, 0, 0)$ are both minimally non-terminating. Since $deg(x_1) \neq \deg(x_2)$, Proposition 4.5.4 tells us that the Riemann–Roch formula does not hold for G_1.

4.6.2 Eulerian digraphs with non-natural Riemann–Roch property

A very simple example of an Eulerian digraph with Riemann–Roch property is a directed cycle. It is straightforward that for a directed cycle, any chip-distribution of degree at least one is non-terminating. On the other hand, since minfas=1, any chip-distribution of degree less than one is terminating. Thus, the minimally non-terminating distributions are exactly the distributions of degree 1. Let K be any distribution of degree two. It is straightforward that the conditions of Theorem 4.5.5 hold, thus a directed cycle has the Riemann–Roch property.

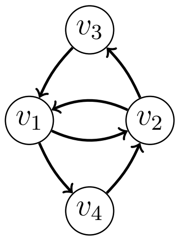

Figure 4.2: G_2, a graph with non-natural Riemann–Roch property

Another example, where there are more than one equivalence classes with the same number of chips is the following graph G_2 (see also Figure 4.2):

$$V(G_2) = \{v_1, v_2, v_3, v_4\}; \quad E(G_2) = \{\overrightarrow{v_1v_2}, \overrightarrow{v_1v_4}, \overrightarrow{v_2v_1}, \overrightarrow{v_2v_3}, \overrightarrow{v_3v_1}, \overrightarrow{v_4v_2}\}.$$

For this graph, $\mathrm{dist}(\mathbf{0}_{G_2}) = \mathrm{minfas}(G_2) = 2$. It is well-known that for an Eulerian digraph, the number of equivalence classes of chip-distributions of a fixed degree equals the number of spanning in-arborescences rooted at an arbitrary vertex v. Thus, this graph has 2 equivalence classes of degree two. It easy to check that the distribution $(2, 0, 0, 0)$ is non-terminating, while the distribution $(1, 1, 0, 0)$ is terminating. So for $K = (4, 0, 0, 0)$ the conditions of Theorem 4.5.5 hold.

4.6.3 A non-Eulerian digraph with natural Riemann–Roch property

We have seen in Section 4.5.1 that an Eulerian digraph has the natural Riemann–Roch property if and only if it is bidirected. Here we show that there exist also non-Eulerian digraphs with the natural Riemann–Roch property.

Let G_3 be the following graph (see also Figure 4.3).

$$\begin{aligned} V(G_3) &= \{v_1, v_2, v_3, v_4\}; \\ E(G_3) &= \{\overrightarrow{v_1v_2}, \overrightarrow{v_1v_4}, \overrightarrow{v_2v_1}, \overrightarrow{v_3v_2}, \overrightarrow{v_3v_2}, \overrightarrow{v_4v_3}, \overrightarrow{v_4v_3}, \overrightarrow{v_4v_1}\}. \end{aligned}$$

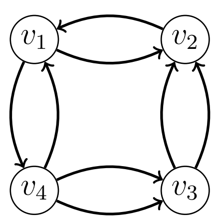

Figure 4.3: G_3, a non-Eulerian graph with the natural Riemann–Roch property

We claim that in this graph, there is only one minimally non-terminating equivalence class, which is the equivalence class of $(1, 0, 0, 3)$.

First, we show that a minimally non-terminating distribution needs to have degree 4. One can immediately see that any non-terminating distribution has degree at least 3, since v_4 has outdegree 3, and in a non-terminating game on a strongly connected digraph, each vertex is fired infinitely often (by Proposition 3.2.2). Hence each non-terminating equivalence class contains an element with at least 0 chips on each vertex, and at least 3 chips on v_4. Hence a non-terminating degree-3 equivalence class could only be the class of $(0, 0, 0, 3)$, but it is easy to check that this distribution is terminating.

In a non-terminating equivalence class of degree 4, there is also an element with at least 3 chips on v_4 and at least 0 chips on the other vertices. From the four choices $(1, 0, 0, 3), (0, 1, 0, 3), (0, 0, 1, 3)$ and $(0, 0, 0, 4)$, we can check that $(1,0,0,3),(0,1,0,3)$ and $(0,0,0,4)$ are equivalent and non-terminating, and $(0,0,1,3)$ is terminating.

Since we only have one minimally non-terminating equivalence class, the conditions of Theorem 4.5.5 trivially hold with $K=(2,0,0,6)\sim(2,1,2,3)=\mathbf{d}^+_{G_3}$. Thus, G_3 has the natural Riemann–Roch property.

Chapter 5

Rotor-routing and the notion of linear equivalence

This chapter is based on [40].

5.1 Introduction

Rotor-routing is a one player game on a ribbon digraph, that can be considered a refinement of chip-firing. It was introduced in the physics literature as a model of self-organized criticality [35, 36, 14]. The rotor walk can also be thought of as a derandomized random walk on a graph [22].

In this chapter, we explore the relationship of rotor-routing with the chip-firing game, and the Picard group of the graph. We analyze the version of rotor-routing, where each vertex has an integer number of chips, which might also be negative. This model has sometimes been called the height-arrow model [12].

In Section 5.2, we characterize recurrent elements for the rotor-routing game on strongly connected digraphs. This result is motivated by the fact that for the chip-firing game, no characterization is known for the recurrent elements on strongly connected digraphs.

In Section 5.3, we define the analogue of the notion of linear equivalence of the chip-firing game for the rotor-routing game. We show that the linear equivalence notions of the two models are related in a simple way. Moreover, whether two configurations of the rotor-routing game are linearly equivalent can be decided in polynomial time.

We use this result to prove polynomial time decidability of the reachability problem for rotor-routing in a special case. This result is an analogue of Theorem 2.4.3. In particular, we show, that it can be decided in polynomial time whether two unicycles lie in the same rotor-router orbit. Using the relationship between linear equivalence for chip-firing and for rotor-routing, we give a simple proof for the fact that the number of rotor-router unicycle orbits equals the order of the Picard group of the graph. Finally, we show that for Eulerian digraphs, the rotor-router action of the Picard group on the set of spanning in-arborescences [21] can also be interpreted in terms of the linear equivalence. Using this interpretation, we give a simpler proof for the result of Chan et al. [10] stating that the rotor-router action of the Picard group of a graph is independent of the base point if and only if all cycles in the graph are reversible.

5.1.1 Definitions

The rotor-routing game is played on a ribbon digraph. A *ribbon digraph* is a digraph together with a fixed cyclic ordering of the outgoing edges from v for each vertex v. For an edge $e = \overrightarrow{vw}$, we denote by e^+ the edge following e in the cyclic order at v. In this chapter, we always assume that our digraphs are strongly connected, and have a ribbon digraph structure. For digraphs with multiple edges, we use the following model for encoding a ribbon structure: For each vertex, we list the out-edges according to the cyclic order broken up at an arbitrary point. If in there are consecutive parallel edges in this order, we only write down one instance of the edge, and the number of the consecutive parallel edges.

Let G be a strongly connected ribbon digraph. A *rotor configuration* on G is a function ϱ that assigns to each vertex v an edge with tail v. We call $\varrho(v)$ the *rotor* at v. For a rotor configuration ϱ, we call the subgraph with edge set $\{\varrho(v) : v \in V(G)\}$ the *rotor subgraph*.

A configuration of the rotor-routing game is a pair (x, ϱ), where $x \in \mathbb{Z}^{V(G)}$ is an integer vector, and ϱ is a rotor configuration on G. We call such pairs *chip-and-rotor configuration*, or just shortly CRC. We will mostly think of x as a chip-distribution, but sometimes we think of it as a divisor. For a vertex v, we refer to $x(v)$ as the number of chips on v.

Given a configuration (x, ϱ), a *routing* at vertex v results in the configuration (x', ϱ'), where ϱ' is the rotor configuration with

$$\varrho'(u) = \begin{cases} \varrho(u) & \text{if } u \neq v, \\ \varrho(u)^+ & \text{if } u = v, \end{cases}$$

and $x' = x - \mathbf{1}_v + \mathbf{1}_{v'}$ where v' is the head of $\varrho'(v)$.

We call the routing at v *legal* (with respect to the configuration (x, ϱ)), if $x(v) > 0$, i.e. the routing at v does not create a negative entry at v. A *legal game* is a sequence of configurations such that each configuration is obtained from the previous one by a legal routing.

An important special case of the rotor-routing game is when the initial configuration has a nonnegative chip-distribution of degree one, i.e. one vertex has one

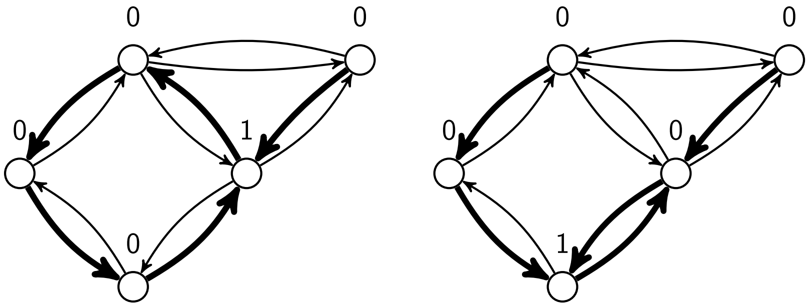

Figure 5.1: A legal rotor-routing step. The rotor edges are drawn by thick lines.

chip, and the other vertices have zero chips. Such a configuration is called a *single-chip-and-rotor configuration*. For such a configuration, there is exactly one vertex at which one can perform a legal routing, namely, the vertex of the chip, and the legal routing again leads to a single-chip-and-rotor configuration. Thus, in this case, the rotor-routing game is deterministic. We call this special case the *classical rotor-routing process*. The *orbit* of a single-chip-and-rotor configuration is defined as the set of configurations reachable from it by a legal game.

5.2 A characterization of recurrect elements

Recurrent elements play an important role in the rotor-router dynamics.

Definition 5.2.1. A chip-and-rotor configuration (x, ϱ) is *recurrent*, if starting from (x, ϱ), there exists a nonempty legal rotor-routing game that leads back to (x, ϱ).

For the classical rotor-routing process, Holroyd et al. [21] gave a characterization for the recurrent configurations. To state their result, we need a definition.

Definition 5.2.2 (unicycle [21])**.** A *unicycle* is a single-chip-and-rotor configuration where the rotor subgraph contains a unique directed cycle, and the chip lies on this cycle.

Theorem 5.2.3 ([21, Theorem 3.8])**.** *If G is strongly connected, then the recurrent single-chip-and-rotor configurations are exactly the unicycles.*

In the following theorem, we generalize this result to the general rotor-routing game. One of the motivations for characterizing recurrent elements in the rotor-routing game is that for chip-firing, no characterization in known for recurrent chip-distributions on strongly connected digraphs.

Theorem 5.2.4. *For a strongly connected digraph G, a chip-and-rotor configuration (x, ϱ) is recurrent if and only if $x \geq 0$, and on each directed cycle in the rotor subgraph there is at least one vertex v with $x(v) > 0$.*

This theorem also shows, that it can be decided in polynomial time whether a chip-and-rotor configuration on a strongly connected digraph is recurrent. In contrast with this, the complexity of deciding whether a chip-distribution on a strongly connected digraph is recurrent is open.

We note that an analogue of the above condition is a necessary condition for a chip-distribution on a strongly connected digraph to be recurrent: A recurrent chip-distribution on a strongly connected digraph in non-negative on each vertex, and there is at least one vertex on every directed cycle that has a positive number of chips. This can proved in the following way: As the period vector of a strongly connected digraph is positive on every vertex, every vertex needs to be fired before we arrive back to the initial distribution. After a vertex is fired, it never again goes negative, hence a recurrent distribution needs to be nonnegative on every vertex. On the other hand, it follows from the proof of [7, Theorem 2.2], that in a recurrent chip-distribution, there is at least one vertex on every directed cycle, that has a positive number of chips.

Proof of Theorem 5.2.4. First we show the "only if" direction. Take a CRC (x, ϱ) which is recurrent. We claim that $x \geq 0$. It is enough to show that in any nonempty legal game that transforms (x, ϱ) back to itself, each vertex is routed at least once. Indeed, since we require legal routings, at the time a vertex is routed, it has positive number of chips, and it can never again become negative.

Since the initial and final rotor configurations are the same, each vertex is routed either zero times, or its rotor makes at least a full turn. In the later case, it passes a chip to each of its out-neighbors. Since the initial and final chip-distributions are also the same, if a vertex receives a chip, it needs to be routed. Hence each vertex reachable in directed path from a routed vertex is also routed. As the graph is strongly connected, this means that if a vertex is routed, all vertices are routed. This finishes the proof of $x \geq 0$.

We claim that there is at least one chip on each rotor cycle. Take a nonempty legal game that transforms (x, ϱ) back to itself. We have proved, that each vertex is routed at least once. Suppose that there is a cycle C in the rotor subgraph, such that $x(v) = 0$ for each $v \in V(C)$. Take the vertex $v \in V(C)$ that was last routed among the vertices of C. Since the final rotor configuration is ϱ, the last time v was routed, the chip moved to the head of $\varrho(v)$. Let us call this vertex w. Note that also $w \in V(C)$. Since originally $x(w) = 0$, the chip-distribution on w is never negative during the process, therefore after routing v, w has a positive number of chips. Since at the end w has zero chips, w needs to be routed after the last routing of v, which is a contradiction.

Now we show the "if" direction. It is enough to show that if for a CRC (x, ϱ), $x \geq 0$, and there is exactly one chip on each rotor cycle, then (x, ϱ) is recurrent. Indeed, if a CRC (x, ϱ) with $x \geq 0$ has at least one chip on each rotor cycle, then there is a chip-distribution x' with $x \geq x' \geq 0$ that has exactly one chip on each rotor- cycle. A legal game from (x', ϱ) is also a legal game from (x, ϱ), and if starting from (x', ϱ) it leads back to (x', ϱ), then starting from (x, ϱ) it leads back to (x, ϱ).

So take a CRC (x, ϱ) with $x \geq 0$ that has exactly one chip on each rotor cycle. Give a name to each chip: $c_1, \ldots, c_k$. Let their initial vertices be $v_1, \ldots, v_k$, respectively. In the rotor subgraph, each rotor cycle is in a different weakly connected component. Let the vertex set of the weakly connected component of v_i be V_i. Then $V(G) = V_1 \cup \cdots \cup V_k$. Moreover, $\{\varrho(v) : v \in V_i - v_i\}$ is an in-arborescence rooted at v_i, that spans V_i. Let us call this arborescence A_i.

Let us do the following procedure: For each vertex, remember how many times it has been routed (zero at the beginning). We call a vertex v *finished* at some time step, if it has been routed exactly $d^+(v) \cdot \mathrm{per}_G(v)$ times. Our procedure ensures that no vertex is routed more times than this. We start with routing the current vertex of c_1, until c_1 arrives at a finished vertex. We say that at this moment, c_1 gets finished. Then we start routing the vertex of c_2 until c_2 also arrives at a finished vertex, etc. until c_k also arrives at a finished vertex. Since we always route the vertex of a chip, we only make legal routings during this procedure. Also, no vertex v gets routed more than $d^+(v) \cdot \mathrm{per}_G(v)$ times, since whenever a chip arrives at a finished vertex, we stop routing it.

It is enough to show that during this procedure, each vertex v is routed exactly $d^+(v) \cdot \mathrm{per}_G(v)$ times. From this, it follows immediately that at the end of the process, we arrive back to (x, ϱ), as then each rotor makes some full turns, and each vertex v forwards $d^+(v) \cdot \mathrm{per}_G(v)$ chips, and receives $\sum_{u \in \Gamma^-(v)} \mathrm{per}_G(u)$ chips. The two quantities are equal because $L_G \mathrm{per}_G = \mathbf{0}_G$.

We show by induction, that by the time $c_1, \ldots, c_i$ are finished, all vertices in $V_1 \cup \cdots \cup V_i$ are finished, and c_j is in v_j for $j = 1, \ldots, i$. For $i = k$, this proves that (x, ϱ) is indeed recurrent.

For $i = 0$, the condition is meaningless. Suppose that the condition holds for some $i - 1$. We show that it also holds for i.

Since by induction hypothesis, $c_1, \ldots, c_{i-1}$ all got finished in their initial positions, before we start routing c_i, all vertices forwarded and received the same number of chips. Thus, while we are routing c_i, if at some moment c_i is at a vertex $v \neq v_i$, then each vertex $u \notin \{v, v_i\}$ received and forwarded the same number of chips, v received one more chips than forwarded, and v_i forwarded one more chips than received. If c_i is at v_i, then each vertex received and forwarded the same number of chips.

Suppose that the first finished vertex reached by c_i is v. Then v has been routed $d^+(v) \cdot \mathrm{per}_G(v)$ times. Since any in-neighbor u of v has been routed at most $d^+(u) \cdot \mathrm{per}_G(u)$ times, any such in-neighbor forwarded at most $\mathrm{per}_G(u)$ chips to v. Hence v received at most $\sum_{u\in\Gamma^-(v)} \mathrm{per}_G(u) = d^+(v)\mathrm{per}_G(v)$ chips. Thus when c_i first reached v as a finished vertex, v received at most as many chips, as it forwarded. Hence $v = v_i$.

We show that each vertex in V_i gets finished by the time c_i gets finished. Since A_i is an in-arborescence rooted at v_i spanning V_i, it is enough to show, that when a vertex v receives the chip for the $d^+(v)\mathrm{per}_G(v)$-th time, each of its in-neighbors in A_i are already finished.

Suppose that v has just received a chip for the $d^+(v)\mathrm{per}_G(v)$-th time. As it received at most $\sum_{u\in\Gamma^-(v)} \mathrm{per}_G(u) = \mathrm{per}_G(v)d^+(v)$ chips from its in-neighbors, to have equality, v must have received $\mathrm{per}_G(u)$ chips from each in-neighbor u. But for those in-neighbors u, where $\overrightarrow{uv} \in A_i$, the chip is forwarded towards v for the $d^+(u)$-th, $2d^+(u)$-th, $\ldots$ times, so from these vertices, a chip must have been forwarded $\mathrm{per}_G(u)d^+(u)$ times, hence they are indeed finished. □

Corollary 5.2.5. *On strongly connected digraphs, the recurrent configurations where the degree of the chip-distribution is one are exactly the unicycles.*

From the proof of Theorem 5.2.4, we can easily deduce a formula for the possible lengths of legal games transforming a recurrent CRC back to itself:

Proposition 5.2.6. *For a strongly connected digraph G, if a CRC (x, ϱ) is recurrent, then for any nonempty legal game that transform it back to itself, there is an integer $k \in \mathbb{N}$ such that each vertex v is routed $k \cdot d^+(v) \cdot \mathrm{per}_G(v)$ times. Moreover, there exists a legal game with $k = 1$.*

Proof. If a legal game transforms a CRC (x, ϱ) back to itself, then each rotor makes some full turns. Thus for each $v \in V(G)$, there exists some $z(v) \in \mathbb{N}$ such that v has been routed $d^+(v) \cdot z(v)$ times. Since the initial and final chip-distributions are also the same, each vertex gave and received the same number of chips. If a vertex u was routed $z(u) \cdot d^+(u)$ times, a vertex $v \in \Gamma^+(u)$ received $z(u)$ chips from it. Thus for each vertex v,

$$\sum_{u\in\Gamma^-(v)} z(u) = z(v) \cdot d^+(v).$$

Hence the vector z is an eigenvector of the Laplacian matrix with eigenvalue zero. Since L_G has a one-dimensional kernel, z is a multiple of per_G.

The construction in the proof of Theorem 5.2.4 shows that for any recurrent CRC, there exists a legal game with $k = 1$ that transforms it back to itself. □

For a unicycle, the rotor-routing game is deterministic, hence we obtain that it takes $\sum_{v\in V(G)} \mathrm{per}_G(v)d^+(v)$ steps for the rotor-router process to return to the initial configuration. This gives the following theorem, originally proved by Pham [34] using linear algebra.

Theorem 5.2.7. *For a strongly connected digraph G, the size of the orbit of any unicycle is $\sum_{v\in V(G)} \mathrm{per}_G(v)d^+(v)$.*

5.3 Linear equivalence

For the chip-firing game, linear equivalence is a linear-algebraic type, computationally well-behaved concept, that proves very useful for analyzing reachability questions. In this section, we generalize the concept of linear equivalence to the rotor-routing game. Then we apply it to analyzing reachability questions in the rotor-routing game, and to give a new interpretation of the rotor-routing action of the Picard group on the set of spanning in-arborescences for Eulerian digraphs. Using the connection between linear equivalence for chip-firing and for rotor-routing, we prove that the number of rotor-router unicycle-orbits equals the order of the Picard group.

In chip-firing, for strongly connected digraphs, linear equivalence is equivalent to reachability where we let non-legal firings to happen. For rotor-routing, we use the analogue of this characterization as definition. Let us call a non-necessarily legal routing an *unconstrained routing*.

Definition 5.3.1 (linear equivalence of chip-and-rotor configurations)**.** We define two configurations (x_1, ϱ_1) and (x_2, ϱ_2) on a strongly connected ribbon digraph to be linearly equivalent, if (x_2, ϱ_2) can be reached from (x_1, ϱ_1) by a sequence of unconstrained routings. We denote this by $(x_1, \varrho_1) \sim (x_2, \varrho_2)$.

Remark 5.3.2. The idea of analyzing the interplay between legal and non-legal rotor-routing games has appeared previously in some papers. See for example [12, 25].

Suppose we have an initial configuration, and a multiset of vertices to perform unconstrained routings at. Then the resulting configuration is independent of the order in which we perform the routings. Hence we can encode a sequence of unconstrained routings in a vector $r \in \mathbb{N}^{V(G)}$ such that $r(v)$ is the number of times vertex v has been routed. We call such a vector a *routing vector*.

Similarly, for chip-firing, by firing a vector $z \in \mathbb{N}^{V(G)}$, we mean firing each vertex v $z(v)$ times. This has the effect of adding $L_G z$ to the chip-distribution, independent of the order in which we perform the (not necessarily legal) firings.

Note the following connection between the effect of firings and routings:

Claim 5.3.3. *If a routing vector r is of the form $r = (d^+(v_1)\cdot z(v_1), \dots, d^+(v_n)\cdot z(v_n))$ for some $z \in \mathbb{N}^{V(G)}$, then routing r from a CRC (x, ϱ) leads to a CRC (x', ϱ), where x' is the chip-distribution we get after firing the vector z from x.*

Proposition 5.3.4. *Linear equivalence is an equivalence-relation.*

Proof. Reflexivity and transitivity are obvious. Let us prove symmetry. It is enough to prove that if we get (x_2, ϱ_2) from (x_1, ϱ_1) by one unconstrained routing at a vertex w, then (x_1, ϱ_1) can also be reached from (x_2, ϱ_2) by unconstrained routings. Take the following routing vector r:

$$r(v) = \begin{cases} d^+(v)\mathrm{per}_G(v) & \text{if } v \neq w, \\ d^+(w)\mathrm{per}_G(w) - 1 & \text{if } v = w. \end{cases}$$

Routing r from (x_2, ϱ_2) is equivalent to routing $(d^+(v_1) \cdot \mathrm{per}_G(v_1), \dots, d^+(v_n) \cdot \mathrm{per}_G(v_n))$ from (x_1, ϱ_1), that by Claim 5.3.3 leads to (x_1, ϱ_1). □

The following lemma shows the connection between the linear equivalence of chip-and-rotor configurations, and the linear equivalence of chip-distributions.

Lemma 5.3.5. *Let G be a strongly connected digraph. If ϱ is a rotor configuration, and $x_1, x_2 \in \mathrm{Chip}(G)$, then $x_1 \sim x_2$ if and only if $(x_1, \varrho) \sim (x_2, \varrho)$.*

Proof. Suppose that $x_1 \sim x_2$. This means that there exists $z \in \mathbb{Z}^{V(G)}$ such that $x_2 = x_1 + L_G z$. Moreover, we can suppose that z is nonnegative, otherwise we can add per_G to it sufficiently many times. From initial configuration (x_1, ϱ_1), route vertices according to the following routing vector: $(d^+(v_1) \cdot z(v_1), \dots, d^+(v_n) \cdot z(v_n))$. Then the resulting chip-moves are exactly the same as in chip-firing after firing the vector z, thus we arrive at the chip-distribution x_2. On the other hand, each rotor made some full turns, hence the final rotor configuration is again ϱ.

Now suppose that $(x_1, \varrho) \sim (x_2, \varrho)$. Fix a routing vector r witnessing the equivalence of (x_1, ϱ) and (x_2, ϱ). Then since the initial and the final rotor configurations are both ϱ, each rotor made some full turns, hence r must be of the form $r = (d^+(v_1) \cdot z(v_1), \dots, d^+(v_n) \cdot z(v_n))$ for some $z \in \mathbb{Z}^{V(G)}$. Then firing z induces the same chip-moves as routing r, hence $x_2 = x_1 + L_G z$, thus $x_1 \sim x_2$. □

Maybe the nicest property of the linear equivalence is that it is computationally well-behaved:

Proposition 5.3.6. *For given chip-and-rotor configurations (x_1, ϱ_1) and (x_2, ϱ_2), deciding whether $(x_1, \varrho_1) \sim (x_2, \varrho_2)$ holds can be done in polynomial time.*

Proof. For each vertex v, let $\alpha(v)$ be the number of out-edges from v such that $\varrho_1(v) < e \leq \varrho_2(v)$ in the cyclic order at v. If we route the routing vector α from (x_1, ϱ_1), we arrive at a configuration (y, ϱ_2), where y is some chip-distribution. This means at most $|E(G)|$ routings. For digraphs with multiple edges, $|E(G)|$ is not necessarily polynomial in the size of the input, but note that for each pair of vertices $u, v \in V(G)$, we can compute how many chips need to pass through the multi-edge $\overrightarrow{uv}$, and we can do this in time linear in the size of the description of the cyclic order at u. Hence we can compute the chip-distribution y in polynomial time.

As $(y, \varrho_2) \sim (x_1, \varrho_1)$, we have $(x_1, \varrho_1) \sim (x_2, \varrho_2)$ if and only if $(y, \varrho_2) \sim (x_2, \varrho_2)$, which by Lemma 5.3.5 is equivalent to $y \sim x_2$. This can be checked in polynomial time by Claim 1.2.7. □

5.3.1 Reachability questions

Notation. Let us denote by $(x_1, \varrho_1) \rightsquigarrow (x_2, \varrho_2)$ if (x_2, ϱ_2) can be reached from (x_1, ϱ_1) by a legal rotor-routing game.

In this section, we examine the reachability problem for rotor-routing from a computational aspect. As Theorem 5.2.7 shows, in the classical rotor-routing process, unicycle-orbits can have exponential size. Hence there exist configurations such that one is only reachable from the other by exponentially many routings. This shows that the question of deciding whether single-chip-and-rotor configuration can be reached from another one by a legal rotor-routing game is nontrivial. However, as the following proposition shows, if the target configuration is recurrent, the reachability problem is decidable in polynomial time. This result is an analogue of Theorem 2.4.3 which concerned the chip-firing game. The proof is also a complete analogue.

Theorem 5.3.7. *Let (x_1, ϱ_1) and (x_2, ϱ_2) be two chip-and-rotor configurations on a strongly connected digraph. If (x_2, ϱ_2) is recurrent, then $(x_1, \varrho_1) \rightsquigarrow (x_2, \varrho_2)$ if and only if $(x_1, \varrho_1) \sim (x_2, \varrho_2)$.*

Proof. The “only if” direction is obvious, since a sequence of legal routings is also a sequence of unconstrained routings.

Let us prove the “if” direction. By our assumption, (x_2, ϱ_2) is recurrent. Let $v_1, v_2, \dots, v_m$ be a sequence of vertices such that routing them in this order is a legal rotor-routing game that transforms (x_2, ϱ_2) back to itself. By Proposition 5.2.6, we can suppose that in this sequence each vertex v occurs $d^+(v)\mathrm{per}_G(v)$ times. As per_G is strictly positive, this means that each vertex occurs at least once.

By our assumption that $(x_1, \varrho_1) \sim (x_2, \varrho_2)$, there exists a routing vector $r \in \mathbb{N}^{V(G)}$ such that routing r transforms (x_2, ϱ_2) to (x_1, ϱ_1).

We proceed by induction on $\sum_{v\in V} r(v)$. If $\sum_{v\in V} r(v) = 0$, then $(x_1, \varrho_1) = (x_2, \varrho_2)$ hence we have nothing to prove. Otherwise let i be the smallest index such that $r(v_i) > 0$. Such an index exists since each vertex occurs in the sequence $v_1, \dots, v_m$. Starting from the configuration (x_2, ϱ_2) route at vertices $v_1, \dots, v_{i-1}$. These are all legal routings by definition. Let the resulting CRC be (x'_2, ϱ'_2).

We claim that routing $v_1, \dots, v_{i-1}$ from (x_1, ϱ_1) is also a legal game. Indeed, as $r(v_1) = \dots = r(v_{i-1}) = 0$, we can get (x_1, ϱ_1) from (x_2, ϱ_2) such that we do not route at $v_1, \dots, v_{i-1}$. Hence $x_1(v_j) \geq x_2(v_j)$ for $j = 1, \dots, i-1$. Also for the same reason, $\varrho_1(v_j) = \varrho_2(v_j)$ for $j = 1, \dots, i-1$. Hence from these two initial configurations, the routing of $v_1, \dots v_j$ for some $1 \leq j \leq i-1$ results in the same the chip-moves, and the same rotor-moves. Therefore after routing $v_1, \dots v_j$ for some $1 \leq j \leq i-1$, the rotors at $v_1, \dots v_{i-1}$ are the same in the two games, and the number of chips is greater or equal on these vertices in the game with initial configuration (x_1, ϱ_1). This shows that routing $v_1, \dots, v_{i-1}$ is indeed a legal game from initial configuration (x_1, ϱ_1). Let the resulting CRC be (x'_1, ϱ'_1). Then $(x_1, \varrho_1) \rightsquigarrow (x'_1, \varrho'_1)$.

Routing r from (x'_2, ϱ'_2) results in (x'_1, ϱ'_1), since we transformed (x_2, ϱ_2) and (x_1, ϱ_1) with the same routings. From (x'_2, ϱ'_2), route v_i (this is also a legal routing). Let the resulting CRC be (x''_2, ϱ''_2). Then for

$$r'(v) = \begin{cases} r(v) & \text{if } v \neq v_i, \\ r(v_i) - 1 & \text{if } v = v_i, \end{cases}$$

we have that routing r' from (x''_2, ϱ''_2) results in (x'_1, ϱ'_1), moreover, $\sum_{v\in V} r'(v) = \sum_{v\in V} r(v) - 1$.

We claim that (x''_2, ϱ''_2) is also a recurrent CRC. Indeed, routing vertices $v_{i+1}, \dots, v_m, v_1, \dots, v_i$ is a legal game that transforms (x''_2, ϱ''_2) to itself. Hence by induction hypothesis, $(x'_1, \varrho'_1) \rightsquigarrow (x''_2, \varrho''_2)$. As $(x_1, \varrho_1) \rightsquigarrow (x'_1, \varrho'_1)$ and $(x''_2, \varrho''_2) \rightsquigarrow (x_2, \varrho_2)$, we have $(x_1, \varrho_1) \rightsquigarrow (x_2, \varrho_2)$. □

Corollary 5.3.8. *Two unicycles $(\mathbf{1}_{v_1}, \varrho_1)$ and $(\mathbf{1}_{v_2}, \varrho_2)$ lie in the same rotor-router orbit if and only if $(\mathbf{1}_{v_1}, \varrho_1) \sim (\mathbf{1}_{v_2}, \varrho_2)$.*

Corollary 5.3.8 together with Proposition 5.3.6 gives us the following:

Proposition 5.3.9. *It can be decided in polynomial time whether two unicycles lie in the same rotor-router orbit.*

5.3.2 The number of unicycle-orbits

In this section, we relate the number of unicycle-orbits of the classical rotor-routing process to the order of the Picard group, using the connection between linear equivalence for chip-firing and for rotor-routing. Let us first state a technical lemma.

Lemma 5.3.10. *Each equivalence class of chip-and-rotor configurations where the degree of the divisor is at least one contains a recurrent configuration.*

Proof. Take a CRC (x, ϱ) with $\deg(x) = k \geq 1$, and start to play a legal rotor-routing game. As $\deg(x) \geq 1$, there exists a vertex v with $x(v) > 0$. Make a routing at v. As the resulting divisor still has degree k, there is once again a vertex with positive number of chips. For this reason, we can play a legal game as long as we wish. In a legal rotor-routing game a vertex can only loose chips if it is routed, but if it is routed, it is not allowed to go negative. Hence in a game with initial configuration (x, ϱ), on any vertex v, the number of chips is always at least $\min\{0, x(v)\}$. Hence the number of chips on any vertex is at most $\sum_{v \in V} \max\{x(v), 0\}$ at any time.

This means that there are only finitely many configurations we can reach from (x, ϱ); therefore, after finitely many steps, we get some configuration for the second time. This one will be recurrent. Moreover, we reached this recurrent configuration by a legal game from (x, ϱ), hence it is linearly equivalent to (x, ϱ). □

Corollary 5.3.11. *The number of CRC-equivalence classes of degree one equals the number of rotor-router unicycle-orbits.*

Proof. From Lemma 5.3.10, each CRC-equivalence classes of degree one contains a recurrent configuration, i.e. a unicycle. On the other hand, by Theorem 5.3.7 two recurrent configurations are linearly equivalent if and only if they lie in the same orbit. □

Proposition 5.3.12. *The order of* $\mathrm{Pic}^0(G)$ *equals the number of rotor-router unicycle-orbits.*

Proof. By Corollary 5.3.11, the number of rotor-router unicycle-orbits equals the number of CRC equivalence classes of degree one.

The order of the Picard group is by definition the number of equivalence classes of degree zero divisors by linear equivalence. The number of equivalence classes of degree zero divisors equals the number of equivalence classes of degree one divisors, as for an arbitrary fixed vertex v, $x \mapsto x + 1_v$ is a bijection between degree zero and degree one divisors that is compatible with the linear equivalence.

Hence we need to show that the number of CRC-equivalence classes of degree one equals the number of divisor equivalence classes of degree one. Fix a rotor configuration ϱ. As each CRC equivalence class contains at least one CRC with rotor configuration ϱ (from an arbitrary CRC of the class we can route each vertex v until its rotor edge becomes $\varrho(v)$), it is enough to count the maximal number of pairwise non-equivalent degree one CRCs with rotor configuration ϱ. By Lemma 5.3.5, this number is exactly the number of equivalence classes of degree one divisors. □

5.3.3 The rotor-router action

In this section, let G be an Eulerian digraph. Holroyd et al. [21] defined a group action of the Picard group on the spanning in-arborescences of the graph, using the rotor-router operation. For Eulerian digraphs, we give an interpretation of this group action in terms of the linear equivalence classes of divisor-and-rotor configurations.

Notation. We denote the set of spanning in-arborescences of G rooted at r by $\text{Arb}(G, r)$. For a $T \in \text{Arb}(G, r)$, and a vertex $v \neq r$, let us denote by $T(v)$ the edge leaving v.

For any fixed edge $\overrightarrow{rw}$, the following mapping ϱ is a rotor configuration with exactly one cycle:

$$\varrho(v) = \begin{cases} T(v) & \text{if } v \neq r, \\ \overrightarrow{rw} & \text{if } v = r. \end{cases}$$

Let us denote $\varrho = T \cup \overrightarrow{rw}$.

Definition 5.3.13 (Rotor-router action, [21])**.** The *rotor-router action* is defined with respect to a base vertex $r \in V(G)$ that we call the *root*. It is a group action of $\text{Pic}^0(G)$ on the spanning in-arborescences of G rooted at r. We denote by $x_r(T)$ the image of a $T \in \text{Arb}(G, r)$ at the action of the divisor $x \in \text{Div}^0(G)$.

$x_r(T)$ is defined as follows: Choose a divisor $x' \sim x$ such that $x'(v) \geq 0$ for each $v \neq r$. Such an x' can easily be seen to exist. Fix any out-edge $\overrightarrow{rw}$ of r. Let $\varrho = T \cup \overrightarrow{rw}$. Start a legal rotor-routing game from (x', ϱ), such that r is not allowed to be routed. Continue until each chip arrives at r. Holroyd et al. [21] shows that this procedure ends after finitely many steps, and in the final configuration $(\mathbf{0}_G, \varrho')$, the edges $\{\varrho'(v) : v \in V(G) - r\}$ form a spanning in-arborescence of G rooted at r. $x_r(T)$ is defined to be this arborescence.

Holroyd et al. [21] shows that, $x_r(T)$ is well defined, i.e. the definition does not depend on our choice of w, x' and on the choice of the legal game. From this, it also follows that this is indeed an action of $\text{Pic}^0(G)$ on $\text{Arb}(G, r)$, i.e. $x_r(T) = x'_r(T)$ if $x \sim x'$. The fact that this is a group action can be easily seen from the definition.

Now for the case of Eulerian digraphs, we give an alternative definition of this group action using the notion of linear equivalence. First we need a technical lemma.

Notation. Let us call a chip-and-rotor configuration $\overrightarrow{rw}$*-good*, if it is of the form $(\mathbf{0_G}, \varrho)$, where $\varrho(r) = \overrightarrow{rw}$, and the edges $\{\varrho(v) : v \in V(G) - r\}$ form a spanning in-arborescence of G rooted at r.

Lemma 5.3.14. *For a strongly connected Eulerian digraph G, vertex $r \in V(G)$ and edge $\overrightarrow{rw} \in E(G)$, in each CRC equivalence class of degree zero, there is exactly one $\overrightarrow{rw}$-good CRC.*

Proof. Take a CRC equivalence class C of degree zero. Let

$$C + \mathbf{1}_r = \{(x + \mathbf{1}_r, \varrho) : (x, \varrho) \in C\}.$$

This is a CRC equivalence class of degree one. A configuration $(\mathbf{0}_G, \varrho) \in C$ is $\overrightarrow{rw}$-good if and only if $(\mathbf{1}_r, \varrho) \in C + \mathbf{1}_r$ is a unicycle with $\varrho(r) = \overrightarrow{rw}$. Thus, it is enough to show, that in each CRC equivalence class of degree one, there is exactly one unicycle $(\mathbf{1}_r, \varrho)$ where $\varrho(r) = \overrightarrow{rw}$.

By Lemma 5.3.10, there exists a recurrent element in each CRC equivalence class of degree one, which is a unicycle by Corollary 5.2.5. If we run the rotor-router process from this unicycle until it returns to the initial position, each vertex v is visited $d^+(v)$ times by the chip. Therefore, the chip reaches r $d^+(r)$ times, and during these visits, the rotor at r turns around. Hence there will be a moment, when the chip is at r, and the rotor at r is $\overrightarrow{rw}$. As the rotor-router process takes unicycles to unicycles [21, Lemma 3.3], this is going to be a unicycle of the form $(\mathbf{1}_r, \varrho)$ where $\varrho(r) = \overrightarrow{rw}$.

Now suppose there are two linearly equivalent unicycles $(\mathbf{1}_r, \varrho_1)$ and $(\mathbf{1}_r, \varrho_2)$ with $\varrho_1(r) = \varrho_2(r) = \overrightarrow{rw}$. Then by Corollary 5.3.8, they lie in the same rotor-router orbit. But we get all elements of the orbit of $(\mathbf{1}_r, \varrho_1)$ by running the rotor-router process until it arrives back to $(\mathbf{1}_r, \varrho_1)$. During this process, the chip visits the vertex r only $d^+(r)$ times, hence the only unicycle equivalent to $(\mathbf{1}_r, \varrho_1)$ where the chip is at r and the rotor at r is $\overrightarrow{rw}$ is itself. □

Remark 5.3.15. For non-Eulerian digraphs, the above lemma is false. That is the reason why we can only handle the case of Eulerian digraphs with the notion of linear equivalence.

Definition 5.3.16 (Alternative definition of the rotor-router action on Eulerian digraphs)**.** Let a divisor x of degree zero act on a spanning in-arborescence T rooted at $r \in V(G)$ as follows:

Fix a vertex w such that $\overrightarrow{rw} \in E(G)$. Let $\varrho = T \cup \overrightarrow{rw}$. Let $(\mathbf{0}_G, \varrho')$ be the unique $\overrightarrow{rw}$-good chip-and-rotor configuration linearly equivalent to (x, ϱ). Let T' be the spanning in-arborescence $\{\varrho'(v) : v \in V(G) - r\}$. Then let $T^x = T'$.

Proposition 5.3.17. *$x_r(T) = T^x$ for any choice of $r \in V(G)$, $x \in \mathrm{Div}^0(G)$ and $T \in \mathrm{Arb}(G, r)$.*

Proof. Let $\varrho = T \cup \overrightarrow{rw}$. In the construction of Definition 5.3.13, we obtain a CRC $(\mathbf{0_G}, \varrho')$ linearly equivalent to (x, ϱ), where $\{\varrho'(v) : v \in V(G) - r\}$ is a spanning in-arborescence. Moreover, since r is not routed during the process, $\varrho'(r) = \overrightarrow{rw}$. Hence $(\mathbf{0_G}, \varrho')$ is $\overrightarrow{rw}$-good, so both definitions give the spanning in-arborescence $\{\varrho'(v) : v \in V(G) - r\}$. □

Corollary 5.3.18. *For an Eulerian digraph G, given two spanning in-arborescences $T_1, T_2 \in \mathrm{Arb}(G,r)$ and a divisor $x \in \mathrm{Div}^0(G)$, it can be decided in polynomial time whether $x_r(T_1) = T_2$.*

Proof. One needs to check whether for an out-edge $\overrightarrow{rw}$ of r, $(x, T_1 \cup \overrightarrow{rw}) \sim (\mathbf{0}_G, T_2 \cup \overrightarrow{rw})$. This can be done in polynomial time by Proposition 5.3.6. □

Base-point independence of the rotor-router action

Let us turn to undirected graphs (that we simultaneously imagine as bidirected graphs). For undirected graphs, the spanning in-arborescences with root r are in one-to-one correspondence with the spanning trees. Therefore, we can think of the rotor-router action with base point r as an action on the spanning trees of the graph. Since now the rotor-router action with any base vertex acts on the same set of objects, one can ask, for which ribbon graphs is the action independent of the base vertex. Chan, Church and Grochow [10] shows that the rotor-router action is independent of the base vertex if and only if the ribbon graph is planar. (Planarity for a ribbon graph means that the ribbon graph structure gives a combinatorial embedding of the graph into the plane.) Their proof proceeds in two steps. First they show the following:

Notation. For a rotor configuration ϱ, let $\overleftarrow{\varrho}$ be the rotor configuration in which each rotor cycle is reversed, and all other rotors are left the same. Let us call this the *reversal* of the rotor configuration. See Figure 1 for an example.

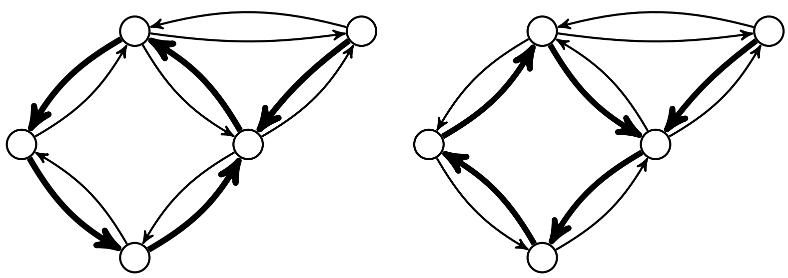

Figure 5.2: A rotor configuration and its reversal. The rotor edges are drawn by thick lines.

Proposition 5.3.19. *[10] A connected ribbon graph G without loops is planar if and only if for any unicycle $(\mathbf{1}_v, \varrho)$, $(\mathbf{1}_v, \varrho) \rightsquigarrow (\mathbf{1}_v, \overleftarrow{\varrho})$.*

The second step in their proof is to show that the rotor-router action is independent of the base vertex if and only if for any unicycle $(\mathbf{1}_v, \varrho)$, $(\mathbf{1}_v, \varrho) \rightsquigarrow (\mathbf{1}_v, \overleftarrow{\varrho})$. We give a simple proof for this second statement using the interpretation of the rotor-routing action in terms of the equivalence classes of chip-and-rotor configurations.

Theorem 5.3.20. *[10] The rotor-router action on an undirected graph is independent of the base vertex if and only if for any unicycle* $(\mathbf{1}_v, \varrho)$, $(\mathbf{1}_v, \varrho) \rightsquigarrow (\mathbf{1}_v, \overleftarrow{\varrho})$.

Proof. From Theorem 5.3.7, the fact that for any unicycle $(\mathbf{1}_v, \varrho)$, $(\mathbf{1}_v, \varrho) \rightsquigarrow (\mathbf{1}_v, \overleftarrow{\varrho})$ is equivalent to the fact that for any unicycle $(\mathbf{1}_v, \varrho)$, $(\mathbf{1}_v, \varrho) \sim (\mathbf{1}_v, \overleftarrow{\varrho})$, which is in turn equivalent to the fact that

$$\text{for any CRC } (0_G, \varrho), \text{ where } \varrho \text{ has exactly one cycle, } (\mathbf{0}_G, \varrho) \sim (\mathbf{0}_G, \overleftarrow{\varrho}).$$

Note also, that this condition implies $(x, \varrho) \sim (x, \overleftarrow{\varrho})$ for any $x \in \mathbb{Z}^V$ where ϱ has exactly one cycle.

First we show the "if" part. Since our graph is connected, it is enough to show, that for any two adjacent vertices $v, w \in V(G)$, the rotor-router action with base vertex v equals the rotor-router action with base vertex w.

For a vertex u and spanning tree T, let us denote by T_u the spanning in-arborescence rooted at u that we get by orienting each edge of T towards u.

Take any spanning tree T of G, and a divisor $x \in \mathrm{Div}^0(G)$. $T_v \cup \overrightarrow{vw}$ is a rotor-configuration, since v and w are adjacent. By definition, $x_v(T) = T'$ where $(\mathbf{0}_G, T'_v \cup \overrightarrow{vw}) \sim (x, T_v \cup \overrightarrow{vw})$. As $\overleftarrow{T_v \cup \overrightarrow{vw}} = T_w \cup \overrightarrow{wv}$ and $\overleftarrow{T'_v \cup \overrightarrow{vw}} = T'_w \cup \overrightarrow{wv}$, we have $(\mathbf{0}_G, T'_v \cup \overrightarrow{vw}) \sim (\mathbf{0}_G, T'_w \cup \overrightarrow{wv})$ and $(x, T_v \cup \overrightarrow{vw}) \sim (x, T_w \cup \overrightarrow{wv})$. Hence by transitivity, $(\mathbf{0}_G, T'_w \cup \overrightarrow{wv}) \sim (x, T_w \cup \overrightarrow{wv})$. Thus $x_w(T) = T'$.

Now we show the "only if" part. Suppose that there exists a chip-and-rotor configuration $(\mathbf{0}_G, \varrho)$, where ϱ has exactly one cycle, such that $(\mathbf{0}_G, \varrho) \not\sim (\mathbf{0}_G, \overleftarrow{\varrho})$. We show that in this case there exists $v, w \in V(G)$, $x \in \mathrm{Div}^0(G)$ and a spanning tree T, such that $x_v(T) \neq x_w(T)$.

Let v be a vertex on the cycle of ϱ, and let w be the vertex such that $\varrho(v) = \overrightarrow{vw}$. Then w is also on the cycle. Let T be the spanning tree we get by forgetting the orientations of $\{\varrho(u) : u \in V(G) - v\}$. Take $(\mathbf{0}_G, \varrho)$, and route at w until the rotor at w becomes $\overrightarrow{wv}$. Let the chip-and-rotor configuration at this moment be (x, ϱ'). Then $(x, \varrho') \sim (\mathbf{0}_G, \varrho)$ by its construction. Let T' be the subgraph that we get by forgetting the orientations of $\{\varrho'(u) : u \in V(G) - w\} = \{\varrho(u) : u \in V(G) - w\}$. T' is a spanning tree, since ϱ has one cycle, and w is on this cycle. Note that $T'_v \cup \overrightarrow{vw} = T'_w \cup \overrightarrow{wv}$.

As $(x, T'_v \cup \overrightarrow{vw}) = (x, \varrho') \sim (\mathbf{0}_G, \varrho) = (\mathbf{0}_G, T_v \cup \overrightarrow{vw})$, $x_v(T') = T$. On the other hand, if $x_w(T') = T$ were true, that would mean, using also the previous equivalence, that $(\mathbf{0}_G, \varrho) \sim (x, T'_v \cup \overrightarrow{vw}) = (x, T'_w \cup \overrightarrow{wv}) \sim (\mathbf{0}_G, T_w \cup \overrightarrow{wv}) = (\mathbf{0}_G, \overleftarrow{\varrho})$, contradicting our assumption. □

Bibliography

[1] Omid Amini and Madhusudan Manjunath. Riemann-Roch for sub-lattices of the root lattice A_n. *Electron. J. Combin.*, 17(1):Research Paper 124, 50, 2010.

[2] Arash Asadi and Spencer Backman. Chip-firing and Riemann-Roch theory for directed graphs. *Preprint*, `http://arxiv.org/abs/1012.0287`, 2011.

[3] 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*, 125:167–198, 1997.

[4] Spencer Backman. Riemann-Roch theory for graph orientations. *Preprint*, `http://arxiv.org/abs/1401.3309`, 2014.

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

[6] Matthew Baker and Farbod Shokrieh. Chip-firing games, potential theory on graphs, and spanning trees. *J. Combin. Theory Ser. A*, 120(1):164–182, 2013.

[7] Anders Björner and László Lovász. Chip-firing games on directed graphs. *J. Algebraic Combin.*, 1(4):305–328, 1992.

[8] Anders Björner, László Lovász, and Peter W. Shor. Chip-firing games on graphs. *European J. Combin.*, 12(4):283–291, 1991.

[9] B. Bond and L. Levine. Abelian networks I. foundations and examples. *SIAM Journal on Discrete Mathematics, to appear*, 2016. arXiv:1309.3445.

[10] M. Chan, T. Church, and J. Grochow. Rotor-routing and spanning trees on planar graphs. *Int. Math. Res. Not.*, 11:3225–3244, 2015.

[11] Robert Cori and Yvan le Borgne. On computation of Baker and Norine's rank on complete graphs. *Electronic Journal of Combinatorics*, 23(1):P1.31, 2016.

[12] Arnaud Dartois and Dominique Rossin. The height-arrow model. In *16th Formal Power Series and Algebraic Combinatorics (FPSAC'04)*, 2004.

[13] D. Dhar. Self-organized critical state of sandpile aoutomaton models. *Physical Review Letters*, 64(14):1613–1616, 1990.

[14] Ioana Dumitriu, Prasad Tetali, and Peter Winkler. On Playing Golf with Two Balls. *SIAM J. Discret. Math.*, 16(4):604–615, 2003.

[15] A. Engel. The probabilistic abacus. *Educ. Stud. in Math.*, 7:1–22, 1975.

[16] Matthew Farrell and Lionel Levine. CoEulerian graphs. *Proc. Amer. Math. Soc.*, 144:2847–2860, 2016.

[17] Tibor Gallai. On directed paths and circuits. *Theory of graphs (Proc. Colloq. Tihany, 1966)*, 1968.

[18] Andreas Gathmann and Michael Kerber. A Riemann-Roch theorem in tropical geometry. *Math. Zeitschrift*, 259:217–230, 2008.

[19] Martin Grötschel, László Lovász, and Alexander Schrijver. *Geometric Algorithms and Combinatorial Optimization.* Springer-Verlag Berlin Heidelberg, 1988.

[20] Jan Hladký, Daniel Král', and Serguei Norine. Rank of divisors on tropical curves. *J. Combin. Theory Ser. A*, 120(7):1521–1538, 2013.

[21] Alexander E. Holroyd, Lionel Levine, Karola Mészáros, Yuval Peres, James Propp, and David B. Wilson. Chip-firing and rotor-routing on directed graphs. In Vladas Sidoravicius and Maria Eulália Vares, editors, *In and Out of Equilibrium 2*, volume 60 of *Progress in Probability*, pages 331–364. Birkhäuser Basel, 2008.

[22] Alexander E. Holroyd and James Propp. Rotor walks and Markov chains. In *ALGORITHMIC PROBABILITY AND COMBINATORICS, American Mathematical Society*, pages 105–126, 2010.

[23] Bálint Hujter, Viktor Kiss, and Lilla Tóthmérész. On the complexity of the chip-firing reachability problem. *Accepted for Proceedings of the American Mathematical Society.* arXiv:1507.03209.

[24] Bálint Hujter and Lilla Tóthmérész. Chip-firing based methods in the Riemann-Roch theory of directed graphs. *Preprint*, `https://arxiv.org/abs/1511.03568`, 2015.

[25] Wouter Kager and Lionel Levine. Rotor-router aggregation on the layered square lattice. *Electronic Journal of Combinatorics*, 17:R152, 2010.

[26] V. Kiss and L. Tóthmérész. Chip-firing games on Eulerian digraphs and **NP**-hardness of computing the rank of a divisor on a graph. *Discrete Appl. Math.*, 193:48–56, 2015.

[27] László Lovász and Peter Winkler. Mixing of random walks and other diffusions on a graph. In Peter Rowlinson, editor, *Surveys in Combinatorics*, pages 119–154. Cambridge University Press, New York, NY, USA, 1995.

[28] Ye Luo. Rank-determining sets of metric graphs. *J. Combin. Theory Ser. A*, 118(6):1775–1793, 2011.

[29] Madhusudan Manjunath. The rank of a divisor on a finite graph: geometry and computation. *Preprint*, `http://arxiv.org/abs/1111.7251`, 2011.

[30] Criel Merino López. Chip firing and the Tutte polynomial. *Annals of Combinatorics*, 1(1):253–259, 1997.

[31] Grigory Mikhalkin and Ilia Zharkov. Tropical curves, their Jacobians and theta functions. *Preprint*, `http://arxiv.org/abs/math/0612267`, 2006.

[32] David Perkinson, Jacob Perlman, and John Wilmes. Primer for the algebraic geometry of sandpiles. *Contemp. Math.*, 605, 2014.

[33] Kévin Perrot and Trung Van Pham. Feedback Arc Set Problem and NP-Hardness of Minimum Recurrent Configuration Problem of Chip-Firing Game on Directed Graphs. *Ann. Comb.*, 19(1):1–24, 2015.

[34] Trung Van Pham. Orbits of rotor-router operation and stationary distribution of random walks on directed graphs. *Advances in Applied Mathematics*, 70:45–53, 2015. arXiv:1403.5875.

[35] V. B. Priezzhev, Deepak Dhar, Abhishek Dhar, and Supriya Krishnamurthy. Eulerian walkers as a model of self-organized criticality. *Phys. Rev. Lett.*, 77:5079–5082, 1996.

[36] Y. Rabani, A. Sinclair, and R. Wanka. Local divergence of Markov chains and the analysis of iterative load-balancing schemes. In *Foundations of Computer Science*, pages 694–703. IEEE, 1998.

[37] Gábor Tardos. Polynomial bound for a chip firing game on graphs. *SIAM J. Discrete Math*, 1:397–398, 1988.

[38] Robert Tarjan. Depth first search and linear graph algorithms. *SIAM Journal on Computing*, 1972.

[39] Mikkel Thorup. Firing games. Technical Report 94/15, University of Copenhagen, 1994.

[40] Lilla Tóthmérész. Algorithmic aspects of rotor-routing and the notion of linear equivalence. *Preprint*, `https://arxiv.org/abs/1507.08235`, 2015.

[41] Josse van Dobben de Bruyn and Dion Gijswijt. Treewidth is a lower bound on graph gonality. *Preprint*, arXiv:1407.7055, 2014.

[42] Josse van Dobben de Bruyne. Reduced divisors and gonality in finite graphs, Bachelor's thesis, Mathematisch Instituut, Universiteit Leiden, 2012.

[43] John S. Wilmes. Algebraic invariants of sandpile graphs, 2010. Bachelor's thesis, The Division of Mathematics and Natural Sciences, Reed College.

Summary

The thesis is about three related topics, chip-firing, graph divisor theory and rotor-routing. Chip-firing and rotor-routing are simple, yet interesting diffusion processes on graphs, that have connections to many parts of mathematics, including the Tutte polynomial, graph orientations and random walks. Graph divisor theory is a discrete analogue of the divisor theory of Riemann surfaces, that has strong connections to chip-firing.

Chip-firing is a one player game played on a digraph, where on each vertex, there is an integer number of chips. If a vertex has at least as many chips, as its outdegree, it is allowed to "fire", i.e. to pass a chip to its neighbors along each out-edge incident to it. In Chapter 2, we investigate the chip-firing reachability question: Given two chip-distributions x and y on a digraph G, decide whether there exists a legal game transforming x to y. We show that this problem is in **co-NP**, and for digraphs with polynomial period length, it is in **P** (even if there are multiple edges). Moreover, we show that if the target distribution is recurrent (i.e. reachable from itself by a nonempty legal game), then a trivial necessary condition is sufficient for the reachability problem. These results are joint work with Bálint Hujter and Viktor Kiss.

Graph divisor theory is a discrete analogue of the divisor theory of Riemann surfaces. Divisors on graphs, and the Picard group of a graph have been defined by Bacher et al. In 2007, Baker and Norine defined the rank of a graph divisor, and proved the analogue of the Riemann–Roch theorem for this notion. It remained an intriguing open question whether the rank of a graph divisor can be computed in polynomial time. In Chapter 3, we show that computing the rank of a divisor on a graph is **NP**-hard, even for simple graphs. The results of this chapter are joint work with Viktor Kiss.

The Riemann–Roch theorem of Baker and Norine inspired the research for Riemann–Roch theorems in similar settings, including tropical curves, lattices, and directed graphs. In Chapter 4, we prove a Riemann–Roch inequality for Eulerian digraphs, that generalizes the Riemann–Roch theorem of Baker and Norine, and which is a stronger version of an earlier result of Amini and Manjunath. We also investigate the natural Riemann–Roch property introduced by Asadi and Backman, proving that an Eulerian digraph has the natural Riemann–Roch property if and only if it corresponds to an undirected graph. These results are joint work with Bálint Hujter.

Rotor routing is a one-player game on a digraph, that can be thought of as a refined version of chip-firing. An important application of this game is that one can define a group action of the Picard group on the spanning in-arborescences through rotor-routing (Holroyd et al.). In Chapter 5, we characterize recurrent elements for the rotor-routing game. Also, we define the linear equivalence of configurations, and for Eulerian digraphs, give an interpretation of the rotor-routing action in terms of linear equivalence.

Összefoglalás

A tézis témája három egymáshoz szorosan kötődő terület, a chip-firing, a rotor-routing és a diszkrét divizor-elmélet. A chip-firing és a rotor-routing két egyszerű dinamikus folyamat gráfokon, melyeknek a matematika sok területével érdekes kapcsolata van (például a Tutte polinommal, gráf-irányításokkal illetve véletlen sétákkal). A diszkrét divizor-elmélet a Riemann-felületek divizor-elméletének diszkrét analógja, mely szoros kapcsolatban áll a chip-firing játékkal.

A chip-firing tekinthető egy irányított gráfon játszott egyszemélyes játéknak. Minden csúcson adott egy egész számú chip. Ha egy csúcson legalább ki-fokszámnyyi chip van, akkor a csúcs "lőhet", azaz minden rá-illeszkedő ki-élen átadhat egy-egy chip-et. A második fejezetben a chip-firing elérhetőségi kérdést vizsgáljuk: Adott x és y chip-kiosztások esetén létezik-e x-et y-ba transzformáló érvényes játék? Megmutatjuk hogy ez a kérdés **co-NP**-ben van, továbbá polinomiális periódushosszú gráfok esetén **P**-ben van (akkor is ha vannak többszörös élek). Ezen kívül megmutatjuk, hogy ha az y chip-kiosztás rekurrens (azaz elérhető saját magából egy nemüres játékkal), akkor az elérhetőség egy triviális szükséges feltétele elégséges is. Ezen eredmények Hujter Bálinttal és Kiss Viktorral közösek.

A diszkrét divizor-elmélet a Riemann-felületek divizor-elméletének diszkrét analógja. A gráfdivizorokat illetve egy gráf Picard csoportját Bacher és szerzőtársai vezették be 1997-ben. Később, 2007-ben Baker és Norine definiálta egy gráfdivizor rangját, majd belátták a Riemann–Roch tétel analógját. Nyitott kérdés maradt hogy egy gráfdivizor rangja kiszámítható-e polinom időben. A harmadik fejezetben megmutatjuk hogy egy gráfdivizor rangjának kiszámítása **NP**-nehéz, még egyszerű irányítatlan gráf esetén is. Ezen fejezet eredményei Kiss Viktorral közösek.

Baker és Norine Riemann–Roch tétele sok kutatót motivált Riemann–Roch jellegű tételek keresésére hasonló modellekben, például trópusi görbéken, rácsokon vagy irányított gráfokon. A negyedik fejezetben belátunk egy Riemann–Roch egyenlőtlenséget irányított Euler gráfokra, mely Amini és Manjunath egy korábbi egyenlőtlenségének erősebb (éles) változata. Ez az egyenlőtlenség irányítatlan gráfokra visszaadja Baker és Norine Riemann–Roch tételét. Ezen kívül vizsgáljuk az Asadi és Backman által bevezetett úgynevezett természetes Riemann–Roch tulajdonságot, és belátjuk, hogy egy irányított Euler gráf pontosan akkor rendelkezik természetes Riemann–Roch tulajdonsággal, ha egy irányítatlan gráfból kapható az irányítatlan éleket két-két szembeirányított élre cserélve. Ezen fejezet eredményei Hujter Bálinttal közösek.

A rotor-routing szintén egy irányított gráfon játszott egyszemélyes játék, mely a chip-firing finomításának tekinthető. A játék egy alkalmazása, hogy segítségével definiálható a Picard csoport egy hatása a gráf adott gyökerű be-fenyőin (Holroyd és szerzőtársai). Az ötödik fejezetben karakterizációt adunk a rotor-routing rekurrens konfigurációira. Majd definiáljuk a rotor-routing konfigurációinak lineáris ekvivalenciáját, és ezen fogalom segítségével Euler gráfok esetére adunk egy ekvivalens definíciót a rotor-hatásra.

ADATLAP
a doktori értekezés nyilvánosságra hozatalához*

I. A doktori értekezés adatai
A szerző neve: Tóthmérész Lilla
MTMT-azonosító: 10054581
A doktori értekezés címe és alcíme: The chip-firing game
DOI-azonosító: 10.15476/ELTE.2016.187
A doktori iskola neve: Matematika Doktori Iskola
A doktori iskolán belüli doktori program neve: Alkalmazott Matematika Doktori Program
A témavezető neve és tudományos fokozata: Király Zoltán, PhD
A témavezető munkahelye: ELTE TTK Matematikai Intézet, Számítógéptudományi Tanszék

II. Nyilatkozatok
1. A doktori értekezés szerzőjeként
a) hozzájárulok, hogy a doktori fokozat megszerzését követően a doktori értekezésem és a tézisek nyilvánosságra kerüljenek az ELTE Digitális Intézményi Tudástárban. Felhatalmazom a Természettudományi kar Dékáni Hivatal Doktori, Habilitációs és Nemzetközi Ügyek Csoportjának ügyintézőjét, hogy az értekezést és a téziseket feltöltse az ELTE Digitális Intézményi Tudástárba, és ennek során kitöltse a feltöltéshez szükséges nyilatkozatokat.
2. A doktori értekezés szerzőjeként kijelentem, hogy
a) az ELTE Digitális Intézményi Tudástárba feltöltendő doktori értekezés és a tézisek saját eredeti, önálló szellemi munkám és legjobb tudomásom szerint nem sértem vele senki szerzői jogait;
b) a doktori értekezés és a tézisek nyomtatott változatai és az elektronikus adathordozón benyújtott tartalmak (szöveg és ábrák) mindenben megegyeznek.
3. A doktori értekezés szerzőjeként hozzájárulok a doktori értekezés és a tézisek szövegének plágiumkereső adatbázisba helyezéséhez és plágiumellenőrző vizsgálatok lefuttatásához.

Kelt: 2016. december 13.

Tóthmérész Lilla
a doktori értekezés szerzőjének aláírása

*ELTE SZMSZ SZMR 12. sz. melléklet