Linear Extensions of Rotor-Routing in Directed Graphs: Reachability Problems

David Auger   Pierre Coucheney   Kossi Roland Etse

Submitted: Nov 15, 2024; Accepted: Mar 3, 2026; Published: Apr 14, 2026

Note: When this paper was first accepted for publication the authors uploaded an old version which was then published on Apr 14, 2026. The correct version was then published on May 10, 2026. Here, to preserve the history, we preserve the version that was first published.

Abstract

We introduce a linear extension of the rotor-routing model in directed graphs, akin to the sandpile model and vector addition systems, together with new rotor mechanisms that extend standard cyclic rotors. In this framework, rotor-routing is interpreted as the simultaneous movement of particles accross two coupled graphs, involving both vertex and arc-based particles. The standard, combinatorial rotor-routing of positive particles (legal routing) based on rotor-configurations, then becomes a special case of a linear equivalence. We give comprehensive reachability results characterizing legal routings among linear equivalences, expanding on previous results, and settle the algorithmic complexities associated with these problems.

Mathematics Subject Classifications: 05C20,05C25,05C50,68Q17,37B15

1 Introduction and related works

1.1 The rotor-routing model

The *rotor-routing model*, also known as *rotor-router* or *rotor-walk* (see [13] for a comprehensive overview), was first introduced by Priezzhev in 1996 as Eulerian walkers [21, 20]. Independently, Propp and Wilson in 1996 proposed it for generating random spanning trees in graphs [25], and it relates to Yanovski et al.'s patrolling algorithms [26]. This model is closely associated with *abelian sandpiles* (or *chip-firing*) model [2, 13]. For general introductions to abelian sandpiles, see [9] and [13], and Section 1.2 hereafter.

In the basic rotor-routing model, a single particle (or pebble, chip) moves deterministically on the vertices of a graph. When the particle is at a vertex v, it follows a

Laboratoire DAVID, Université de Versailles-Saint-Quentin-En-Yvelines, France,
(david.auger@uvsq.fr, pierre.coucheney@uvsq.fr, kossi-roland.etse@uvsq.fr)

 https://doi.org/10.37236/13580

predetermined sequence of arcs: the first time it visits v, it takes arc 1; the next time, arc 2; and so on, cycling back to arc 1 after all arcs have been used. This deterministic movement is applied at every vertex. On average, each arc (or transition) is crossed with the same frequency as in a random walk.

This simple rule defines the rotor-routing model, which exhibits many intriguing properties. For example, in an eulerian and strongly connected graph, the particle will eventually circulate repeatedly on an eulerian tour [21]. In sufficiently connected graphs, the particle will ultimately reach target vertices known as sinks, though the exploration time can be exponential in the number of vertices. The problem of determining the first sink reached, given a starting arc configuration and an initial vertex, is known as the ARRIVAL problem. Defined in [5], it was shown to belong to the complexity class NP ∩ co-NP. While it is not known to be in P, it was proved to lie in the smaller class UP ∩ co-UP [10]. Additionally, a subexponential algorithm based on computing a Tarski fixed point was proposed in [11].

1.2 Abelian sandpiles and rotor-routing as special Vector Addition Systems

In this paper, we will show how, similarly to the abelian sandpiles model, we may envision the model of rotor-routing a special case of Vector Addition Systems (VAS) [15], which are for reachability issues equivalent to Petri Nets.

To define a VAS, we need a finite set of *transitions* $T \subset \mathbb{Z}^d$ where $d \geqslant 1$. The *states* of the system are elements of $\mathbb{N}^d$. A transition $t \in T$ from a state $v \in \mathbb{N}^d$ is *legal*, if $v + t \geqslant 0$, i.e. $v + t$ is a state. This defines an elementary *legal transition* from v to $v + t$, and the reachability problem consists in deciding the existence of a finite sequence of legal transitions (t_i) from some state v_0 to another state v_1, i.e. $v_1 = v_0 + t_1 + t_2 + \cdots + t_k$ and every intermediate step $v_0 + t_1 + t_2 + \cdots + t_i$ for $i \leqslant k$ is nonnegative.

As an elementary example of reachability in VAS, consider $d = 2$ and

$$T = \{(1, -1), (-1, 2)\}.$$

Then state $(1, 1)$ can legally reach state $(2, 1)$ by the sequence of transitions

$$((-1, 2), (1, -1), (1, -1)),$$

but $(0, 0)$ cannot reach legally $(1, 0)$, since applying any transition in $(0, 0)$ would violate the nonnegativity constraint.

Reachability in VAS thus consists of determining whether an algebraic, linear relation, (i.e. $v_1 - v_0$ belongs to the subgroup of $\mathbb{Z}^n$ generated by T) can be decomposed as a sequence of legal transitions – which is what we call a legal sequence in this work. The problem of deciding reachability has been an important question in the field of VAS, and it is known that reachability is decidable [17].

Now, let us turn our attention to abelian sandpiles, a model which is intrinsically linked to rotor-routing and is a special case of VAS. Consider a finite directed graph $G = (V, A)$. A state is an element of $\mathbb{N}^V$, and is called a particle (or chip) configuration. It is interpreted as the quantity of particles lying on every vertex of G. Transitions, here called *firings*, are defined for every vertex $v \in V$, and consist of adding one particle to every outneighbour of v, and removing those particles from v. This transition is called a *firing at* v, and such a firing is *legal* if the resulting configuration is nonnegative, which amounts to saying that before firing, v must have more particles than its number of outneighbours. See Fig. 1 for an example of firing in an abelian sandpile graph. Note that this model is a special case of VAS, with conservative transitions (i.e. the total number of particles remain constant). It is known that in the general case, legal reachability for abelian sandpiles is not likely to be in complexity class P [23].

The definition of legal firings in abelian sandpiles, has been extended by several authors [8, 23] to particle configurations in $\mathbb{Z}^V$, i.e. not necessarily nonnegative. In this more general context, we require for a legal firing that before firing at v, the configuration has enough particles on v to remain nonnegative at v after firing (i.e. more particles than its outdegree). During such a process, a vertex containing a negative number of particles, can only receive particles and not emit them. As was the case for VAS, the legal reachability issues in abelian sandpiles (even in this generalized context) involve determining whether an algebraic relation between configurations of particles can be decomposed into a legal sequence of firings, which amounts to checking non-negativity conditions.

In this paper, we show that standard rotor-routing in graphs, even in its generalized form with negative particles [9, 22], can be viewed as a special case of a more general model, namely Generalized Rotor Mechanism (GRM) multigraphs, where rotor-routing is a special case of conservative VAS, similar to abelian sandpiles. Legal reachability in this model also involves determining whether a linear relation between rotor and particle configurations can be decomposed into a series of legal transitions (routing steps). As in all previous cases, legality is determined by verifying non-negativity conditions.

(a) Initial configuration.

(b) Configuration obtained after the central vertex has fired.

Figure 1: Example of a transition in an abelian sandpile graph. The values represent the number of particles at each vertex.

1.3 Different notions of routing

Throughout this paper, we consider the movement of particles in a directed graph with vertex set V and arc set A (specifically, a multigraph). The term 'particle' can be replaced with pebble, chip, or counter, and has no physical significance. The positions of multiple particles are represented by a map $\sigma : V \to \mathbb{N}$, which counts the number of particles at each vertex. This is called a particle configuration. Notably, particles are indistinguishable and characterized solely by their positions.

In this paper, the term *routing* refers to the process of moving particles along the arcs of a graph according to specific rules that vary depending on the context. Mathematically, routing is defined as a rule that transforms one particle configuration into another one. Elementary routing operations involve moving a single particle along an arc: the particle count at the head of the arc is incremented by 1, while the count at the tail is decremented by 1. If these rules can be applied to all particle configurations without additional constraints, the routing is termed *linear*. Conversely, if certain conditions (typically non-negativity constraints) must be satisfied, the routing is referred to as *legal*.

In this work, we outline four distinct notions of routing, each with both a linear and a legal variant:

- **Standard rotor-routing in rotor multigraphs**: This refers to the classical concept of rotor-routing with standard cyclic rotor mechanisms. In this context, a configuration consists of both the particle configuration and the rotor configuration, where each non-sink vertex is associated with an outgoing arc.
- **Free routing in multigraphs**: This is the most basic form of routing, where particles move through a multigraph without the involvement of rotor mechanisms or rotor configurations. While free routing is not rotor-routing per se, it serves as a foundational tool for rotor-routing in GRM multigraphs.
- **Rotor-routing in cyclic GRM multigraphs**: This type of routing occurs in a subclass of GRM multigraphs, where the arc mechanism operates cyclically, which simulates the case of standard rotor-routing. However, it diverges from standard rotor-routing by allowing any formal sums of arcs instead of rotor configurations.
- **Rotor-routing in GRM multigraphs**: In this more general context, mechanisms of rotors are no more limited to a cyclic behavior.

1.4 Reachability problems

The reachability problem we address in this paper is whether, given an initial and a final configuration, there exists a routing from the initial to the final configuration. This problem is examined across all the routing types we previously defined, for both the linear and legal cases, extending the result given in [23] for standard rotor-routing. We also solve these problems in two scenarios: when the routing vector (i.e., the number of elementary operations at each vertex or arc) is specified, and when it is not. The results are summarized in the following table.

1.5 Organization of the paper

The paper is organized as follows. In Section 2, we introduce the standard rotor-routing framework and present significant results from prior research. Here, multiple particles

		Routing vector not specified	Routing vector specified
Standard rotor	Linear	[23], Proposition 3.3	(∗)
	Legal	[23], Theorem 3.4	Proposition 20
Free routing	Linear	Proposition 7	(∗)
	Legal	Proposition 11	Theorem 14
Cyclic GRM	Linear	Proposition 17	(∗)
	Legal	Theorem 30	Proposition 20
GRM	Linear	Proposition 17	(∗)
	Legal	Theorem 22	Theorem 18

Table 1: This table references the results that characterize the positive instances of the reachability problem and/or its computational complexity. All of these problems are in P, except for legal rotor-routing in GRM multigraphs, which is NP-complete. Cases marked with a star (∗) correspond to computing the result of a linear operator using the specified routing vector, which can be efficiently done via matrix multiplication.

are routed legally within a graph following standard rotor rules. This section is crucial as it establishes the notation used throughout the paper. While most of the standard rotor-routing results are not directly applied, they are generalized within the context of GRM multigraphs in Section 4.

In Section 3, we explore free routing and introduce the boundary operator, a key tool for Section 4. The focus of this section is on characterizing legal free routings within the framework of free routing.

Section 4 formally defines rotor-routing in multigraphs with generalized rotor mechanisms (GRM), extending the standard 'cyclic' mechanism. Two generalizations are presented: the method for updating arcs in each rotor and the notion of arc configurations that replace rotor configurations.

In Section 5, we characterize legal routings in GRM multigraphs when the routing vector is given as input. We then provide a simplified characterization for the cyclic case, which includes the standard rotor-routing model.

Finally, in Section 6, we address the reachability problem in cases where the routing vector is not specified a priori. We demonstrate that this problem is NP-complete in general, and present a polynomial-time algorithm for the cyclic case.

In a forthcoming paper, we will explore the algebraic properties of the GRM model, demonstrating how it enables a symmetric treatment of the rotor group and sandpile group, while also offering new insights into the Arrival problem.

2 Standard rotor-routing context and background

In this section, we recall the framework of directed graphs and rotor-routing, together with some known results that we use or generalize in the rest of the paper. We call the model of rotor-routing that is presented here the *standard rotor-routing* context, which is the model that can be found in most articles on the subject. For simplicity, we leave out the case of linear rotor-routing [22] and focus here solely on the case where positive particles are routed, according to a so-called *rotor configuration*.

2.1 Graphs

Multigraphs. A **directed multigraph** G is a tuple $G = (V, A, \text{head}, \text{tail})$ where V and A are respectively finite sets of *vertices* and *arcs*, and *head* and *tail* are maps from A to V defining incidence between arcs and vertices. An arc with tail x and head y is said to be from x to y. Note that multigraphs can have multiple arcs with the same head and tail, as well as loops.

For two sets $V_1, V_2 \subset V$, define $A(V_1, V_2)$ as the set of arcs with tail in V_1 and head in V_2. For a vertex $u \in V$, we denote by $A^+(u)$ the subset of arcs going out of u, i.e. $A^+(u) = A(u, V)$, as well as $A^-(u) = A(V, u)$.

Paths and connectedness. If $x, y \in V$, a **directed path from x to y** is a finite sequence of arcs $a_1, a_2 \cdots a_k$ such that $\text{head}(a_i) = \text{tail}(a_{i+1})$ for $1 \leqslant i \leqslant k-1$, and also $\text{tail}(a_1) = x$ and $\text{head}(a_k) = y$ (note that such a path is usually defined as a sequence of vertices, but both definitions are equivalent). This definition includes the empty sequence from x to x. The graph is said to be **strongly connected** if there is a directed path from any vertex to any other vertex. A **directed cycle** is a nonempty directed path from a vertex x to the same vertex x. This includes the case of a single arc (a loop) with the same tail and head.

The **strongly connected components** of G are the equivalence classes of the equivalence relation on vertices, where x and y are considered equivalent if there is a directed path from x to y and a directed path from y to x in G. A strongly connected component $C \subset V$ is a **leaf component** if $A(C, V \setminus C) = \emptyset$. In particular, if v is a **sink vertex** of G, i.e. $A^+(v) = \emptyset$, then $\{v\}$ is a leaf component of G.

We define an **undirected path from x to y** as a finite sequence of arcs $a_1, a_2 \cdots a_k$ such that there is a sequence $v_0, v_1, \cdots, v_k$ of vertices with $v_0 = x$, $v_k = y$ and such that for all $1 \leqslant i \leqslant k$ we have

- either $\text{tail}(a_i) = v_{i-1}$ and $\text{head}(a_i) = v_i$;
- or $\text{tail}(a_i) = v_i$ and $\text{head}(a_i) = v_{i-1}$.

In the first case we say that a_i is forward oriented in the path, and in the second case that it is backward oriented. Note that an undirected path corresponds to a path with the usual definition in the undirected graph where we replace every arc of G by an undirected edge.

An **undirected cycle** is an undirected path from a vertex x to the same vertex. In particular, we obtain an undirected cycle for every arc a by following it forward and then backward. Such undirected paths and cycles are called **arc-elementary** if they do not contain twice the same arc, in any orientation.

A **weakly connected component** of G is a maximal subset of vertices $V_1 \subset V$ such that there is an undirected path from any vertex of V_1 to any other vertex of V_1. The graph is said to be **weakly connected** if there is only one connected component which is V itself.

Trees, forests and rooted arborescences. An **undirected spanning forest** is a subset $F \subset A$ such that for every vertices x, y in the same (weakly) connected component of G, the graph $(V, F, \text{head}, \text{tail})$ contains exactly one arc-elementary undirected path from x to y. If G is connected, an undirected spanning forest is called an undirected spanning tree.

If $S \subset V$, an arborescence rooted in S is a set of arcs $T \subset A$ such for every $v \in V \setminus S$:

(i) there is exactly one arc $a \in T$ such that $\text{tail}(a) = v$;

(ii) there is a directed path in the subgraph $(V, T, \text{head}, \text{tail})$ from v to a vertex of S.

Note that such an arborescence exists if and only if for every vertex $v \in V \setminus S$, there is a directed path from v to some $s \in S$.

2.2 Free abelian groups

We refer to [16] for standard notions of abelian groups. We use additive notation. An abelian group $(H, +)$ is **free** if it admits a **basis**, i.e. a family $(h_i)_{i \in I}$ of elements such that each $h \in H$ can be written uniquely as a sum $h = \sum_{i \in I} c_i h_i$ with all $c_i \in \mathbb{Z}$. If there is a finite basis, then all basis have the same cardinal which is called the **rank** of H.

The universal property of free groups says that if $f : X \to R$ is a map from the elements of a basis X of H, to an abelian group R, then f extends in a unique way in an homomorphism from H to R.

If X is a finite set, the **free abelian group** on X is the set of formal sums with integer coefficients

$$c = \sum_{x \in X} c_x \cdot x$$

where $c_x \in \mathbb{Z}$ for every $x \in X$, with pointwise sum of coefficients. It can also be viewed as $\mathbb{Z}^X$, the set of maps from X to $\mathbb{Z}$ together with standard pointwise sum. We denote this group as C_X. Elements of X are identified with particular elements of C_X, and X can be viewed as a basis of C_X, which is called the canonical basis. If $x \in X$, we write c_x for the coordinates in the canonical basis of some $c \in C_X$.

We say that $c \in C_X$ is nonnegative, and write $c \geqslant 0$, if $c_x \geqslant 0$ for all $x \in X$. Likewise, we say that $c_1 \leqslant c_2$ if $c_2 - c_1 \geqslant 0$. Let C_X^+ the set of $c \in C_X$ such that $c \geqslant 0$. If $c \in C_X^+$ and $c_x > 0$, we say that ***x* is an element of *c*** and write $x \in c$. Indeed, $c \in C_X^+$ can be identified with a multiset on X.

If $G = (V, A, \text{head}, \text{tail})$ is a directed multigraph, we denote by C_V the free group on V; its elements are called **particle configurations**. Likewise, C_A denotes the free group on A and its elements are **arc configurations**. In the whole paper, instead of considering vectors or sets like $\mathbb{Z}^V$ and $\mathbb{Z}^A$, we consider formal sums C_V and C_A. This allows more concise notation: for instance, the element of C_V with coefficient 3 on v_1 and -5 on v_2 and 0 elsewhere will be simply denoted by $3v_1 - 5v_2$.

2.3 Laplacian homomorphism and matrix

Let $G = (V, A, \text{head}, \text{tail})$ be a directed multigraph. The Laplacian homomorphism is the homomorphism Δ from C_V to itself whose value on every vertex $v \in V$ (viewed as an element of C_V) is

$$\Delta(v) = \sum_{a \in A^+(v)} (\text{head}(a) - \text{tail}(a))$$

The Laplacian matrix L is the matrix of Δ in the canonical base of C_V. The following result is classic for undirected graphs and known as *Kirchoff's matrix-tree theorem*, but the directed version is a bit less common (see [19] for a proof).

Theorem 1. *Let $S \subset V$ and let L' be the reduced Laplacian matrix, obtained by removing from L the lines and columns corresponding to S. Then the determinant of L' is the number of arborescences rooted in S.*

Of particular interest to us is also the nullspace of Δ, which is given by the following result whose proof can be found in [1].

Theorem 2. *Let $k \geqslant 0$ be the number of leaf components of G. Denote by $S_1, S_2, \dots S_k$ these components. Then there exist $\sigma^1, \sigma^2, \dots, \sigma^k \in C_V$ such that*

(i) $(\sigma^1, \sigma^2, \dots, \sigma^k)$ is a basis of the null space of Δ;

(ii) for all $1 \leqslant i \leqslant k$, $\sigma^i_v > 0$ for all $v \in S_i$ and $\sigma^i_v = 0$ for all $v \in V \setminus S_i$.

This basis $(\sigma^1, \sigma^2, \dots, \sigma^k)$ is unique and its elements are called **primitive period vectors** *of G.*

In particular, if G is strongly connected, it admits a unique primitive period vector and the nullspace of Δ has rank 1. Moreover, if G is **eulerian**, i.e. $|A^+(v)| = |A^-(v)|$ for every $v \in V$, then the only primitive vector is $\sum_{v \in V} v$, i.e. its coordinates in the canonical basis are all 1. This is the case for the graph G_1 of Fig. 2.

2.4 Rotor structure

Let $G = (V, A, \text{head}, \text{tail})$ be a directed multigraph.

Rotor orders and rotor graphs. A **circular ordering** on a finite set X is a map $\theta : X \to X$ such that, for all $x \in X$, the sequence of iterates $x, \theta(x), \theta^2(x), \dots$ generates the whole set X.

If G is a multigraph, a **rotor** at $v \in V$ is a circular ordering on $A^+(v)$. A **rotor multigraph** $G = (V, A, \text{head}, \text{tail}, \theta)$ is such that:

- $(V, A, \mathrm{head}, \mathrm{tail})$ is a multigraph;
- for all vertices $v \in V$, the restriction of θ to $A^+(v)$ is a rotor at v.

If v is a **sink** vertex, i.e. $A^+(v) = \emptyset$, then the second condition is trivial. When the rotor multigraph G is fixed, we denote by S the set of its sinks. Depending on the context, S can be empty or not. A directed multigraph is **stopping** if for every vertex u, there is a directed path from u to a sink.

A **rotor configuration** in a rotor multigraph is a map ρ that associates to every $v \in V \setminus S$ an arc $\rho(v) \in A^+(v)$. We can identify rotor configurations with particular elements of C_A, namely $\sum_{v \in V \setminus S} \rho(v)$.

Two examples of rotor multigraphs are given on Fig. 2 and 3. The first one is strongly connected and sinkless, while the second is stopping.

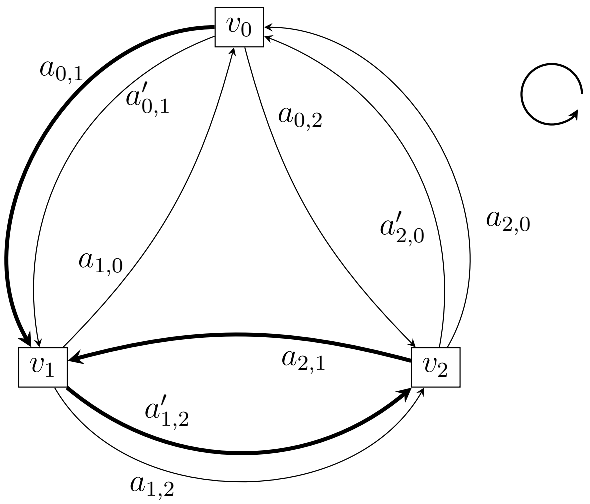


Figure 2: A rotor multigraph G_1 with no sinks, which is strongly connected. Every vertex has out-degree 3 and in-degree 3. As an example, we have $\mathrm{head}(a_{2,0}) = v_0$ and $\mathrm{tail}(a_{2,0}) = v_2$. The rotor order at every vertex is given by anticlockwise ordering; e.g. $\theta(a_{2,0}) = a'_{2,0}$, $\theta(a'_{2,0}) = a_{2,1}$ and $\theta(a_{2,1}) = a_{2,0}$. A rotor configuration ρ_1 with $\rho_1(v_0) = a_{0,1}$, $\rho_1(v_1) = a'_{1,2}$ and $\rho_1(v_2) = a_{2,1}$ is depicted in bold.

2.5 Standard rotor-routing

Let $G = (V, A, \mathrm{head}, \mathrm{tail}, \theta)$ be a rotor multigraph. Classically, rotor-routing is concerned with so-called *chip configurations*, namely nonnegative particle configurations with our current terminology. If $\sigma \in C_V$ and $\sigma_v > 0$, we interpret σ_v as the number of particles on vertex v. Routing these particles consists in moving them along an arc of a rotor configuration.

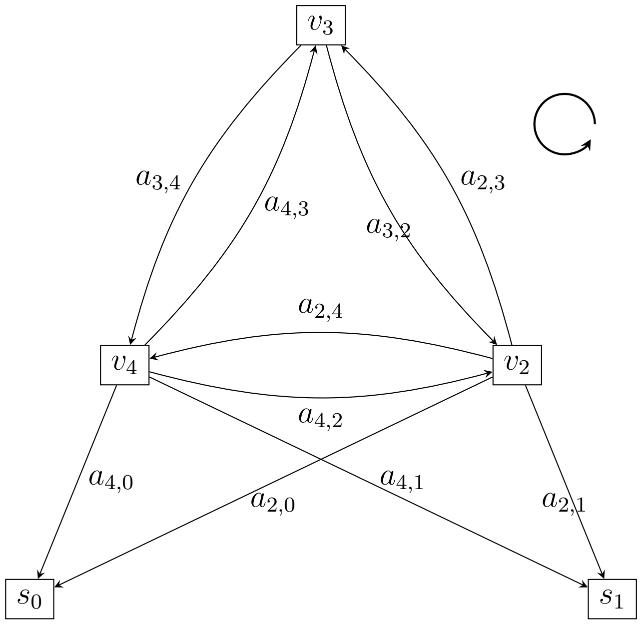


Figure 3: A stopping rotor multigraph G_2, with two sinks s_0 and s_1.

Rotor-routing operation and rotor walks. Consider a rotor configuration ρ and a nonnegative particle configuration $\sigma \in C_V^+$. A rotor-routing operation at $v \in V \setminus S$ is valid, only if $\sigma_v > 0$. In this case, the routing operation transforms (ρ, σ) into (ρ', σ') where:

- ρ' is equal to ρ except on v where $\rho'(v) = \theta(\rho(v))$;
- σ' is equal to σ except $\sigma'_v = \sigma_v - 1$ and $\sigma'_{v_1} = \sigma_{v_1} + 1$ where $v_1 = \text{head}(\rho(v))$.

We interpret this as a particle moving along the arc $\rho(v)$ from v to v_1, then the rotor configuration at v being updated to the next arc in the rotor ordering. Note that the resulting particle configuration σ' is also nonnegative. A **rotor walk** is a finite or infinite sequence of configurations $(\rho_0, \sigma_0), (\rho_1, \sigma_1), (\rho_2, \sigma_2), \ldots$ such that each new couple of configurations is obtained from the previous one by a routing operation. This sequence *starts* in (ρ_0, σ_0), and if finite *ends* in some (ρ_k, σ_k). Such a rotor walk is *maximal* if it is infinite or ends in a configuration (ρ_k, σ_k) where no valid routing operation can be applied, i.e. σ_k is 0 on v for all $v \in V \setminus S$. Fig. 4 shows an example of routing.

If there is a single particle (i.e. $\sigma = v$ for some vertex $v \in V$), then there is a unique maximal rotor walk starting in (ρ, σ). If there are several particles, there is a choice in the next particle to be routed. A first, fundamental result is the following [13, 22]:

Theorem 3. *1. if G is stopping, all maximal rotor walks are finite. Moreover, from a starting configuration (ρ, σ) where ρ is a rotor configuration and $\sigma \geqslant 0$, all maximal rotor walks end in the same configuration (ρ', σ'), and for every vertex $v \in V \setminus S$,* the number of times that a routing operation is performed at v does not depend on the choice of the maximal rotor walk ;

2. *if G is strongly connected, all maximal rotor walks are infinite. Moreover, in the case of a single particle configuration σ, let $p \in C_V$ be the unique primitive period vector of G. Then all maximal rotor walks that start in (ρ, σ) are ultimately periodic, and during a least period the number of times that a routing is performed at a vertex v is equal to $|A^+(v)| \cdot p(v)$.*

A more general result describing the asymptotic structure of maximal rotor walks, when the graph is neither stopping not strongly connected, can be found in [6] (Thm. 2.5.10).

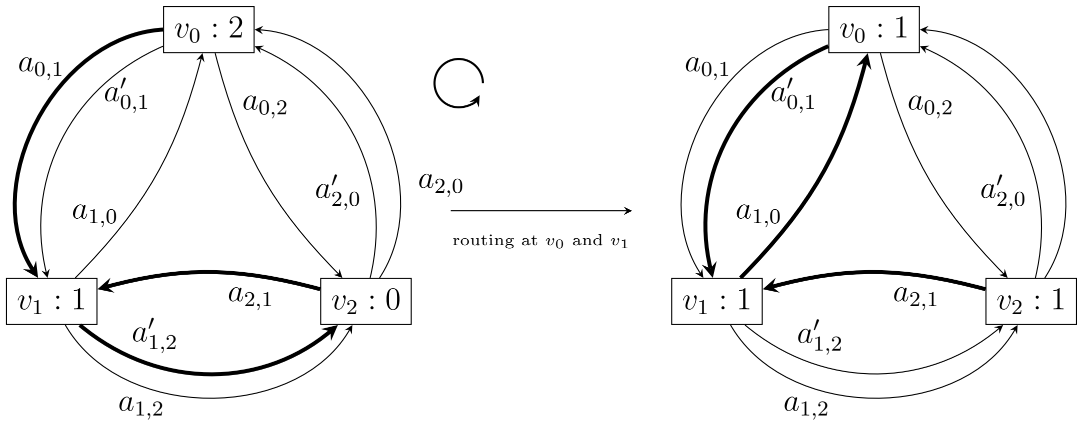


Figure 4: In the graph G_1 of Fig 2. On the left, a rotor configuration ρ_0 (arcs in bold) and a particle configuration σ_0 (numbers in vertices) are given. The rotor walk $(\rho_0, \sigma_0), (\rho_1, \sigma_1), (\rho_2, \sigma_2)$ is legal, and consists in routing a particle in v_0 and a particle in v_1. The resulting configurations (ρ_2, σ_2) are given on the right.

Arrival **Problem.** Suppose that G is a stopping rotor multigraph. The Arrival Problem, introduced in [5], consists in determining the final configuration σ' when applying a maximal rotor walk from (σ, ρ), where $\sigma \in C_V^+$ and ρ is a rotor configuration. The configuration σ' is then uniquely determined by Theorem 3. Equivalently, one can ask how many particles will settle on a given sink or use a decision version of the problem.

No polynomial-time algorithm is currently known for solving the Arrival problem, even in the case where σ consists in a single particle. In particular, routing such a particle may require an exponential number of steps. The most efficient algorithm known to date is subexponential, as described in [11].

Flows Let G be a stopping multigraph, ρ a rotor configuration and $\sigma \in C_V^+$, such that all maximal rotor walks that starts from (ρ, σ) end in (ρ', σ'), with $\sigma'_v = 0$ for all $v \notin S$, as stated in Theorem 3.1.

Define the **run** of (ρ, σ) as the map $f : A \to \mathbb{N}$ where $f(a)$ is the number of times that a particle travels along arc a during such a maximal walk (this number is also independent of the choice of the maximal walk, by the same result). Then the run f satisfies the following equations:

- for all $v \in V$, flow conservation at v:

$$\sum_{a \in A^-(v)} f(a) + \sigma_v = \sum_{a \in A^+(v)} f(a) + \sigma'_v \tag{1}$$

- for all $v \in V \setminus S$, rotor condition at v:

$$f(\rho(v)) \geqslant f(\theta(\rho(v))) \geqslant \cdots f(\theta^i(\rho(v))) \geqslant f(\theta^{i+1}(\rho(v))) \geqslant \cdots \geqslant f(\rho(v)) - 1 \tag{2}$$

Conversely, we call a **flow** for (ρ, σ, σ') a map $f : A \to \mathbb{N}$ satisfying all equations (1) and (2); hence the run of (ρ, σ) is a flow for (ρ, σ, σ'). Appendix A.1 gives detailed examples of run and flows in the graph G_2 of Fig. 3. It turns out that these equations are sufficient to characterize σ', by the following result, due to Dohrau et al [5].

Theorem 4. *Suppose that G is stopping, and that all maximal rotor walks that starts from (ρ, σ) end in (ρ', σ'). Let $\sigma_1 \in C_V$ with $\sigma_1(v) = 0$ for all $v \in V \setminus S$, then $\sigma_1 = \sigma'$ if and only if there exists a flow for (ρ, σ, σ_1).*

This result proves that Arrival belongs to the complexity class NP and to the class co-NP. Indeed, one can certify that the ending configuration is σ' by giving a flow for (ρ, σ, σ'), and one can also certify that the ending configuration is not σ' by giving a flow for (ρ, σ, σ'') where $\sigma'' \neq \sigma'$.

We can note that if f is the run for (ρ, σ), then it is easy to compute σ' with equations (1). It is also possible to compute ρ' with equations (2), since for all $v \in V \setminus S$:

- either $f(a)$ is constant for all $a \in A^+(v)$, which implies that $\rho'(v) = \rho(v)$;
- or there is $i > 0$ such that $f(\theta^i(\rho(v))) < f(\rho(v))$, and then the minimal such i gives $\rho'(v) = \theta^i(\rho(v))$.

If f is a flow for (ρ, σ, σ') but not the run for (ρ, σ), we can also build from f a corresponding rotor configuration ρ'' by equations (1), with the same relations. However, this configuration will not be ρ'. If we want to give a certificate for ρ', we need to characterize the run among flows by additional conditions. The following result is due to Gärtner et al [10] and implies in particular that Arrival belongs to UP and co-UP.

Theorem 5. *Let G be a stopping rotor multigraph. Let f be a flow for (ρ, σ, σ') and ρ' the rotor configuration built from f. Let $V_1 \subset V$ the set of active vertices for f, i.e. $v \in V$ such that $\sum_{a \in A^+(v)} f(a) > 0$, and let $A_1 \subset A$ the set of arcs $\theta^{-1}(\rho'(v))$ for $v \in V_1$. Then f is the run for (ρ, σ) if and only if $(V, A_1, head, tail)$ contains no directed cycles.*

The idea behind this result is that A_1 is the set of traces of the run, i.e. the arcs corresponding to the last time a particle is routed at each vertex, and following these arcs must lead to a sink.

This result was generalized by Tóthmérész [23] to the case of graphs that are not necessarily stopping and allow the presence of negative particles. The main results of this paper, notably Theorem 18, Proposition 20, Theorem 22 and Theorem 30, can be viewed as generalizations of Theorem 5 to various extensions of the standard rotor-routing exposed in the present part.

Recurrent configurations. Let G be a strongly connected rotor multigraph. A configuration (ρ, σ) is called *recurrent* if there exists a non-empty rotor walk that returns from (ρ, σ) to itself. Such configurations exist since, by Theorem 3.2, any maximal rotor walk on G is infinite. Characterization of recurrent configurations has been analyzed in [22]:

Theorem 6. *Let G be a strongly connected multigraph. Let $A_1 \subset A$ be the set of arcs $\theta^{-1}(\rho(v))$ for $v \in V \setminus S$. Then configuration (ρ, σ) is recurrent if and only if for every directed cycle C in A_1, there is an arc in C whose head v satisfies $\sigma_v > 0$.*

Turn and Move routing. We note that another definition of rotor-routing exists and is also widely used in the literature, which we refer to as *Turn and Move* routing (in contrast with the *Move and Turn* routing that we use). In this version, when a routing step at v for (ρ, σ) is processed, the rotor $\rho(v)$ is first updated to $\theta(\rho(v))$ and then a particle moves along the latter arc. This definition is also quite standard, and both definitions have their merits and flaws in terms of description of properties of routing. Ultimately, they are equivalent, as they are conjugate by the turn operator θ, as shown by the following commutative diagram, which provides a correspondence between two rotor walks, routed either with Move and Turn rotor (M&T) routing as used in this paper, or Turn and Move (T&M) as just described.

$$\begin{array}{ccccccc} (\rho_0,\sigma_0) & \xrightarrow{T\&M(v_0)} & (\rho_1,\sigma_1) & \xrightarrow{T\&M(v_1)} & \dots & \xrightarrow{T\&M(v_{k-1})} & (\rho_k,\sigma_k) \\ \big\downarrow{\scriptstyle \theta\times id} & & \big\downarrow{\scriptstyle \theta\times id} & & & & \big\downarrow{\scriptstyle \theta\times id} \\ (\rho_0^+,\sigma_0) & \xrightarrow{M\&T(v_0)} & (\rho_1^+,\sigma_1) & \xrightarrow{M\&T(v_1)} & \dots & \xrightarrow{M\&T(v_{k-1})} & (\rho_k^+,\sigma_k) \end{array}$$

3 Free routing in multigraphs

In this section, we study the free routing of particles in a graph without any constraints, notably without rotors. In this model, a particle can always move along any arc, hence the term 'free'. We introduce the boundary operator and highlight its key properties. This operator is central to our analysis, enabling us to count how many particles traverse each arc, thereby extending the concept of run/flow from Section 2.5. The terminology for the boundary operator, boundaries, and cycles is derived from standard simplicial homology of graphs (see, for instance, [12]).

Let $G = (V, A, \text{head}, \text{tail})$ be a multigraph. Recall from Subsection 2.2 that C_V is the free group on V and C_A the free group on A, and their elements are respectively called particle configurations and arc configurations. By the universal property of free groups, note that head and tail extend in a unique way as homomorphisms from C_A to C_V. Denote by $\mathcal{W}$ the set of weak connected components of G and by $C_{\mathcal{W}}$ the free group on $\mathcal{W}$. The connected component of a vertex v is denoted by $\deg(v)$, and we keep the same notation for the extension of deg as an homomorphism called degree from C_V to $C_{\mathcal{W}}$. If we denote by $\deg_w(\sigma)$ the coefficient of w in $\deg(\sigma)$, then $\deg_w(\sigma) = \sum_{v \in w} \sigma_v$ is the sum of all σ_v for all $v \in V$ in the component w. This **degree** measures the total number of particles in each component, and for all particle routing definitions used in this paper, it remains invariant. As particles move along arcs, they stay within the same connected components, ensuring the degree is preserved.

Routing involves moving particles along arcs in both forward and backward directions, requiring careful tracking of the arcs used. To this end, we introduce the boundary operator $\partial : C_A \to C_V$, defined as

$$\partial = \text{head} - \text{tail}.$$

For a particle configuration $\sigma \in C_V$ and an arc a, $\sigma + \partial(a)$ represents the configuration obtained by moving a particle from tail(a) to head(a), while $\sigma + \partial(-a)$ represents the reverse movement. This movement is algebraic, meaning σ does not need to be positive where particles are taken. More generally, $\sigma + \partial(r)$ for $r \in C_A$ consists in moving particles of σ along the arcs in r.

If $P = (a_1, a_2, \cdots, a_k)$ is an undirected path in G from x to y, we associate to this path the sum $\sum_{i=1}^{k} \alpha_i a_i$ with coefficient $\alpha_i = 1$ if a_i is forward-oriented in the path, and $\alpha_i = -1$ if it is backward-oriented. If $r \in C_A$ is built from an undirected path P in this way, we say that it represents P. In this case, we have

$$\partial(\sum_{i=1}^{k} \alpha_i a_i) = y - x.$$

The rest of this section is devoted to studying some properties of ∂.

3.1 Boundaries and sections

The image of ∂, i.e. the set of $\sigma \in C_V$ such that there is $r \in C_A$ with $\partial(r) = \sigma$, is denoted by B_V. It is a subgroup of C_V whose elements are called **boundaries**. The following is well known:

Proposition 7. *The image $B_V \subset C_V$ of ∂ is the nullspace of* deg.

Note that $\deg(\partial(a)) = 0_{\mathcal{W}}$ for every arc $a \in A$ since head(a) and tail(a) belong to the same weakly connected component of G. For the converse implication, we shall rely on the existence of **boundary sections**. A boundary section is an homomorphism s from C_V to C_A such that $\partial \circ s(\sigma) = \sigma$ for every $\sigma \in C_V$ satisfying $\deg(\sigma) = 0_{\mathcal{W}}$. Clearly, the existence of such a section proves that the nullspace of deg is in the image of ∂, hence settles Proposition 7.

Here is a way to construct a boundary section: let $b : \mathcal{W} \to V$ be a choice of a vertex $b(w) \in w$ for each connected component $w \in \mathcal{W}$ (i.e. $\deg \circ b = id_{\mathcal{W}}$). Such a b is called a **basepoint**. For every $v \in V$, let $s(v)$ be a representant of an arbitrary undirected path from $b(\deg(v))$ to v. Then s extends in an unique way as an homomorphism from C_V to C_A. For every $v \in V$, we have $\partial \circ s(v) = v - b(\deg(v))$, hence $\partial \circ s = id_{C_V} - b \circ \deg$, and in particular if $\deg(\sigma) = 0_{\mathcal{W}}$ then $\partial \circ s(\sigma) = \sigma$.

3.2 Cycles

The nullspace of ∂, i.e. the subgroup of arc configurations $r \in C_A$ such that $\partial(r) = 0_V$, is denoted as Z_A. Its elements are called **cycles** and Z_A the **cycle space**.

Proposition 8. *Elements of Z_A are characterized by Kirchoff's Law at every $v \in V$, i.e. for $r \in C_A$*

$$\sum_{a \in A^+(v)} r_a = \sum_{a \in A^-(v)} r_a$$

If r_a is interpreted as a flow along arc a, this equation means that the algebraic sum of flows leaving every vertex v is equal to the sum of entering flows.

Proof. We have for any $r \in C_A$,

$$\partial(r) = \sum_{a \in A} r_a(\text{head}(a) - \text{tail}(a))$$

$$= \sum_{v \in V} \left(\sum_{a \in A^-(v)} r_a - \sum_{a \in A^+(v)} r_a \right) \cdot v,$$

hence the result. □

3.3 Linear and legal free routings

In the multigraph G, a **linear free routing** with **routing vector** $r \in C_A$ is the operation of transforming $\sigma \in C_V$ into $\sigma + \partial(r)$. This amounts to moving simultaneously $|r_a|$ particles along each arc $a \in A$, forward if $r_a > 0$ and backward if $r_a < 0$. This routing can always be applied without further conditions. A **routing sequence** for r is a finite sequence $(a_i)_{0 \leqslant i \leqslant k-1}$ of arcs, such that $r = \sum_{0 \leqslant i \leqslant k-1} a_i$. It describes an order in which we can route along arcs in r one by one. A routing sequence from σ to σ' is a routing sequence such that $\sigma' = \sigma + \partial(r)$, where r is the routing vector of the sequence.

If $\sigma' = \sigma + \partial(r)$ with $\sigma, \sigma' \in C_V$ and $r \in C_A$, we write $\sigma \overset{r}{\underset{\partial}{\sim}} \sigma'$ and say that there is a linear routing from σ to σ' with routing vector r. Alternatively, we write $\sigma \overset{*}{\underset{\partial}{\sim}} \sigma'$ if there exists an $r \in C_A$ such that $\sigma \overset{r}{\underset{\partial}{\sim}} \sigma'$. This clearly defines an equivalence relation $\overset{*}{\underset{\partial}{\sim}}$ on C_V.

From the results of Sections 3.1 and 3.2, it follows that $\sigma \overset{*}{\underset{\partial}{\sim}} \sigma'$ if and only if $\sigma' - \sigma \in B_V$, which is equivalent to $\deg(\sigma) = \deg(\sigma')$. If $\sigma \overset{r}{\underset{\partial}{\sim}} \sigma'$, then the set of all possible routing vectors from σ to σ' is $r + Z_A$.

Among these linear routings, we want to identify those that correspond to legal routings, which we now define. Once again, this is not to be confused with rotor-routing as defined in Sec. 2.5: here there are neither rotor orderings nor rotor configurations, and we simply move particles along arcs of the graph. We say that the linear free routing from σ with routing vector $a \in A$ is **legal** for σ if $\sigma_{\mathrm{tail}(a)} \geqslant 1$. A routing sequence $(a_i)_{0 \leqslant i \leqslant k-1}$ from σ to σ' is legal if for every i, routing along a_i is legal for σ_i, where $\sigma_0 = \sigma$ and $\sigma_{i+1} = \sigma_i + \partial(a_i)$ for $0 \leqslant i \leqslant k-1$ (and consequently $\sigma_k = \sigma'$). If there exists a legal routing sequence from σ to σ', we write $\sigma \overset{*}{\underset{\partial}{\to}} \sigma'$, and use notation $\sigma \overset{r}{\underset{\partial}{\to}} \sigma'$ if we want to point out that there is a legal sequence with routing vector r. Note that by definition, $\sigma \overset{r}{\underset{\partial}{\to}} \sigma'$ implies $\sigma \overset{r}{\underset{\partial}{\sim}} \sigma'$ and $\sigma \overset{*}{\underset{\partial}{\to}} \sigma'$, both of which imply $\sigma \overset{*}{\underset{\partial}{\sim}} \sigma'$.

With these definitions, we observe that legal free routing is a special case of VAS as discussed in Sec. 1.2, which is conservative in the total number of particles. The legal reachability problem, in this context, involves decomposing a linear relation into a legal sequence.

3.4 Existence of a legal free routing

We now study how we can decide if $\sigma \overset{*}{\underset{\partial}{\to}} \sigma'$ and compute a legal sequence. A necessary condition for its existence is that there is a nonnegative routing vector from σ to σ', i.e. $r \in C_A^+$ with $\sigma \overset{r}{\underset{\partial}{\sim}} \sigma'$. In the case where $\sigma, \sigma' \geqslant 0$, the problem reduces to move particles in any order from $V^+ = \{v \in V : \sigma_v > \sigma'_v\}$ to $V^- = \{v \in V : \sigma_v < \sigma'_v\}$ and can be viewed as a flow or matching problem, by deciding which particle in V^+ will be routed to which vertex in V^-, counted with multiplicities. However, in presence of vertices with negative values, the situation is a little more complicated, as shows the example on Fig. 5.

A first problem is that vertices v with $\sigma'_v < 0$ can never be involved in a legal routing: they must remain *inactive*. These vertices act like an obstacle to legal routing operations and could as well be turned into sinks. However, vertices can begin with $\sigma_v < 0$, receive particles and become active during the sequence.

In the rest of this section, we suppose that G, σ, σ' are given and that we want to find a legal routing sequence from σ to σ'. If r is the routing vector of a legal routing sequence, then $r \in C_A^+$. We recall that if $r \in C_A^+$, we say that a is an element of r and write $a \in r$, if $r_a \geqslant 1$.

We solve the problem in two steps:

- first, we show that $\sigma \overset{*}{\underset{\partial}{\to}} \sigma'$ is equivalent to the existence of $r \in C_A^+$ with $\sigma \overset{r}{\underset{\partial}{\sim}} \sigma'$, satisfying additional properties (Proposition 9). We call such a vector a *legal routing vector* from σ to σ', since it asserts the existence of the legal routing sequence. We

(a) σ configuration (b) σ' configuration

Figure 5: A graph with σ given on the left and σ' on the right. There is a single linear routing vector r from σ to σ' which is $2a_1 + a_2$. However, none of the three sequences (a_1, a_1, a_2), (a_1, a_2, a_1), (a_2, a_1, a_1) are legal because the middle vertex remains nonpositive at all times.

also describe how to construct greedily a legal routing sequence from a legal routing vector (Lemma 10).

- Second, we show that deciding existence and computing a legal routing vector can be decided by standard weighted matching or network flow algorithms (Proposition 11) (hence in polynomial time).

3.4.1 From legal vectors to legal sequences

If $r, r' \in C_A^+$, we write $r \leqslant r'$ if for all $a \in A$, we have $r_a \leqslant r'_a$. A vertex v is **active** in r if there is an element $a \in r$ with $\operatorname{tail}(a) = v$. An active vertex is a vertex that will emit particles during a routing with routing vector r.

A **legal routing vector** from σ to σ' is $r \in C_A^+$ with $\sigma \underset{\partial}{\overset{r}{\sim}} \sigma'$, such that every active vertex v in r satisfies $\sigma'_v \geqslant 0$.

Proposition 9. *Let $\sigma, \sigma' \in C_V$. Then $\sigma \underset{\partial}{\overset{*}{\to}} \sigma'$ if only if there is a legal routing vector r from σ to σ'. In this case there is $r' \in C_A^+$ with $\sigma \underset{\partial}{\overset{r'}{\to}} \sigma'$ and $r' \leqslant r$.*

The proof is based on the following lemma, which explains how to construct greedily a legal routing sequence from a legal routing vector. We say that a routing sequence is ***r*-bounded**, if the routing vector r' of the sequence satisfies $r' \leqslant r$. Let $\mathcal{S}(\sigma, r)$ be the set of r-bounded routing sequences that are legal for σ. If $s_1, s_2 \in \mathcal{S}(\sigma, r)$, we say that $s_1 \leqslant s_2$ if s_1 is a prefix of s_2.

Lemma 10. *Suppose $r \in C_A^+$ is a legal routing vector from σ to σ'. Let $s \in \mathcal{S}(\sigma, r)$ with routing vector r_s such that $\sigma^s = \sigma + \partial(r_s) \neq \sigma'$. Then the set of arcs $a \in r - r_s$ whose routing is legal for σ^s is nonempty, and adding any such arc a to s extends s into $s \cdot a \in \mathcal{S}(\sigma, r)$.*

Proof. Since $\deg(\sigma^s) = \deg(\sigma) = \deg(\sigma')$, there is a vertex v such that $\sigma^s_v > \sigma'_v$. Since $\sigma^s \overset{r-r_s}{\underset{\partial}{\rightsquigarrow}} \sigma'$, there is $a \in A^+(v)$ with $a \in r - r_s$. Then $a \in r$, which is a legal routing vector, so $\sigma'_v \geqslant 0$ and $\sigma^s_v \geqslant 1$. Routing along a is legal for σ^s, hence $s \cdot a \in \mathcal{S}(\sigma, r)$. □

Proof of Proposition 9. Lemma 10 shows that the condition is sufficient. Conversely, suppose that there is a legal routing sequence $(a_i)_{0 \leqslant i \leqslant k-1}$ from σ to σ', and let $r \in C^+_A$ be its routing vector. If v is an active vertex in r, then there is a_i with $\text{tail}(a_i) = v$; consider the last such arc, denoted by a_ℓ. Let $\sigma^i = \sigma + \partial(\sum_{j=0}^{i-1} a_j)$. Since the routing is legal, we have $\sigma^\ell_v \geqslant 1$ and $\sigma^{\ell+1}_v \geqslant 0$. All arcs a_i for $i > \ell$ satisfy $\text{tail}(a_i) \neq v$, hence σ^i_v cannot decrease for $i = \ell + 1, \ldots, k-1$. Hence $\sigma'_v \geqslant 0$. We proved that r is legal. □

3.4.2 Existence of legal routing vectors

We now reduce the problem of deciding the existence, and computing a legal routing vector from σ to σ', to the following problem:

Bipartite Weighted Degree Constraint Problem (BWDC)	
INPUT:	a simple bipartite undirected graph (V_1, V_2, E) and a weight function $w : V_1 \cup V_2 \to \mathbb{N}$.
QUESTION:	is there a weight function on edges $f : E \to \mathbb{N}$ such that for every vertex $v \in V_1 \cup V_2$, the sum of all weights $f(e)$ of edges incident to v is equal to $w(v)$?

This problem is a classic network flow problem which can be solved by standard strongly polynomial algorithms like Edmond-Karp's or Dinic's algorithms [7, 4].

For the reduction, suppose that G, σ, σ' are given. Let $V^+ = \{v \in V : \sigma_v > \sigma'_v\}$ and $V^- = \{v \in V : \sigma_v < \sigma'_v\}$, and and consider the following instance (V^+, V^-, E_1, w) of BWDC where:

- the vertex set is $V^+ \cup V^-$
- there is an edge e between $v^+ \in V^+$ and $v^- \in V^-$ if and only if there is a directed path from v^+ to v^- in G', where G' is obtained from G by removing all arcs with tail in $\{v \in V : \sigma'_v < 0\}$
- the weight of $v^+ \in V^+$ is $w(v^+) = \sigma_v - \sigma'_v$ and the weight of $v^- \in V^-$ is $w(v^-) = \sigma'_v - \sigma_v$.

Proposition 11. *There is a legal routing vector from σ to σ' in G if and only if* BWDC *has a solution on instance (V^+, V^-, E_1, w). From any solution, a legal routing vector can be computed in polynomial time.*

Proof. First suppose that there is a legal routing vector r. Let $M = \sum_{v \in V^+}(\sigma_v - \sigma'_v)$. A consequence of the existence of a legal routing sequence (Lemma 10) is that we can find a legal routing vector $r' \leqslant r$ from σ to σ', that can be written as $r' = \sum_{i=1}^{M} r'_i$ and r'_i

represents a directed path from V^+ to V^-. We now define the weight $f(e)$ of an edge of E_1 from v^+ to v^- as the number of directed paths from v^+ to v^- among r'_i; then f is a solution to the BWDC instance.

Conversely, if f is a solution of the BWDC problem, we consider for every edge $e \in E$ from v^+ to v^-, an $r_e \in C_A$ representing a directed path from v^+ to v^- in G'. Then

$$r = \sum_{e \in E_1} f(e) \cdot r_e$$

is a routing vector from σ to σ', which is legal since all its elements belong to G'. □

Before ending this section, remark that the existence of a legal routing vector r just certifies that $\sigma \xrightarrow[\partial]{r'} \sigma'$ with $r' \leqslant r$, and does not imply that $\sigma \xrightarrow[\partial]{r} \sigma'$, as shows the example on Fig. 6. The situation is very similar to the notion of flows and runs for rotor-routing as explained in Sec. 2.5.

(a) σ configuration (b) σ' configuration

Figure 6: Graph G with σ on the left and σ' on the right. The vector $r = a_1 + a_2$ is a legal routing vector from σ to σ', but the only legal routing sequence is (a_1) with routing vector $r' = a_1 < r$.

3.5 Existence of legal routing with given routing vector.

We just saw that legal routing vectors characterize the existence of legal routing sequences, but that not every legal routing vector can be obtained as the routing vector of a legal sequence. We now state conditions that ensure $\sigma \xrightarrow[\partial]{r} \sigma'$ if $\sigma \underset{\partial}{\overset{r}{\sim}} \sigma'$ with $r \in C_A^+$. This is analogous to the characterization of runs for rotor-routing in 2.5 but for free routing in graphs.

In the proof of Lemma 10, we showed that if $\sigma \underset{\partial}{\overset{r}{\sim}} \sigma'$ with $r \in C_A^+$ and every active vertex v in r satisfies $\sigma'_v \geqslant 0$, we can build greedily a legal routing sequence from σ to σ' whose routing vector r' satisfies $r' \leqslant r$. However, if we construct this sequence without additional precaution, this process can end with $r \neq r'$ as shown in the example of Fig. 7.

To ensure that the whole routing vector r corresponds to a legal routing sequence, we need an additional condition on r, which is met on the example on Fig. 7. We need a little more terminology to describe the condition nicely.

(a) σ configuration (b) σ' configuration

Figure 7: A graph with σ given on the left and σ' on the right. We want to route a particle from bottom to top. Clearly $\sigma' = \sigma + \partial(a+b+c)$, and (a,b,c) is a legal routing sequence from σ to σ'. However, if we try to construct greedily a routing legal sequence and do not select b before c, we obtain the legal sequence (a,c) from σ to σ', which is legal but cannot be extended in a longer legal sequence: we forgot to use arc b before using c.

Consider σ, σ' and $r \in C_A^+$ such that $\sigma \overset{r}{\underset{\partial}{\sim}} \sigma'$. In this context, a vertex v is **transitory** if it is active in r and if $\sigma'_v = 0$. A transitory vertex is a vertex in which particles will not settle in the end. We denote by $\mathrm{Trans}(r, \sigma')$ the set of transitory vertices in the context of $\sigma \overset{r}{\underset{\partial}{\sim}} \sigma'$; note that σ is not relevant in the definition of transitory vertices.

Likewise, we say that a set of arcs $R \subset A$ is **guiding** for $\sigma \overset{r}{\underset{\partial}{\sim}} \sigma'$ if:

(i) R contains only elements of r;

(ii) for every transitory vertex $v \in \mathrm{Trans}(r, \sigma')$, G contains a directed path, with all arcs in R, from v to a vertex $v' \notin \mathrm{Trans}(r, \sigma')$ (i.e. v' is not active, or $\sigma'_v \geqslant 1$).

It is easy to see that a guiding set R which is minimal for inclusion contains exactly one arc $a \in r$ with $\mathrm{tail}(a) = v$ for every transitory vertex v, and contains no cycles (directed, or undirected). Conversely, such a set is guiding and we call it a **guiding tree**. For a guiding tree T and a transitory vertex $v \in \mathrm{Trans}(r, \sigma')$, we denote by $T(v)$ the arc a in T with $\mathrm{tail}(a) = v$.

Consider as an example Fig. 7 once again. Both $a+c$ and $a+b+c$ are routing vectors that admit a legal routing sequence. In each case, the two vertices whose value is 0 in σ' (middle and down) are transitory since they are active in the routing, and the unique guiding tree is $\{a, c\}$.

Let T be a guiding tree for $\sigma \overset{r}{\underset{\partial}{\sim}} \sigma'$ and consider an r-bounded legal routing sequence $s = (a_i)_{0 \leqslant i \leqslant k-1}$ for σ, with routing vector r'. We say that s is **guided by *T*** if for every transitory vertex $v \in \mathrm{Trans}(r, \sigma')$:

- if v is not an active vertex of $r - r'$, then the last arc with tail v in s is $T(v)$;
- if v is an active vertex of $r - r'$, then $T(v) \in r - r'$.

The main idea in this definition is that if a legal routing sequence has routing vector $r' = r$, then saying that it is guided by T is just saying that $T(v)$ is the last arc of the sequence with tail v, for every transitory vertex v; otherwise if the routing vector r' is such that $r' \lneq r$, the condition ensures $T(v)$ remains available in order to extend the sequence. Note that by construction, a prefix of legal routing sequence that is guided by T, is also guided by T. In the example of Fig. 7, a legal sequence guided by $T = \{a, c\}$ for $r = a + b + c$ will have to use arc b before arc c.

Lemma 12. *Let $(a_i)_{0 \leqslant i \leqslant k-1}$ be a legal routing sequence for $\sigma \xrightarrow[\partial]{r} \sigma'$. For any transitory vertex $v \in Trans(r, \sigma')$, let $T(v)$ be the last arc with tail v appearing in the sequence. Then $T := \{T(v)$ for $v \in Trans(r, \sigma')\}$ is a guiding tree for $\sigma \overset{r}{\underset{\partial}{\sim}} \sigma'$, and $(a_i)_i$ is guided by T.*

Proof. Let $(\sigma^i)_{0 \leqslant i \leqslant k}$ be the sequence of particle configurations in the legal sequence. Consider a transitory vertex v. By construction T contains exactly one arc of r with tail v; let i_v be the last index with $a_{i_v} = T(v)$ and let $v' = \text{head}(T(v))$. If v' is also transitory then $i_{v'} > i_v$: indeed, suppose that $i_{v'} \leqslant i_v$: then $\sigma^{i_v}_{v'} \geqslant 0$ and then $\sigma^{i_v+1}_{v'} \geqslant 1$. By definition of $i_{v'}$, all arcs a_i for $i > i_v + 1$ satisfy $\text{tail}(a_i) \neq v'$, so that $\sigma'_{v'} \geqslant \sigma^{i_v+1}_{v'} \geqslant 1$. This contradicts the fact that $\sigma'_{v'} = 0$. It follows that there can be no directed cycle in T, and no undirected cycle as well, so T is a guiding tree. $\square$

The next lemma proves that under conditions of Lemma 10, together with the existence of a guiding tree T, we can build greedily a legal routing sequence from σ to σ' that will have a routing vector exactly r. The difference with Lemma 10 is that we must make sure for transitory vertices that the last routing in which they are involved is the arc $T(v)$. Hence, the guiding tree is the "last exit" of a particle in a transitory vertex, and will guide the particle to its final position. This is very similar to the correspondence between Eulerian circuits and rooted arborescences in an Eulerian directed graph (see for instance [24]).

Let $\mathcal{S}(\sigma, r, T)$ be the set of r-bounded legal sequences for σ that are guided by T.

Lemma 13. *Let $r \in C^+_A$. Suppose that $\sigma \overset{r}{\underset{\partial}{\sim}} \sigma'$, that every active vertex v of r satisfies $\sigma'_v \geqslant 0$, and that there exists a guiding tree T for the routing. Let $s \in \mathcal{S}(\sigma, r, T)$ with routing vector r_s such that $r_s \neq r$.*

Then there is an active vertex v of $r - r_s$ with $\sigma^s(v) \geqslant 1$. Define an arc $a \in A^+(v)$ by:

- *if $v \notin Trans(r, \sigma')$, let a be any arc $a \in r - r_s$ with tail v;*
- *if $v \in Trans(r, \sigma')$ and there are at least two distinct arcs in $r - r_s$ with tail v, let a be any such arc that is not $T(v)$;*
- *if $v \in Trans(r, \sigma')$ and $T(v)$ is the only arc in $r - r_s$ with tail v, let $a = T(v)$.*

Then $s \cdot a \in \mathcal{S}(\sigma, r, T)$.

Proof. We prove first that there is an active vertex v of $r - r_s$, such that $\sigma^s_v \geqslant 1$. Suppose that there is no such vertex and consider the set V_s of active vertices of $r - r_s$. By hypothesis, $\sigma'_v \geqslant 0$ for every $v \in V_s$. On the other hand, since no arcs in $r - r_s$ emit particles from $V \setminus V_s$ into V_s, it follows that $\sum_{v \in V_s} \sigma'_v \leqslant \sum_{v \in V_s} \sigma^s_v$. Putting these two facts together, we obtain

$$0 \leqslant \sum_{v \in V_s} \sigma'_v \leqslant \sum_{v \in V_s} \sigma^s_v = 0.$$

It follows that $\sigma'_v = 0$ for every $v \in V_s$ so that V_s contains only transitory vertices, and for arc $a \in r - r_s$, head$(a) \in V_s$. By hypothesis s is guided by T, so $T(v) \in r - r_s$ for every $v \in V_s$. Since arcs $a \in r - r_s$ are such that head$(a) \in V_s$ and tail$(a) \in V_s$, it implies that T contains a directed cycle, which is a contradiction.

By choosing arc a as stated in the Lemma, we ensure that $s \cdot a \in \mathcal{S}(\sigma, r, t)$. □

Here is the main result of this section:

Theorem 14. *If* $\sigma \underset{\partial}{\overset{r}{\sim}} \sigma'$ *with* $r \in C^+_A$, *then* $\sigma \underset{\partial}{\overset{r}{\to}} \sigma'$ *if and only if*

(i) every active vertex v *satisfies* $\sigma'_v \geqslant 0$, *and*

(ii) the set of arcs which are elements of r *is guiding.*

Proof. If the set of elements of r is guiding, then $\sigma \underset{\partial}{\overset{r}{\sim}} \sigma'$ admits a guiding tree so by Lemma 13, the conditions are sufficient.

Conversely if there is a legal routing sequence with routing vector r, then Condition (i) is necessary by Proposition 9 and Condition (ii) is necessary by Lemma 12. □

Figure 8 illustrates this result.

3.6 Traces of a legal free routing

A **trace** of a legal routing $\sigma \underset{\partial}{\overset{r}{\to}} \sigma'$ is an arc a for which there is a legal routing sequence with routing vector r that ends with arc a. Note that except for loops, an arc $a \in r$ is a trace of $\sigma \underset{\partial}{\overset{r}{\to}} \sigma'$ if and only if we have $\sigma \underset{\partial}{\overset{r-a}{\to}} (\sigma' - \partial(a))$ as well. The following result is easy:

Lemma 15. *Suppose that* G *has weak connected components* $V_1, V_2, \cdots, V_k$. *Define* A_i *as the set of arcs with head and tail in* V_i *and let* $G_i = (V_i, A_i, head, tail)$. *Let* $\sigma \in C_V$ *and* $r \in C^+_A$, *decomposed as* $\sigma = \sum_{i=1}^k \sigma_i$ *and* $r = \sum_{i=1}^k r_i$ *according to weak connected components, and let* $\sigma'_i = \sigma_i + \partial(r_i)$. *Then for every* i, *an arc* $a \in A_i$ *is a trace of* $\sigma \underset{\partial}{\overset{r}{\to}} \sigma'$ *in* G *if and only if it is a trace of* $\sigma_i \underset{\partial}{\overset{r_i}{\to}} \sigma'_i$ *in* G_i.

(a) σ configuration (b) σ' configuration

Figure 8: A graph with σ given on the left and σ' on the right. Consider $\sigma \overset{a}{\underset{\partial}{\sim}} \sigma'$: the unique transitory vertex is the bottom one since the top one is not active. Then $\{a\}$ is a guiding tree for that routing and there is a legal routing sequence with routing vector a. Consider now $\sigma \overset{a+b}{\underset{\partial}{\sim}} \sigma'$. Both vertices are transitory, but there is no guiding tree for $a + b$. One can easily check that there is no legal routing sequence from σ with routing vector $a + b$.

3.7 Summary of complexity results for reachability questions in free routing

We can sum up the algorithmic complexities underlying the reachability problems studied in this section by the following result. Polynomial here means polynomial in sizes of G, σ, σ' or r (configurations being encoded in binary, i.e. strongly polynomial).

Proposition 16. *Let G be a multigraph. Let $\sigma, \sigma' \in C_V$ and $r \in C_A^+$*

(i) the complexity of deciding if $\sigma \overset{*}{\underset{\partial}{\sim}} \sigma'$ is polynomial – by checking if $\deg(\sigma) = \deg(\sigma')$*;*

(ii) the complexity of deciding if $\sigma \overset{*}{\underset{\partial}{\to}} \sigma'$ is polynomial – by reduction to a network flow algorithm (Proposition 11*);*

(iii) the complexity of deciding if $\sigma \overset{r}{\underset{\partial}{\to}} \sigma'$ is polynomial – check conditions of Theorem 14.

Note however that in (*ii*) and (*iii*), legal routing sequences can be obtained easily from appropriate routing vectors, but they can have exponential length.

4 Rotor-routing in Generalized Rotor Mechanisms multigraphs

Now that we have defined the necessary tools in previous sections, we can define the generalization of standard rotor-routing. The generalization is twofold: first, the graphs where we study rotor-routing will be no more limited to standard rotor multigraphs with a cyclic rotor on every vertex. Every vertex will be allowed to have a more complex mechanism for updating arcs, leading to the model of *generalized rotor mechanism* (GRM) multigraphs. Second, the linear routing takes place in the space $C_A \times C_V$ of arcs and vertices at the same time, whereas in the previous section we considered just vertices. We will interpret this as free routing, in the sense of Section 3, simultaneously in two graphs.

The motivation is twofold: first, it puts the accent on the symmetry between transformations in configurations of vertices (particles) and arcs (rotors) during rotor-routing. Second, some important reachability results, developed in Section 5, are valid in the generalized case.

4.1 Motivation

Consider once again the rotor multigraph G_1 of Fig. 2 and suppose that there is a single particle $\sigma = v_2$ on the vertex v_2, together with a rotor configuration ρ such that $\rho(v_2) = a_{2,1}$, as depicted on the right side of Fig. 9. If we proceed in a routing step at v_2, in the sense of standard rotor-routing, the particle will be transferred to v_1, so that σ becomes $\sigma + \partial(\rho(v_2)) = \sigma + \partial(a_{2,1})$, where ∂ is the boundary operator in G as defined in Sec. 3. At the same time, ρ is updated so that $\rho(v_2)$ becomes $\rho'(v_2) = a_{2,0}$. If we see ρ as a formal sum of arcs $\sum_{v \in V} \rho(v)$, we can write this transformation as $\rho + a_{2,0} - a_{2,1}$. We interpret this as the routing of an 'arc particle' in the graph G^A depicted in the left side of Fig. 9.

In order to distinguish the graphs used to route "vertex particles" and "arc particles", we denote by G^V the graph G_1 and ∂^V its boundary operator. On the other hand, let us denote by ∂^A the boundary operator of the graph G^A. The vertices of G^A correspond to arcs of G^V, and the arcs of G^A join every arc a of G^V to its successor $\theta(a)$ in the rotor ordering; hence G^A is a collection of disjoint directed cycles. To avoid confusion, while we keep using the standard terminology of vertices and arcs for G^V, we respectively use the terms 'arcs' and 'faces' for formal vertices and arcs of G^A.

With these notation, the rotor-routing step described just above can be written as

$$(\rho', \sigma') = (\rho, \sigma) + (\partial^A(f_{2,1}), \partial^V(a_{2,1})).$$

Note that $a_{2,1} = \mathrm{tail}^A(f_{2,1})$, where tail^A is the tail operator in G^A.

We can then see a rotor step as two simultaneous free routings, in the sense of Sec. 3, in graphs G^A and G^V. In what follows, we define a generalized rotor mechanism multigraph, where the graph G^A is no more limited to be a collection of disjoint directed cycles.

4.2 Definition of generalized rotor mechanism multigraphs

We generalize the standard rotor mechanism, where G^A consists in the union of directed cycles simply by allowing any multigraph on every $A^+(v)$ instead of a directed cycle.

More precisely, let $G^V = (V, A, \mathrm{head}^V, \mathrm{tail}^V)$ be a multigraph (for particles). Choose for every $v \notin S$, where S is the set of sinks in G^V, a multigraph $G^A(v) = (A^+(v), F(v), \mathrm{head}^A, \mathrm{tail}^A)$ where $F(v)$ is any abstract finite set, and $\mathrm{head}^A, \mathrm{tail}^A$ are defined from $F(v)$ to $A^+(v)$ without restriction.

Let G^A be the union of the graphs $G^A(v)$, and let F be the union of all $F(v)$ for $v \in V \setminus S$, so that $G^A = (A, F, \mathrm{head}^A, \mathrm{tail}^A)$ is a multigraph. The elements of F are called **faces**.

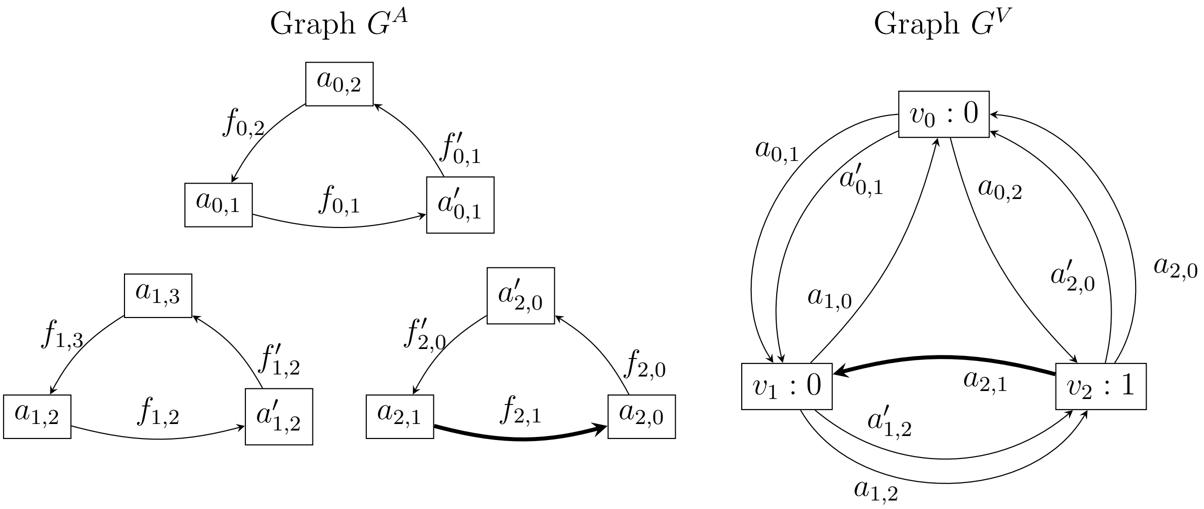


Figure 9: The rotor multigraph G_1 of Fig. 2 defined as a generalized rotor mechanism (G^A, G^V). Among others, the face in bold $f_{2,1}$ and the arc in bold $a_{2,1}$ are coupled.

We call a couple of multigraphs (G^A, G^V) built like this, a **Generalized Rotor Mechanism** (GRM from now on). We denote as $\deg^V$ and $\deg^A$ the degree mappings in graphs G^V and G^A as defined in Sec. 3. An example of GRM multigraph is given on Fig. 10.

4.3 Definition of linear rotor-routing in GRM multigraphs

Let (G^A, G^V) be a GRM multigraph as defined above with $G^A = (A, F, \text{head}^A, \text{tail}^A)$ and $G^V = (V, A, \text{head}^V, \text{tail}^V)$. Formal sums of faces are denoted C_F. Define

$$\mathcal{L} : C_F \to C_A \times C_V$$

by

$$\mathcal{L} = \partial^A \times (\partial^V \circ \text{tail}^A).$$

Let $(r, \sigma) \in C_A \times C_V$. We define **the linear rotor-routing along** $\phi \in C_F$ as the operation that transforms (r, σ) into $(r, \sigma) + \mathcal{L}(\phi)$, and ϕ is called the **routing vector** of the routing operation. Note that if ϕ is a single face, with $a' = \text{head}^A(\phi)$ and $a = \text{tail}^A(\phi)$, this transformation adds $a' - a$ to r, and adds $v' - v$ to σ, where $v' = \text{head}^V(a)$ and $v = \text{tail}^V(a)$.

We say that (r, σ) and (r', σ') are equivalent (modulo linear routing), denoted by

$$(r, \sigma) \overset{*}{\underset{\mathcal{L}}{\sim}} (r', \sigma'),$$

if $(r' - r, \sigma' - \sigma) \in Im(\mathcal{L})$, i.e. if there is a routing vector that transforms (r, σ) into (r', σ'). If we want to specify that the routing vector is ϕ, we write $(r, \sigma) \overset{\phi}{\underset{\mathcal{L}}{\sim}} (r', \sigma')$. Note that the linear routing operation is completely algebraic and can be computed by forming the matrix of $\mathcal{L}$. An example of this matrix is given in Appendix A.2.

4.4 Definition of legal rotor-routing in GRM multigraphs

An elementary linear routing is the routing along a face $f \in C_F$. Such a linear routing is said **legal** for $(r, \sigma) \in C_A \times C_V$ if $r_a \geqslant 1$ and $\sigma_v \geqslant 1$, where $a = \text{tail}^A(f)$ and $v = \text{tail}^V(a)$. The interpretation is that there is a real 'vertex particle' on v in σ and a real 'arc particle' on a in r, as it is the case in standard rotor-routing: adding $\mathcal{L}(f)$ consists in moving respectively these particles along a, from v to $\text{head}^V(a)$, and along f from a to $\text{head}^A(f)$.

Consider the special case of $(r, \sigma) = (\rho, \sigma)$ where ρ is a rotor configuration, viewed as a sum of arcs. In this case, for every non-sink vertex v, there is exactly one arc $a \in A^+(v)$ such that $\rho_a > 0$, namely $\rho(v)$. If $\sigma_v > 0$ and we want to apply a GRM legal routing in order to move the particle on v, the process is similar to the routing in a standard rotor multigraph, with the the following differences:

- when we want to move the particle from v along the arc $\rho(v) \in A^+(v)$, we can have more than one possible update on $\rho(v)$. Choosing a face $f \in F(v)$ with $\text{tail}^A(f) = \rho(v)$ determines the evolution $+\text{head}(f) - \rho(v)$ that will happen on ρ, simultaneously with the movement of the particle;
- it can also happen that there is no face f with $\text{tail}^A(f) = \rho(v)$. In this case, no legal routing is available: the particle cannot legally move anymore, because $\rho(v)$ acts like a sink in the set of arcs.

Some basic generalized rotor mechanisms could be for instance, a rotor multigraph where every arc a can be updated to the next arc $\theta(a)$, or to the previous arc $\theta^{-1}(a)$, for every routing along a, or a rotor multigraph with an arc-sink for every vertex, e.g. $G^A(v)$ is a directed path for every non sink vertex instead of a cycle.

It can be noted that linear routing preserves degrees. In other words, if $(r, \sigma) \underset{\mathcal{L}}{\overset{*}{\sim}} (r', \sigma')$, then $\deg^A(r) = \deg^A(r')$ and $\deg^V(\sigma) = \deg^V(\sigma')$. This means that the total algebraic number of particles should be the same in every connected component of G^V, and that the sum of arcs should be the same in every mechanism $G^A(v)$ (or in each of the weak connected components of the mechanism, if not connected).

A **routing sequence** for a routing vector $\phi \in C_F$ is a finite sequence of faces $f_0, f_1, \ldots, f_k$ such that $\phi = \sum_i f_i$. This sequence is legal for (r, σ) if, when applying in order elementary routing steps along $f_0, f_1, \ldots, f_k$, every step is legal.

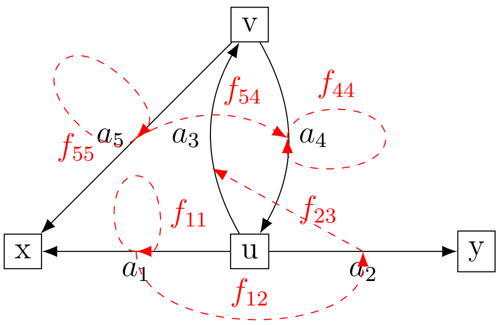
For legal routings, we use notation akin to the case of free routing, namely

$$(r, \sigma) \underset{\mathcal{L}}{\overset{*}{\to}} (r', \sigma')$$

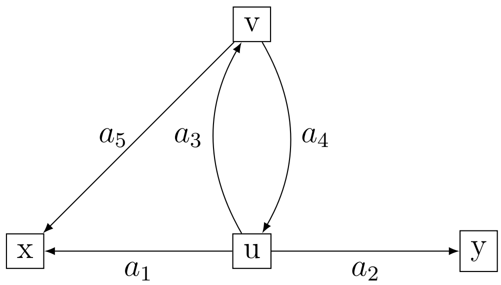
if there is a legal routing sequence from (r, σ) to (r', σ'), and $(r, \sigma) \underset{\mathcal{L}}{\overset{\phi}{\to}} (r', \sigma')$ if there is a legal sequence with routing vector exactly ϕ.

4.5 Cyclic GRM multigraphs

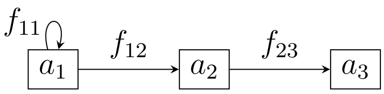
If $G = (V, A, \text{head}, \text{tail}, \theta)$ is a rotor multigraph, we can associate to G a GRM multigraph (G^A, G^V) defined as follows:



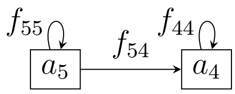
(a) Generalized rotor mechanism in one picture: the dashed red arcs represent the possible evolution of arcs when routed.



(b) Graph $G^V = (V, A, \text{head}^V, \text{tail}^V)$, with $V = \{x, y, u, v\}$ and $A = \{a_1, a_2, a_3, a_4, a_5\}$



(c) Graph $G^A(u)$ on $A^+(u)$ with face set $F(u) = \{f_{11}, f_{12}, f_{23}\}$



(d) Graph $G^A(v)$ on $A^+(v)$ with face set $F(v) = \{f_{44}, f_{54}, f_{55}\}$

Figure 10: a GRM multigraph G. In Fig. (a): representation in one picture of the GRM, with arcs in full black and faces in dashed red. The other figures corresponds to graphs G^V (Fig. (b)) and G^A respectively, the last graph being split into graph $G^A(u)$ (Fig. (c)) and $G^A(v)$ (Fig. (d)).

- $G^V = (V, A, \text{head}, \text{tail})$
- $G^A = (A, F, \text{head}^A, \text{tail}^A)$ where $F = \{(a, \theta(a)) : a \in A\}$, and $\text{head}^A((a, \theta(a)) = \theta(a)$, $\text{tail}^A((a, \theta(a)) = a$ for all $a \in A$.

A GRM multigraph built from a rotor multigraph like this is called a **cyclic GRM multigraph**, since $G^A(v)$ is a directed cycle on the 'vertex' set $A^+(v)$ for every $v \in V \setminus S$ (notation S will always refer to the sinks of G^V). We say that a cyclic GRM multigraph is stopping, or strongly connected, or any other property of rotor multigraphs, if the corresponding rotor multigraph satisfies the property. Since the weakly connected components of G^A are precisely the sets $A^+(v)$ for all $v \in V \setminus S$, we denote by $\deg_v^A(r)$ the coefficient of $A^+(v)$ in $\deg^A(r)$, which simply means $\deg_v^A(r) = \sum_{a \in A^+(v)} r_a$.

In this context, a rotor configuration is an element $\rho \in C_A$ such that for any $v \in V \setminus S$, there is unique $a \in A^+(v)$ such that $\rho_a = 1$, and $\rho_a = 0$ for the others. It is equivalent to saying that $\rho \in C_A^+$ and $\deg_v^A(\rho) = 1$. In a cyclic GRM multigraph, if ρ is a rotor configuration and $\sigma_v > 0$, the linear rotor-routing along the face $(\rho(v), \theta(\rho(v))$ matches the standard definition of rotor-routing. Hence, *the notion of legal routing in cyclic GRM multigraphs completely emulates standard rotor-routing as defined in Section 2.5.* In the following, we speak of cyclic GRM multigraphs instead of rotor multigraphs to avoid clashes in definitions of rotor-routing, as the settings of cyclic GRM multigraphs and rotor multigraphs are essentially the same, but the definitions of rotor-routing differ.

4.6 Computing routing vectors

Given a routing vector $\phi \in C_F$, the matrix of $\mathcal{L}$ allows to compute the linear rotor-routing along ϕ in polynomial time (see Appendix A.2 for a detailed example). Conversely, if $r, r' \in C_A$ and $\sigma, \sigma' \in C_V$ are given as input, one may seek to decide if a routing vector ϕ such that $(r', \sigma') = (r, \sigma) + \mathcal{L}(\phi)$ exists, and if so, compute such a routing vector. As mentioned before, a necessary condition is that $\deg^V(\sigma) = \deg^V(\sigma')$ and $\deg^A(r) = \deg^A(r')$.

In what follows, the term 'polynomial' means polynomial in the size of G, together with $r, r' \in C_A$ and $\sigma, \sigma' \in C_V$ (encoded in binary).

Proposition 17. *Let $r, r' \in C_A$ and $\sigma, \sigma' \in C_V$. There is a polynomial time algorithm that decides whether $(r, \sigma) \overset{*}{\underset{\mathcal{L}}{\sim}} (r', \sigma')$, and if so, returns a routing vector $\phi \in C_F$ such that $(r', \sigma') = (r, \sigma) + \mathcal{L}(\phi)$ and ϕ has polynomial size.*

Proof. This amounts to computing an integral solution of a system of linear diophantine equations, if it exists. This can be achieved by using the Smith decomposition of the matrix of $\mathcal{L}$ (see Appendix B for details on how to use the decomposition in order to solve the system). It remains to compute effectively the decomposition of the matrix, which can be done in polynomial time [14]. $\square$

5 Legal rotor-routings in GRM multigraphs with specified routing vector

We explore the concept of legal rotor-routing within GRM multigraphs, as defined in the preceding section. The main difficulties associated with standard rotor-routing comes from its intricate combinatorial properties. To address these, we characterize the subset of linear routings that correspond to legal movements, following an approach similar to that in Section 3.3, but specifically adapted for linear rotor-routing in GRM multigraphs. Our objective is to identify the source of complexity in legal rotor-routing when framed within the linear setting.

5.1 General case

We consider a generalized routing mechanism as defined in Sec. 4.2. First note that if $(f_i)_{0\leqslant i\leqslant k-1}$ is a legal routing sequence for (r,σ) in a generalized routing mechanism multigraph (G^A,G^V) then $(f_i)_{0\leqslant i\leqslant k-1}$ and $(\mathrm{tail}^A(f_i))_{0\leqslant i\leqslant k-1}$ are legal routing sequences in G^A and G^V respectively for r and σ in the sense of Sec. 3.3 (free legal routing).

The opposite direction is more tricky. The coupling condition complicates things and we cannot consider any two legal routing sequences in G^A and G^V (resp. for ϕ and $\mathrm{tail}^A(\phi)$) to build a legal routing sequence for ϕ in the GRM multigraph as illustrated in Figure 11. By Theorem 14, we may observe that for a vertex v, if the last routing in $G^A(v)$ is along a face $f\in F(v)$, with $a=\mathrm{tail}^A(f)$, then we must at the same time ensure that a belongs to some guiding tree for the routing of particles in G^V, and f is a trace of the routing in $G^A(v)$.

(a) Names of vertices, arcs and faces. (b) Configurations (r,σ) and (r',σ').

Figure 11: Consider the routing vector $\phi=f_{12}+f_{22}$ and let $\alpha=\mathrm{tail}^A(\phi)=a_1+a_2$. Let $r=a_1$, $r'=a_2$, $\sigma=u$ and $\sigma'=v$ as depicted on Fig.(b). Then we have $r\xrightarrow[\partial^A]{\phi} r'$ where the only free legal routing sequence with routing vector ϕ is (f_{12},f_{22}), and $\sigma\xrightarrow[\partial^V]{\alpha}\sigma'$ where the only free legal routing sequence with routing vector α is (a_2,a_1). However, there is no legal rotor-routing from (r,σ) to (r',σ') with routing vector ϕ.

The following result presents these conditions, providing both necessary and sufficient criteria for the existence of a legal routing with a specified routing vector. The proof of sufficiency not only establishes these criteria but also outlines an algorithm for computing a legal routing sequence, building on the previous discussion.

Theorem 18. *Consider a GRM multigraph* (G^A, G^V)*. Let* $r, r' \in C_A$ *and* $\sigma, \sigma' \in C_V$*. Let* $\phi \in C_F^+$ *with* $\alpha = tail^A(\phi)$*. Then* $(r, \sigma) \xrightarrow[\mathcal{L}]{\phi} (r', \sigma')$ *if and only if:*

(i) $r \xrightarrow[\partial^A]{\phi} r'$ *in the sense of free routing;*

(ii) $\sigma \xrightarrow[\partial^V]{\alpha} \sigma'$ *in the sense of free routing;*

(iii) if T *is the set of traces of* $r \xrightarrow[\partial^A]{\phi} r'$*, then* $\{tail^A(f) : f \in T\}$ *is guiding for* $\sigma \underset{\partial^V}{\overset{\alpha}{\sim}} \sigma'$*.*

Note that (i) and (ii) are characterized by Theorem 14.

Proof. Let us make the convention that a sequence of elements of F (resp. of A) is said legal for $r \in C_A$, (resp. for $\sigma \in C_V$) if it is legal in the sense of free routing in G^A (resp. in G^V), as per Section 3.3; and that a sequence of elements of F is said legal for (r, σ), if it is legal in the sense of linear rotor-routing in (G^A, G^V) as described in Section 4.3.

Let us introduce some notation for the proof. Since G^A is the union of the weak connected components $G^A(v)$ for $v \in V \setminus S$, we decompose as in Lemma 15 any routing vector ϕ in G^A as $\phi = \sum_{v \in V \setminus S} \phi_{|v}$, where $\phi_{|v} \in C_{F(v)}$. Define also the restrictions of $r, r' \in C_A$ to $A^+(v)$ by $r_{|v}$, $r'_{|v}$ so that $r = \sum_v r_{|v}$ and $r' = \sum_v r'_{|v}$. If $r \underset{\partial^A}{\overset{\phi}{\sim}} r'$ then for every v we have $r_{|v} \underset{\partial^A}{\overset{\phi_{|v}}{\sim}} r'_{|v}$. For a routing sequence $s = (f_i)_i$, we can as well decompose the sequence as subsequences $s_{|v}$ for each non sink vertex v, where faces appearing in $s_{|v}$ belong to F_v; then clearly s is legal for r if and only if all $s_{|v}$ are legal for $r_{|v}$.

We now show the necessity of conditions (i)-(iii): suppose the existence of the legal routing sequence $(f_i)_{0 \leqslant i \leqslant k-1}$ for (r, σ). We denote $a_i = \mathrm{tail}^A(f_i)$ for $0 \leqslant i \leqslant k-1$. By definition of legality in GRM multigraphs, $(f_i)_{0 \leqslant i \leqslant k-1}$ and $(a_i)_{0 \leqslant i \leqslant k-1}$ are respectively legal routing sequences for $r \xrightarrow[\partial^A]{\phi} r'$ and $\sigma \xrightarrow[\partial^V]{\alpha} \sigma'$, which proves (i) and (ii). Then, by Lemma 12, for any transitory vertex v of $\sigma \underset{\partial^V}{\overset{\alpha}{\sim}} \sigma'$, if $t(v)$ denotes the last arc with tail v appearing in the sequence $(a_i)_{0 \leqslant i \leqslant k-1}$, then the set of all $t(v)$ forms a guiding tree for this routing. For such a transitory vertex v, let $i(v)$ be the last index of $t(v)$ in that sequence. Then $f_{i(v)}$ is the last element of $F(v)$ appearing in $(f_i)_i$, so it is by Lemma 15 a trace of $r_{|v} \xrightarrow[\partial^A]{\phi_{|v}} r'_{|v}$, hence a trace of $r \xrightarrow[\partial^A]{\phi} r'$, so that (iii) is true.

Conversely, suppose that conditions (i)-(iii) are satisfied. By (iii), we can suppose for every $v \in \mathrm{Trans}(\alpha, \sigma')$, that there is $f(v) \in F(v)$, so that if $t(v) = \mathrm{tail}^A(f(v))$:

- $f(v)$ is a trace of $r \xrightarrow[\partial^A]{\phi} r'$, and

- $\{t(v)\}_v$ is a guiding tree for $\sigma \overset{\alpha}{\underset{\partial^V}{\rightsquigarrow}} \sigma'$.

By Lemma 15, for every vertex $v \in V \setminus S$, we can consider a (possibly empty) legal routing sequence $s(v) = (f_0^v, \ldots, f_{k_v}^v)$ for $r_{|v} \overset{\phi_{|v}}{\underset{\partial^A}{\to}} r'_{|v}$, such that, if $v \in \mathrm{Trans}(\alpha, \sigma')$ then $f_{k_v}^v = f(v)$. This sequence describes an ordering of all routing steps that must be done in $F(v)$ for every v. Our strategy is to construct recursively a legal routing sequence $s = (f_i)_{0 \leqslant i \leqslant k}$ for $(r, \sigma) \overset{\phi}{\underset{\mathcal{L}}{\to}} (r', \sigma')$ that coincides with $s(v)$ for every active vertex v, i.e. $s_{|v} = s(v)$. This will ensure that the routing of particles will be guided by t and that the sequence can be extended.

Suppose that $\ell \geqslant 0$ and let $s^\ell = (f_i)_{0 \leqslant i \leqslant \ell-1}$ is a routing sequence with routing vector ϕ^ℓ, such that:

- s^ℓ is legal for $(r, \sigma) \overset{\phi^\ell}{\underset{\mathcal{L}}{\to}} (r^\ell, \sigma^\ell)$;
- $\phi^\ell \leqslant \phi$;
- $s^\ell_{|v}$ is a prefix of $s(v)$ for every vertex v.

Suppose that $\phi^\ell \neq \phi$, and let $\alpha^\ell = \mathrm{tail}^A(\phi^\ell)$. By construction of $s(v)$, the sequence $(a_i)_{0 \leqslant i \leqslant \ell-1}$ with $a_i = \mathrm{tail}^A(f_i)$ is guided by t and is α-bounded. Then, by Lemma 13, there is an active vertex v of $\alpha - \alpha^\ell$, such that $\sigma^\ell_v \geqslant 1$. Then we choose the face $f \in F(v)$ with $f \in \phi - \phi^\ell$ such that $s^\ell_{|v}$ appended with f is still a prefix of $s(v)$. Then, the extended sequence $s^{\ell+1} = (f_0, \ldots, f_{\ell-1}, f)$ satisfies

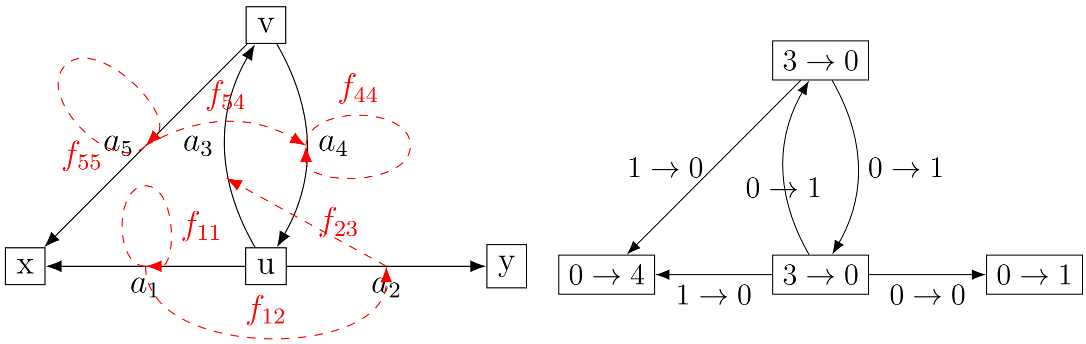
- $s^{\ell+1}$ is legal for (r, σ) since by construction $\sigma^\ell_v \geqslant 1$ and $s^{\ell+1}_{|v}$ is a prefix of a legal sequence for $r_{|v} \overset{\phi_{|v}}{\underset{\partial^A}{\to}} r'_{|v}$;
- $\phi^{\ell+1} \leqslant \phi$;
- $s^{\ell+1}_{|v}$ is a prefix of $s(v)$ for every vertex v by construction also.

Finally, starting from the empty sequence, we obtain recursively a sequence s^ℓ with $\phi^\ell = \phi$ satisfying the three properties above, hence a legal sequence for $(r, \sigma) \overset{\phi}{\underset{\mathcal{L}}{\to}} (r', \sigma')$.

□

This theorem is illustrated in an example of the generalized rotor mechanism in Figure 12. An example where conditions (i) and (ii) of the theorem are satisfied, but not condition (iii) is shown in Figure 11.

We note that the characterization given by Theorem 18 can be checked in polynomial time. Indeed, (i) and (ii) can be checked in polynomial time as already mentioned in Proposition 16, and checking (iii) amounts to computing traces in a free routing, then checking accessibility by standard graph algorithms. A simple way to check if a face $f \in \phi$ is a trace of the free routing $r \overset{\phi}{\underset{\partial^A}{\to}} r'$, is to check if $r''_a \geqslant 1$ and $r \overset{\phi - f}{\underset{\partial^A}{\to}} r''$, where $r'' = r' - \partial_A(f)$ and $a = \mathrm{tail}^A(f)$, which is also polynomial by Proposition 16.



(a) Names of vertices, arcs and faces.

(b) Configurations (r, σ) and (r', σ') on arcs and vertices.

Figure 12: GRM multigraph of Fig. 10. Configurations σ and σ' are, respectively, the values at the tail and the head of the arrows in the squares at the vertices of the graph in Fig. (b), configurations r and r' are the values associated to the arcs. Consider $\phi = f_{11} + f_{12} + f_{23} + f_{44} + f_{54} + f_{55}$ and $\alpha = \text{tail}^A(\phi) = 2a_1 + a_2 + a_4 + 2a_5$, then $(r', \sigma') = (r, \sigma) + \mathcal{L}(\phi)$. Clearly $\sigma \underset{\partial^V}{\overset{\alpha}{\sim}} \sigma'$ and $r \underset{\partial^A}{\overset{\phi}{\sim}} r'$ admit legal routing sequences with routing vectors α and ϕ respectively. The set of traces of $r \underset{\partial^V}{\overset{\phi}{\to}} r'$ is $\{f_{23}, f_{44}\}$. Transitory vertices for the routing $\sigma \underset{\partial^V}{\overset{\alpha}{\sim}} \sigma'$ are u and v and $\text{tail}^A(f_{23}) + \text{tail}^A(f_{44}) = a_2 + a_4$ which is guiding for this routing. Hence there is a legal routing sequence with routing vector ϕ.

5.2 Cyclic case

In this part, we characterize the routing vectors that admit a legal routing sequence in a cyclic GRM multigraph, obtaining a condition which is easier to check than the general condition of Theorem 18. We recall that a cyclic GRM multigraph is equivalent to a standard rotor-routing graph, with we allow routing with any arc configuration (see Section 4.5).

We begin with a useful lemma that characterizes traces in the legal free routing of arcs in the multigraph G^A.

Lemma 19. *Suppose that (G^A, G^V) is a cyclic GRM multigraph. Consider a legal free routing $r \xrightarrow[\partial^A]{\phi} r'$ in G^A. Let $f_0 \in \phi$, and let $a_0 = tail^A(f_0)$ so that $head^A(f_0) = \theta(a_0)$. Then f_0 is a trace of this routing if and only if $\theta(a_0) \notin Trans(\phi, r')$, i.e. if $r'_{\theta(a_0)} \geqslant 1$, or there is no $f \in F$ with $\phi_f \geqslant 1$ and $tail^A(f) = \theta(a_0)$.*

Proof. If f_0 is a trace, then $r \xrightarrow[\partial^A]{\phi - f_0} r''$, where $r'' = r' - \partial^A(f_0)$. If $a_0 = \theta(a_0)$, which means that $\text{tail}^V(a_0)$ has outdegree 1 in G^V, then $r_{a_0} = r'_{a_0} = r''_{a_0}$; since a_0 is active in ϕ, we must have from the beginning $r_{a_0} \geqslant 1$ so that $r'_{a_0} \geqslant 1$, therefore $\theta(a_0) = a_0 \notin \text{Trans}(\phi, r')$. If $a_0 \neq \theta(a_0)$, and $\theta(a_0)$ is active in ϕ, then it is also active in $\phi - f_0$, then $r''_{\theta(a_0)} \geqslant 0$, so $r'_{\theta(a_0)} \geqslant 1$, and $\theta(a_0) \notin \text{Trans}(\phi, r')$.

Conversely, suppose that $\theta(a_0) \notin \text{Trans}(\phi, r')$. Since $f_0 \in \phi$, a_0 is active in ϕ. If $a_0 \neq \theta(a_0)$, we have $r'_{a_0} \geqslant 0$ so $r''_{a_0} \geqslant 1$. If $a_0 = \theta(a_0)$, then $r_{a_0} = r'_{a_0} = r''_{a_0} \geqslant 1$. In all cases, $r''_{a_0} \geqslant 1$. So if we prove that $r \overset{\phi - f_0}{\underset{\partial^A}{\sim}} r''$ admits a legal routing sequence, it will be legal to add f_0 at the end, proving that f_0 is a trace. To see the existence of a legal sequence with routing vector $\phi - f_0$, we check the conditions of Theorem 14.

First, for any active arc a in ϕ, we have $r'_a \geqslant 0$, since $r \xrightarrow[\partial^A]{\phi} r'$. The active arcs in $\phi - f_0$ are those in ϕ, possibly excluding arc a_0. Meanwhile, the only possible arc a such that $r''_a < r'_a$ is $\theta(a_0)$, and $r'_{\theta(a_0)} - 1 \leqslant r''_{\theta(a_0)} \leqslant r'_{\theta(a_0)}$. If $\theta(a_0)$ is active in $\phi - f_0$, then it is also active in ϕ. However, we assumed that $\theta(a_0) \notin \text{Trans}(\phi, r')$, which implies $r'_{\theta(a_0)} \geqslant 1$, and therefore $r''_{\theta(a_0)} \geqslant 0$. Hence, every arc a active in $\phi - f_0$ satisfies $r''_a \geqslant 0$.

Second, we check the existence of a guiding tree for $r \overset{\phi - f_0}{\underset{\partial^A}{\sim}} r''$. Let $v_0 = \text{tail}^V(a_0)$. Since $r \overset{\phi - f_0}{\underset{\partial^A}{\sim}} r''$ and $r \overset{\phi}{\underset{\partial^A}{\sim}} r'$ only differ in $G^A(v_0)$, the only remaining task is to verify the existence of a guiding tree for $r_{|v_0} \overset{\phi_{|v_0} - f_0}{\underset{\partial^A}{\sim}} r''_{|v_0}$ (we recall that notation $r_{|v_0}$ means the part or r with arcs in $A^+(v_0)$).

If $F(v_0) \cap \text{Trans}(\phi - f_0, r'') = \emptyset$, then there is nothing to check, the empty set is a guiding tree for $r_{|v_0} \overset{\phi_{|v_0} - f_0}{\underset{\partial^A}{\sim}} r''_{|v_0}$. Otherwise let $a \in \text{Trans}(\phi_{|v_0} - f_0, r''_{|v_0})$ and suppose that the set of faces which are elements of $\phi_{|v_0} - f_0$ is not guiding. Since $G^A(v_0)$ is a directed cycle, we also have $\theta(a) \in \text{Trans}(\phi_{|v_0} - f_0, r''_{|v_0})$, $\theta^2(a) \in \text{Trans}(\phi_{|v_0} - f_0, r''_{|v_0})$ and so on, so that $A^+(v_0) \subset \mathrm{Trans}(\phi_{|v_0} - f_0, r''_{|v_0})$, and in particular $a_0 \in \mathrm{Trans}(\phi_{|v_0} - f_0, r''_{|v_0})$. This contradicts the fact that $r''_{a_0} \geqslant 1$, which was proved above.

We checked the two conditions of Theorem 14. $\square$

Consider now a free linear routing $r \overset{\phi}{\underset{\partial^A}{\sim}} r'$. If $r \overset{\phi}{\underset{\partial^A}{\to}} r'$, by Lemma 19, the traces of this routing are exactly the faces $f \in \phi$ such that $\mathrm{tail}^A(f)$ belongs to the following set T_A:

$$T_A = \{a \in \alpha : r'_{\theta(a)} > 0\} \cup \{a \in \alpha : \theta(a) \notin \alpha\},$$

where $\alpha = \mathrm{tail}^A(\phi)$. Note that traces are entirely characterized by the values ϕ and r'.

Based on this characterization, we can derive the following result as an adaptation of Theorem 18 for GRM multigraphs. Unlike Theorem 18, which relied on applying the characterization from Theorem 14 twice as a subroutine, this result is self-contained.

Proposition 20. *Suppose that (G^A, G^V) is a cyclic GRM multigraph and $(r, \sigma) \overset{\phi}{\underset{\mathcal{L}}{\sim}} (r', \sigma')$ for some $\phi \in C_F^+$. Let $\alpha = tail^A(\phi)$ and*

$$T_A = \{a \in \alpha : r'_{\theta(a)} > 0\} \cup \{a \in \alpha : \theta(a) \notin \alpha\}.$$

Then $(r, \sigma) \overset{\phi}{\underset{\mathcal{L}}{\to}} (r', \sigma')$ if and only if

(i) $\sigma'_v \geqslant 0$ for every vertex v active in α;

(ii) $r'_a \geqslant 0$ for every arc a active in ϕ;

(iii) $T_A \cap A^+(v) \neq \emptyset$ for every vertex v active in α;

(iv) T_A is guiding for $\sigma \overset{\alpha}{\underset{\partial^V}{\sim}} \sigma'$.

Proof. By Theorem 18, there is a legal routing sequence from (r, σ) to (r', σ') with routing vector ϕ if and only if

(1) $\sigma \overset{\alpha}{\underset{\partial^V}{\to}} \sigma'$

(2) $r \overset{\phi}{\underset{\partial^A}{\to}} r'$

(3) $\{\mathrm{tail}^A(f) : f \in T_F\}$ is guiding for $\sigma \overset{\alpha}{\underset{\partial^V}{\to}} \sigma'$ where T_F is the set of traces of $r \overset{\phi}{\underset{\partial^A}{\to}} r'$.

First suppose that (i), (ii), (iii) and (iv) are satisfied. By Theorem 14 conditions (i) and (iv) of the proposition imply condition (1), since the set of arcs T_A is contained in α.

Let v be active in α. By (iii), there is $a \in A^+(v) \cap T_A$. By definition of T_A we have $\theta(a) \notin \mathrm{Trans}(\phi, r')$. We show the existence of a guiding tree for $r_{|v} \overset{\phi_{|v}}{\underset{\partial^A}{\sim}} r'_{|v}$.

If $F(v) \cap \mathrm{Trans}(\phi, r') = \emptyset$, then there is nothing to check and the empty set is suitable. Otherwise let $\hat{a} \in \mathrm{Trans}(\phi_{|v}, r'_{|v})$ and suppose that the set of faces that are elements of

$\phi_{|v}$ is not guiding. Since $G^A(v)$ is a directed cycle, we also have $\theta(\hat{a}) \in \mathrm{Trans}(\phi_{|v}, r'_{|v})$, $\theta^2(\hat{a}) \in \mathrm{Trans}(\phi_{|v}, r'_{|v})$ and so on, so that $A^+(v) \subset \mathrm{Trans}(\phi_{|v}, r'_{|v})$. This is a contradiction with $\theta(a)$ not being transitory. Hence the support of ϕ is guiding. This together with (ii) implies by Theorem 14 that $r \xrightarrow[\partial^A]{\phi} r'$. Hence condition (2) is satisfied. Finally, by Lemma 19, we have $T_A = \mathrm{tail}^A(T_F)$, hence (iv) and (3) are equivalent.

Conversely, suppose that condition (1), (2) and (3) are satisfied. We already showed that (3) implies (iv).

By (2), for every active vertex v in α there must be an arc $a \in A^+(v)$ which is not transitory for $r \xrightarrow[\partial^A]{\phi} r'$. This corresponds to condition (iii).

Finally, condition (1) implies (i), and condition (2) implies (ii), hence the equivalence between the two sets of conditions. □

To conclude this section, we show how to apply these results to the case of standard rotor-routing. We suppose that $\sigma, \sigma' \in C_V^+$, and that r, r' are rotor configurations. In the context of cyclic GRM multigraphs, this means that $r, r' \in C_A^+$ and that for all $v \notin S$, we have $\deg_v(r) = \deg_v(r') = 1$. We denote by ρ and ρ' two such arc configurations.

Corollary 21. *Suppose that (G^A, G^V) is a cyclic GRM multigraph and that $(\rho', \sigma') \overset{\phi}{\underset{\mathcal{L}}{\sim}} (\rho, \sigma)$, with $\phi \in C_F^+$, with ρ, ρ' rotor configurations and $\sigma, \sigma' \in C_V^+$. Let $\alpha = tail^A(\phi) \in C_A^+$. Then $(\rho, \sigma) \xrightarrow[\mathcal{L}]{\phi} (\rho', \sigma')$ if and only if*

$$\{a \in \alpha : \theta(a) \in \rho'\}$$

is guiding for $\sigma \overset{\alpha}{\underset{\partial^V}{\sim}} \sigma'$.

Proof. Recall that in this context T_A as defined in Proposition 20 can be expressed as

$$T_A = \{a \in \alpha : \theta(a) \in \rho'\} \cup \{a \in \alpha : \theta(a) \notin \alpha\}.$$

We prove that $T_A = \{a \in \alpha : \theta(a) \in \rho'\}$. Let $a \in \alpha$ such that $\theta(a) \notin \alpha$. Then $\rho'_{\theta(a)} = \rho_{\theta(a)} + \alpha_a$ which implies that $\rho'_{\theta(a)} \geqslant 1$ and then $\rho'_{\theta(a)} = 1$. Thus, $\{a \in \alpha : \theta(a) \notin \alpha\} \subset \{a \in \alpha : \theta(a) \in \rho'\}$.

Suppose that $\{a \in \alpha : \theta(a) \in \rho'\}$ is guiding for $\sigma \overset{\alpha}{\underset{\partial^V}{\sim}} \sigma'$. We check conditions (i) to (iv) of Proposition 20.

Since $\sigma \in C_V^+$ and $\rho' \in C_A^+$, conditions (i) and (ii) are satisfied. Moreover, for every vertex $v \notin S$ there is $a \in A^+(v)$ such that $\rho'_a > 0$. Hence $\{a \in \alpha : \rho'_{\theta(a)} > 0\} \cap A^+(v)$ is nonempty for every active vertex v in α and condition (iii) is satisfied. Condition (iv) follows from $T_A = \{a \in \alpha : \theta(a) \in \rho'\}$.

Conversely, condition (iv) implies that $\{a \in \alpha : \theta(a) \in \rho'\}$ is guiding for $\sigma \overset{\alpha}{\underset{\partial^V}{\sim}} \sigma'$. □

In the special case where G^V is stopping, and we aim to simulate maximal rotor walks (see Sec. 2.5) in GRM multigraphs, the previous result leads to the characterization of runs among flows, as stated in Theorem 5.

6 Legal reachability in GRM multigraphs

In the previous section, we developed a method to determine whether a legal routing exists in GRM multigraphs for a given routing vector. We also derived a simplified characterization for cyclic GRM multigraphs, both of which are verifiable in polynomial time. In this section, we extend our analysis to address the same problem, without the assumption of a specified routing vector. More formally, we define the LEGAL REACHABILITY IN GRM MULTIGRAPH as follows:

LEGAL REACHABILITY IN GRM MULTIGRAPH (LR-GRMM)	
INPUT:	(G^A, G^V) a GRM multigraph, $r, r' \in C_A$ and $\sigma, \sigma' \in C_V$.
QUESTION:	does $(r, \sigma) \xrightarrow[\mathcal{L}]{*} (r', \sigma')$?

The challenge lies in the fact that multiple routing vectors ϕ may satisfy the equation $(r', \sigma') = (r, \sigma) + \mathcal{L}(\phi)$. Among these, some routing vectors may admit a legal routing sequence, while others may not.

We begin by examining the general case and prove:

Theorem 22. LEGAL REACHABILITY IN GRM MULTIGRAPH *problem is NP-complete.*

We shall then focus on the case of cyclic GRM multigraphs, where a polynomial-time algorithm is feasible, as a specific routing vector can be tested in this context.

6.1 Proof of Theorem 22

A routing vector $\phi \in C_F^+$ with $(r, \sigma) \xrightarrow[\mathcal{L}]{\phi} (r', \sigma')$ is a certificate that $((G^A, G^V), r, r', \sigma, \sigma')$ is a positive instance of LR-GRMM, which can be checked in polynomial time by Theorem 18. Hence LR-GRMM is in NP. The rest of the section is dedicated to the proof of NP-hardness by polynomial reduction from a boolean satisfiability problem.

We consider a special version of the 3-SAT problem, where boolean formulas are given in conjunctive normal form with clauses of 3 literals, where each variable appears exactly twice unnegated and exactly twice negated, as in the formula

$$(x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee \overline{x_2} \vee \overline{x_3}) \wedge (\overline{x_1} \vee x_2 \vee \overline{x_3}) \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3).$$

This restriction of 3-SAT, which we will call 3-SAT-(2,2), has been proved NP-complete [3]. An instance of 3-SAT-(2,2) is then a boolean formula made of n variables $x_1, x_2, \ldots, x_n$ and m clauses of three literals $c_1, c_2, \ldots, c_m$, such that for all i, x_i appears in exactly two clauses c_j, and $\overline{x_i}$ appears in exactly two clauses c_j as well.

To an instance of 3-SAT-(2,2), we will associate an instance $((G^A, G^V), r, r', \sigma, \sigma')$ of LR-GRMM, where (G^A, G^V) is a GRM multigraph. We shall first define a core multigraph, consisting only of arcs and vertices. Then, the clause gadgets will add some faces to the core multigraph, and then the variable gadgets will add more vertices and arcs, as well as faces. Finally, we will define r, r', σ and σ'.

Core multigraph. We begin by defining the core multigraph:

- the set of vertices is $V = V_{\text{var}} \cup V_{\text{clause}} \cup S$ with
  - the set of *variable vertices*, $V_{\text{var}} = \{x_i\}_{i \in \{1,\dots,n\}}$
  - the set of *clause vertices*, $V_{\text{clause}} = \{c_j\}_{j \in \{1,\dots,m\}}$
  - the set of sinks $S = \{s, s_{\text{sat}}\}$
- the set of arcs is $A = A^+_{\text{var}} \cup A^-_{\text{var}} \cup A_{\text{clause,s}} \cup A_{\text{clause,sat}}$ with
  - A^+_{var} contains an arc from every variable vertex to each of the two clause vertices where it appears unnegated; more precisely $A^+_{\text{var}} = \{a^+_{i,j}\}_{i \in \{1,\dots,n\}, j \in \{1,2\}}$, where $\text{tail}(a^+_{i,j}) = x_i$ and $\{\text{head}(a^+_{i,1}), \text{head}(a^+_{i,2})\}$ are the two clause vertices c such that x_i appears unnegated in c;
  - A^-_{var} contains an arc from every variable vertex to each of the two clause vertices where it appears negated; $A^-_{\text{var}} = \{a^-_{i,j}\}_{i \in \{1,\dots,n\}, j \in \{1,2\}}$, where $\text{tail}(a^-_{i,j}) = x_i$ and $\{\text{head}(a^-_{i,1}), \text{head}(a^-_{i,2})\}$ are the two clause vertices c such that x_i appears negated in c;
  - $A_{\text{clause,s}}$ contains an arc from every clause vertex to s; more precisely $A_{\text{clause,s}} = \{a_{j,1}\}_{j \in \{1,\dots,m\}}$ with $(\text{tail}(a_{j,1}), \text{head}(a_{j,1})) = (c_j, s)$;
  - $A_{\text{clause,sat}}$ contains an arc from every clause vertex to s_{sat}; more precisely $A_{\text{clause,sat}} = \{a_{j,0}\}_{j \in \{1,\dots,m\}}$ with $(\text{tail}(a_{j,0}), \text{head}(a_{j,0})) = (c_j, s_{\text{sat}})$.

Figure 13 shows an example of this core multigraph.

In the following, we extend this graph by adding two gadgets called respectively *clause gadget* and *variable gadget* that will ensure that any legal routing for the LR-GRMM instance is coherent with the choice of a satisfying assignment of the 3-SAT-(2,2) instance, if any.

Clause gadget. In the core multigraph, for each clause vertex $c_j, j \in \{1,\dots,m\}$, we have $A^+(c_j) = \{a_{j,0}, a_{j,1}\}$. We add for every j a set of faces $F(c_j) = \{f^{01}_j, f^{11}_j\}$ where $(\text{tail}^A(f^{01}_j), \text{head}^A(f^{01}_j)) = (a_{j,0}, a_{j,1})$ and $(\text{tail}^A(f^{11}_j), \text{head}^A(f^{11}_j)) = (a_{j,1}, a_{j,1})$. See Fig. 14 for an illustration of the clause gadget.

Variable gadget. For every $x_i, i \in \{1,\dots,n\}$, we add two sink vertices s^{start}_i and s^{end}_i, and two arcs a^{start}_i and a^{end}_i, respectively from x_i to s^{start}_i and s^{end}_i. At this point, we have

$$A^+(x_i) = \{a^{\text{start}}_i, a^{\text{end}}_i, a^+_{i,1}, a^+_{i,2}, a^-_{i,1}, a^-_{i,2}\}$$

where:

- $a^+_{i,1}, a^+_{i,2}$ (resp. $a^-_{i,1}, a^-_{i,2}$) are arcs whose heads are the clause vertices where x_i appears unnegated (resp. negated)
- a^{start}_i is such that $\text{head}(a^{\text{start}}_i) = s^{\text{start}}_i$

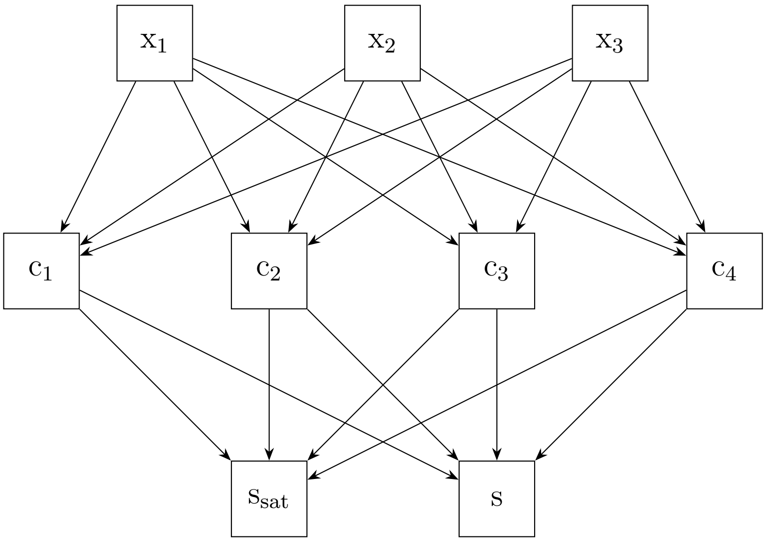


Figure 13: core multigraph built from the 3-SAT-(2,2) instance $c_1 \wedge c_2 \wedge c_3 \wedge c_4 = (x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee \overline{x_2} \vee \overline{x_3}) \wedge (\overline{x_1} \vee x_2 \vee \overline{x_3}) \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3)$. The graph does not differentiate among unnegated and negated variables.

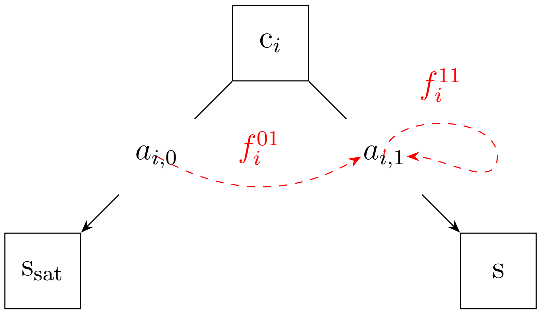


Figure 14: The clause gadget for c_i. Faces are represented by dashed red arcs.

- a_i^{end} is such that $\text{head}(a_i^{\text{end}}) = s_i^{\text{end}}$.

We now add for every i the sets of faces $F(x_i) = \{f_{i,1}^+, f_{i,2}^+, f_{i,3}^+, f_{i,1}^-, f_{i,2}^-, f_{i,3}^-\}$, as illustrated in Fig. 15. Faces $f_{i,1}^+, f_{i,2}^+, f_{i,3}^+$ are the *positive faces* of x_i and $f_{i,1}^-, f_{i,2}^-, f_{i,3}^-$ are the *negative faces* of x_i.

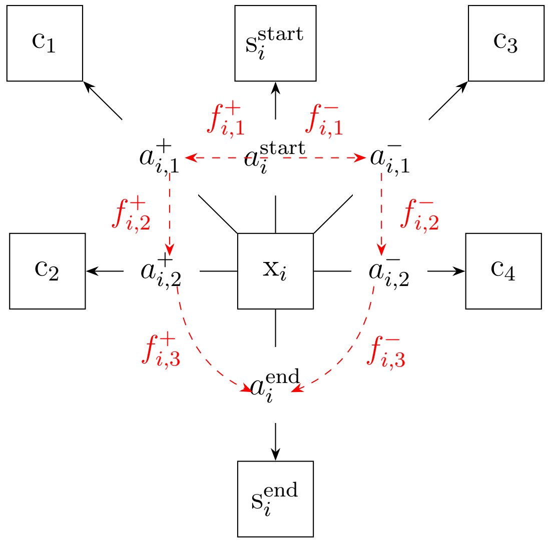


Figure 15: The variable gadget for x_i which is assumed to appear unnegated in clauses c_1 and c_2, and negated in clause c_3 and c_4. Faces are represented by dashed red arcs. The positive faces are $f_{i,1}^+, f_{i,2}^+, f_{i,3}^+$, and the negative faces are $f_{i,1}^-, f_{i,2}^-, f_{i,3}^-$.

Full specification of the LR-GRMM instance. Finally, to a 3-SAT-(2,2) instance, we associate the LR-GRMM problem $((G^A, G^V), r, r', \sigma, \sigma')$ defined by

- (G^A, G^V) is the GRM multigraph as described above by union of the core multigraph, and all variable and clause gadgets;
- $\sigma = 3\sum_{i=1}^{n} x_i$ (i.e. three particles on each variable vertex) ;
- $\sigma' = ms_{\text{sat}} + (2n - m)s + \sum_{i=1}^{n} s_i^{\text{start}}$;
- $r = \sum_{i=1}^{n} a_i^{\text{start}} + \sum_{j=1}^{m} a_{j,0}$;
- $r' = \sum_{i=1}^{n} a_i^{\text{end}} + \sum_{j=1}^{m} a_{j,1}$.

Proof of the reduction. It is given in two separate lemmas.

Lemma 23. *If the* 3-SAT-(2,2) *instance is satisfiable, then the corresponding instance of* LR-GRMM *has a solution.*

Proof. Assume that the 3-SAT-(2,2) instance has a satisfiable assignment. We will construct a solution for the corresponding LR-GRMM problem $((G^A, G^V), r, r', \sigma, \sigma')$ by giving a routing vector $\phi = \phi^{\text{var}} + \phi^{\text{clause}}$, where the supports of ϕ^{var} and ϕ^{clause} are respectively contained in the union of the faces set $F(x_i)$ for all variable vertices x_i, and in the union of $F(c_i)$ for all clause vertices c_i. We will prove that $(r, \sigma) \underset{\mathcal{L}}{\overset{\phi}{\to}} (r', \sigma')$.

Let $\phi^{\text{var}} \in C_F$ be defined as $\phi^{\text{var}} = \sum_{i=1}^{n} \phi_i^{\text{var}}$, where $\phi_i^{\text{var}} = f_{i,1}^+ + f_{i,2}^+ + f_{i,3}^+$ if the satisfying assignment of x_i is true, and $\phi_i^{\text{var}} = f_{i,1}^- + f_{i,2}^- + f_{i,3}^-$ if it is false. In both cases, note that $\partial^A(\phi_i^{\text{var}}) = a_i^{\text{end}} - a_i^{\text{start}}$.

If we define $(r_0, \sigma_0) = (r, \sigma) + \mathcal{L}(\phi_{\text{var}})$, then it follows that $r_0 = \sum_{i=1}^{n} a_i^{\text{end}} + \sum_{j=1}^{m} a_{j,0}$. On the other hand, all particles were transferred in this routing from variable vertices x_i to clause vertices c_j and sinks s_i^{start}, and we can write $\sigma_0 = \sum_{i=1}^{m} s_i^{\text{start}} + \sum_{j=1}^{m} \ell_j c_j$ for some integers $\ell_1, \ldots, \ell_m$ satisfying $1 \leqslant \ell_j \leqslant 3$ for all $1 \leqslant j \leqslant m$, and $\sum_{i=1}^{m} \ell_i = 2n$ (the fact that $\ell_j \geqslant 1$ follows from the assignment satisfying the formula). Note that this routing can be done legally, simply by following the order $1, 2, 3$ on faces.

Then, consider ϕ_{clause} as the routing vector consisting in legally routing from (r_0, σ_0) all particles ℓ_j from clause vertices c_j to sinks s_{sat} and s, for all $1 \leqslant j \leqslant m$. Since $1 \leqslant \ell_j \leqslant 3$ for all $1 \leqslant j \leqslant m$, at least one outgoing arc of each clause vertex c_i is routed and will emit one particle to s_{sat}, whereas the other routed arcs in $A^+(c_j)$, if any, will emit particles to sink s. Then we have $(r', \sigma') = (r, \sigma) + \mathcal{L}(\phi^{\text{var}} + \phi^{\text{clause}})$ and the whole routing can be done legally. ☐

Lemma 24. *Let* $((G^A, G^V), r, r', \sigma, \sigma')$ *be the* LR-GRMM *problem associated to a given* 3-SAT-(2,2) *instance. If* $((G^A, G^V), r, r', \sigma, \sigma')$ *has a solution, then the* 3-SAT-(2,2) *instance is satisfiable.*

Proof. Assume that $((G^A, G^V), r, r', \sigma, \sigma')$ has a solution and consider a legal routing sequence from (r, σ) to (r', σ').

Consider a variable vertex x_i with $i \in \{1, \ldots, n\}$: since there are 3 particles on x_i in σ, that in r the only arc of $A^+(x_i)$ is a_i^{start}, and that in r' the only arc of $A^+(x_i)$ is a_i^{end}, we see that the only possible legal routings in $A^+(x_i)$ are in that order ($f_{i,1}^+$, $f_{i,2}^+$, $f_{i,3}^+$) or ($f_{i,1}^-$, $f_{i,2}^-$, $f_{i,3}^-$) (consider Fig. 15). Hence, one particle is routed to s_i^{start}, and the other two are either routed to the 2 clause vertices where x_i appears negated, or to the 2 clause vertices where x_i appears unnegated.

Consider now a clause vertex c_j with $j \in \{1, \ldots, m\}$: since the only arc in r of $A^+(c_j)$ is $a_{j,0}$, and in r' the only arc of $A^+(c_j)$ is $a_{j,1}$, we see that there must be at least one particle routed from a variable vertex to c_i during the routing sequence.

From this, we can built a truth assignment, by letting x_i be true if and only if two of its 4 particles were routed to the 2 clauses where x_i is unnegated. We then see that the 3-SAT-(2,2) problem is satisfiable. ☐

6.2 Cyclic case

We now consider the case where the GRM multigraph is cyclic. We will demonstrate that the Legal Reachability in Cyclic GRM Multigraph problem can be solved in polynomial time. To achieve this, a deeper understanding of the set of routing vectors between the two configurations in question is required.

6.2.1 Routing vectors in cyclic GRM multigraphs

A natural question related to the legal reachability problem is to determine, when $(r, \sigma) \overset{*}{\underset{\mathcal{L}}{\sim}} (r', \sigma')$, how many routing vectors exist between the two arc-particle configurations. Answering this question in the general case of GRM multigraphs is challenging, as it heavily depends on the topology of the mechanisms. However, in the case of a cyclic instance, we can provide a solution.

Suppose that we have

$$(r - r', \sigma - \sigma') = \mathcal{L}(\phi_1) = \mathcal{L}(\phi_2).$$

It follows that $\mathcal{L}(\phi_1 - \phi_2) = 0$, i.e. $\partial^A(\phi_1 - \phi_2) = 0$ and $\partial^V \circ \mathrm{tail}^A(\phi_1 - \phi_2) = 0$.

With a slight abuse of notation, let us define for all $v \in V$ the sum $F(v) \in C_F$ as the sum of faces in $G^A(v)$, i.e. $\sum_{f \in F(v)} f$, and extend F to an homomorphism $F : C_V \to C_F$. For $p \in C_V$, $F(p)$ can be thought as a routing vector that makes p_v full turns on the cyclic rotor at v, for every $v \in V$. We can then use the following result (recall that primitive period vectors are defined in Theorem 2).

Proposition 25. *Let (G^A, G^V) be a cyclic GRM multigraph and let k be the number of leaf components of G that are not singletons $\{s\}$ where $s \in S$ is a sink. Then the rank of $\ker(\partial^A) \cap \ker(\partial^V \circ \mathit{tail}^A)$ is k. More precisely, let $p_1, p_2, \ldots, p_k \in C_V$ be the primitive period vectors of G^V corresponding to these components. A basis of $\ker(\partial^A) \cap \ker(\partial^V \circ \mathit{tail}^A)$ is then $(F(p_1), F(p_2), \ldots, F(p_k))$.*

Proof. With notation above, it is easy to see that the Laplacian operator Δ is $\Delta = \partial^V \circ \mathrm{tail}^A \circ F$.

Because of the cyclic structure of $G^A(v)$ for every $v \in V \setminus S$, by Proposition 8 it is easy to see that $(F(v))_{v \in V \setminus S}$ is a basis of $\ker(\partial^A)$, and that F induces an isomorphism from $C_{V \setminus S} \subset C_V$ onto $\ker(\partial^A)$. An element $F(\sigma) \in \ker(\partial^A)$, with $\sigma \in C_{V \setminus S}$, is then in $\ker(\partial^V \circ \mathrm{tail}^A)$ if and only if $\partial^V \circ \mathrm{tail}^A(F(\sigma)) = \Delta(\sigma) = 0$. Hence,

$$\ker(\partial^A) \cap \ker(\partial^V \circ \mathrm{tail}^A) = F(\ker(\Delta)).$$

By Theorem 2, a basis for $\ker(\Delta)$ is given by $p_1, p_2, \ldots, p_k$ together with each element of S, and so $(F(p_i))_{1 \leqslant i \leqslant k}$ is a basis of $F(\ker(\Delta))$. □

Interpreting this result for standard rotor-routing, we can state that:

Corollary 26. *Let (G^A, G^V) be a cyclic rotor multigraph. Then:*

- *if (G^A, G^V) is stopping and $(r, \sigma) \overset{*}{\underset{\mathcal{L}}{\sim}} (r', \sigma')$ then the routing vector from (r, σ) to (r', σ') is unique;*

- *if* (G^A, G^V) *is strongly connected and* $(r,\sigma) \overset{\phi}{\underset{\mathcal{L}}{\sim}} (r',\sigma')$ *then routing vectors are of the form* $\phi + kp_1$ *with* $k \in \mathbb{Z}$*, and* $\ker(\partial^A) \cap \ker(\partial^V \circ tail^A) = \mathbb{Z} \cdot F(p_1)$*, with* $p_1 > 0$*,* $p_1 \in C_V$*.*

Proof. In the first case, note that all leaf components are of the form $\{s\}$ with $s \in S$, hence $\ker(\partial^A) \cap \ker(\partial^V \circ \mathrm{tail}^A)$ is trivial.

In the second case, there is a single primitive vector $p_1 > 0$ and $\ker(\partial^A) \cap \ker(\partial^V \circ \mathrm{tail}^A)$ has rank 1. □

In the case of a strongly connected multigraph, the existence of p_1 implies that if $(r,\sigma) \overset{*}{\underset{\mathcal{L}}{\sim}} (r',\sigma')$ then there is a minimal routing vector $\phi_1 \geqslant 0$ from (r,σ) to (r',σ'). In other words, if $\phi \geqslant 0$ is a routing vector from (r,σ) to (r',σ') then $\phi_1 \leqslant \phi$.

To compute ϕ_1, one can start with any routing vector ϕ such that $(r,\sigma) \overset{\phi}{\underset{\mathcal{L}}{\sim}} (r',\sigma')$, which can be found using Proposition 17. Then $\phi_1 = \phi + k^* p_1$, where k^* is the smallest integer k such that $\phi + kF(p_1)$ is nonnegative.

6.2.2 Legal reachability in strongly connected cyclic GRM multigraphs

Let (G^A, G^V) be a strongly connected cyclic GRM multigraph and let $p \in C_V^+$ be the primitive period vector. We say that $(r,\sigma) \in C_A \times C_V$ is *recurrent* if there is a legal routing sequence to itself with routing vector $F(p)$, i.e. $(r,\sigma) \overset{F(p)}{\underset{\mathcal{L}}{\to}} (r,\sigma)$.

Lemma 27. *Let* (G^A, G^V) *be a strongly connected cyclic GRM multigraph. Suppose that* $(r,\sigma) \overset{\phi_1}{\underset{\mathcal{L}}{\to}} (r',\sigma')$ *for some routing vector* $\phi_1 \in C_F^+$ *such that* $\phi_1 \geqslant F(p)$*. Let* $\phi_0 \in C_F^+$ *be the smallest nonnegative routing vector from* (r,σ) *to* (r',σ')*. Then* (r',σ') *is recurrent and* $(r,\sigma) \overset{\phi_0+F(p)}{\underset{\mathcal{L}}{\longrightarrow}} (r',\sigma')$*.*

Proof. Let $\phi \in C_F$ satisfy $\phi \geqslant F(p)$ and $(r,\sigma) \overset{\phi}{\underset{\mathcal{L}}{\sim}} (r',\sigma')$.

According to Proposition 20, we have $(r,\sigma) \overset{\phi}{\underset{\mathcal{L}}{\to}} (r',\sigma')$ if and only if the following conditions hold, where $\alpha = \mathrm{tail}^A(\phi)$ and $T_A = \{a \in \alpha : r'_{\theta(a)} > 0\} \cup \{a \in \alpha : \theta(a) \notin \alpha\}$:

(i) $\sigma'_v \geqslant 0$ for every vertex v active in α;

(ii) $r'_a \geqslant 0$ for every arc $a \in \alpha$;

(iii) $T_A \cap A^+(v) \neq \emptyset$ for every vertex v active in α;

(iv) T_A is guiding for $\sigma \overset{\alpha}{\underset{\partial^V}{\sim}} \sigma'$.

Given that $\sum_{v \in V} v \leqslant p$, and $F(p) \leqslant \phi$, it follows that all arcs are in α and all vertices are active in α. Hence $\{a \in \alpha : \theta(a) \notin \alpha\}$ is an empty set, i.e. $T_A = \{a \in \alpha : r'_{\theta(a)} > 0\}$. Then the set of conditions (i), (ii), (iii) and (iv) can be simplified as:

(1) $r' \in C_A^+$;

(2) $\sigma' \in C_V^+$;

(3) for every vertex $v \in V$, $\deg_v^A(r') \geqslant 1$;

(4) for every $v \in V$ such that $\sigma'(v) = 0$, there is a directed path within the set $\{a \in A : r'_{\theta(a)} > 0\}$ to a vertex v' where $\sigma'(v') > 0$.

Therefore, these conditions do not depend on the initial configuration (r, σ), but only on (r', σ'). Since they are satisfied with $\phi = \phi_1$, it follows that $(r', \sigma') \xrightarrow[\mathcal{L}]{F(p)} (r', \sigma')$ is also true, as well as $(r, \sigma) \xrightarrow[\mathcal{L}]{\phi_0 + F(p)} (r', \sigma')$.

□

The contrapositive of this lemma indicates that any legal routing sequence from (r, σ) to a non-recurrent configuration (r', σ') with routing vector $\phi \in C_F^+$ must satisfy the condition that there exists a face $f \in F$ such that $\phi_f < p_a$ where p is the primitive period routing vector and $a = \mathrm{tail}^A(f)$. This is stated in the following corollary.

Corollary 28. *Let (G^A, G^V) be a strongly connected cyclic GRM multigraph. Assume that (r', σ') is not recurrent. If there is $\phi \in C_F^+$ such that $(r, \sigma) \xrightarrow[\mathcal{L}]{\phi} (r', \sigma')$, then ϕ is the smallest nonnegative vector such that $(r, \sigma) \overset{\phi}{\underset{\mathcal{L}}{\sim}} (r', \sigma')$.*

It has been shown in [23] that, in the classical routor-routing framework, if there is a legal routing sequence from (ρ, σ) to (ρ', σ'), then there is a legal routing sequence with the smallest nonnegative routing vector, known as reduced routing vector. However, this does not hold in the more general context of cyclic GRM multigraphs, as illustrated in Figure 16.

Based on these results, the following proposition provides a unique routing vector that can be used to determine whether a legal routing sequence exists.

Proposition 29. *Let (G^A, G^V) be a strongly connected cyclic GRM multigraph. Let (r, σ) and (r', σ') such that $(r', \sigma') \overset{*}{\underset{\mathcal{L}}{\sim}} (r, \sigma)$. Let $p \in C_V^+$ be the primitive period vector and $\phi_0 \in C_F^+$ be the smallest nonnegative routing vector from (r, σ) to (r', σ'). Let $\phi = \phi_0 + F(p)$ if (r', σ') is recurrent, or $\phi = \phi_0$ otherwise. Then $(r, \sigma) \xrightarrow[\mathcal{L}]{*} (r', \sigma')$ if and only if $(r, \sigma) \xrightarrow[\mathcal{L}]{\phi} (r', \sigma')$.*

6.2.3 General case

We can now state the main result of this section, i.e. the existence of a polynomial algorithm for legal reachability in the general case of cyclic GRM. This extends Theorem 3.4 from Tóthmérész [23] to the context of linear rotor-routing, and to the case where the starting arc configuration is not a rotor configuration.

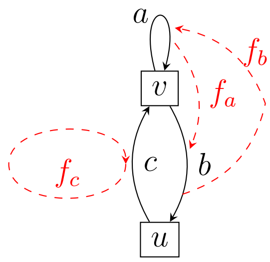


Figure 16: This is an example to prove that for strongly connected cyclic GRM multigraphs, the smallest ϕ such that $(r,\sigma) \xrightarrow[\mathcal{L}]{\phi} (r',\sigma')$ is not always the smallest nonnegative vector ϕ_0 such that $(r,\sigma) \overset{\phi_0}{\underset{\mathcal{L}}{\sim}} (r',\sigma')$; however $\phi_1 = \phi_0 + F(p)$ always works if (r',σ') is recurrent. In the case above, the primitive period vector is $p = u + v$ and $\text{tail}^A(F(u+v)) = a + b + c$. Let $\sigma = \sigma' = u$, $r = 2a + c$ and $r' = a+b+c$. We have $(r',\sigma') = (r,\sigma) + \mathcal{L}(f_a)$. The smallest nonnegative routing vector from (r,σ) to (r',σ') is f_a. However, there does not exist a legal sequence with this routing vector. Instead, a legal routing sequence from (r,σ) to (r',σ') exists with the routing vector $f_a + (f_a + f_b + f_c)$, which corresponds to routing the sequence of faces f_c, f_a, f_a, and f_b respectively. Note that (r',σ') is a recurrent configuration.

Theorem 30. *The restriction of* Legal Reachability in GRM multigraph *to cyclic GRM multigraphs is in P.*

Proof. Let (G^A, G^V) be a cyclic GRM multigraph and let $(r,\sigma), (r',\sigma') \in C_A \times C_V$ such that $(r,\sigma) \overset{*}{\underset{\mathcal{L}}{\sim}} (r',\sigma')$. We shall prove that we can construct in polynomial time a routing vector $\tilde{\phi} \in C_F$ such that $(r,\sigma) \xrightarrow[\mathcal{L}]{*} (r',\sigma')$ if and only if $(r,\sigma) \xrightarrow[\mathcal{L}]{\tilde{\phi}} (r',\sigma')$.

To do this, we consider the k leaf components $V_1, V_2, \cdots, V_k$ of G^V, and decompose faces and arcs in (G^A, G^V) as $F_1, F_2, \cdots F_k$ and $A_1, A_2, \cdots, A_k$, according to which component they belong to. Define V_0, A_0 and F_0 for elements that do not belong to a leaf component. This partitions F, A and V in $k+1$ subsets each.

Thus, we obtain strongly connected cyclic GRM multigraphs $(G_1^A, G_1^V), \ldots, (G_k^A, G_k^V)$. We also denote by (G_0^A, G_0^V) the stopping cyclic GRM multigraph induced by (G^A, G^V) on the arc set A_0. The faces of (G_0^A, G_0^V) are F_0, its arcs are A_0, and its vertices are V_0 together with $\text{head}^V(a)$ for every $a \in A_0$. If $\text{head}^V(a) \notin V_0$, then it is a sink of (G_0^A, G_0^V).

Assume that there is $\phi \in C_F^+$ such that $(r,\sigma) \xrightarrow[\mathcal{L}]{\phi} (r',\sigma')$. For every $0 \leqslant i \leqslant k$, let $\phi^i \in C_F^+$ such that $\phi^i_f = \phi_f$ if $f \in F_i$ and $\phi^i_f = 0$ otherwise, so that $\phi = \sum_{i=0}^k \phi^i$. By definition of leaf components, any routing in a leaf component will not change configurations outside of that component. Hence, there is a legal routing sequence that routes all faces of ϕ^0 first, then all faces of ϕ^1, ϕ^2 and so on until ϕ^k. For $0 \leqslant i \leqslant k$, let r_i (resp. σ_i) be equal to r' (resp. σ') for arcs (resp. vertices) in A_j (resp. V_j) for $j \leqslant i$, and to r (resp. σ) for arcs (resp. vertices) in A_j (resp. V_j) for $k \geqslant j > i$. We note that $(r_0, \sigma_0) = (r, \sigma) + \mathcal{L}(\phi^0)$, and $(r_i, \sigma_i) = (r_{i-1}, \sigma_{i-1}) + \mathcal{L}(\phi^i)$ for all $1 \leqslant i \leqslant k$. By identifying the routing along ϕ^i for $1 \leqslant i \leqslant k$ to a routing in the strongly connected cyclic GRM $(G_i^{A_i}, G_i^{V_i})$, we obtain by Proposition 29 a canonical routing vector $\tilde{\phi}^i \in C_F^+$, computable in polynomial time, such that the support of $\tilde{\phi}^i$ is in F_i, and $(r_{i-1}, \sigma_{i-1}) \xrightarrow[\mathcal{L}]{\tilde{\phi}^i} (r_i, \sigma_i)$. Then, by Corollary 26, there is a unique routing vector from (r, σ) to (r_0, σ_0), namely ϕ^0, since (G_0^A, G_0^V) is stopping. Hence, there is a legal routing sequence with routing vector $\tilde{\phi} = \phi^0 + \tilde{\phi}^1 + \cdots + \tilde{\phi}^k$.

All in all, deciding whether there is a legal routing sequence from (r, σ) to (r', σ') is equivalent to checking if there is a legal routing sequence from (r, σ) to (r', σ') with routing vector $\tilde{\phi}$. This can be checked in polynomial time according to Proposition 20. □

References

[1] A. Björner and L. Lovász. Chip-firing games on directed graphs. *Journal of algebraic combinatorics*, 1:305–328, 1992. Publisher: Springer.

[2] A. Björner, L. Lovász, and P. W. Shor. Chip-firing games on graphs. *European Journal of Combinatorics*, 12(4):283–291, 1991. Publisher: Elsevier.

[3] A. Darmann and J. Döcker. On simplified NP-complete variants of Monotone 3-Sat. *Discrete Applied Mathematics*, 292:45–58, 2021.

[4] E. A. Dinic. Algorithm for solution of a problem of maximum flow in networks with power estimation. In *Soviet Math. Doklady*, volume 11, pages 1277–1280, 1970.

[5] J. Dohrau, B. Gärtner, M. Kohler, J. Matoušek, and E. Welzl. ARRIVAL: A zero-player graph game in NP ∩ coNP. In *A journey through discrete mathematics*, pages 367–374. Springer, 2017.

[6] L. Duhaze-Pradines. *Reachability problems for general rotor walks in graphs.* PhD thesis, Université Paris-Saclay, 2023.

[7] J. Edmonds and R. M. Karp. Theoretical improvements in algorithmic efficiency for network flow problems. *Journal of the ACM (JACM)*, 19(2):248–264, 1972.

[8] M. Farrell and L. Levine. CoEulerian graphs. *Proceedings of the American Mathematical Society*, 144(7):2847–2860, 2016.

[9] G. P. Giacaglia, L. Levine, J. Propp, and L. Zayas-Palmer. Local-to-global principles for the hitting sequence of a rotor walk. *The electronic journal of combinatorics (EJC)*, Volume 19, Issue 1, 2012.

[10] B. Gärtner, T. D. Hansen, P. Hubácek, K. Král, H. Mosaad, and V. Slívová. ARRIVAL: Next Stop in CLS. In *45th International Colloquium on Automata, Languages, and Programming (ICALP 2018).* Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

[11] B. Gärtner, S. Haslebacher, and H. P. Hoang. A Subexponential Algorithm for ARRIVAL. In *ICALP 2021*, volume 198, pages 69:1–69:14, 2021.

[12] A. Hatcher. *Algebraic Topology.* Cambridge University Press, 2002.

[13] A. E. Holroyd, L. Levine, K. Mészáros, Y. Peres, J. Propp, and D. B. Wilson. Chip-Firing and Rotor-Routing on Directed Graphs. In *In and Out of Equilibrium 2*, pages 331–364. Springer, 2008.

[14] R. Kannan and A. Bachem. Polynomial algorithms for computing the smith and hermite normal forms of an integer matrix. *SIAM Journal on Computing*, 8(4):499–507, 1979.

[15] R. M. Karp and R. E. Miller. Parallel program schemata. *Journal of Computer and system Sciences*, 3(2):147–195, 1969.

[16] S. Lang. *Algebra*, volume 211. Springer Science & Business Media, 2012.

[17] E. W. Mayr. An algorithm for the general petri net reachability problem. In *Proceedings of the thirteenth annual ACM symposium on Theory of computing*, pages 238–246, 1981.

[18] M. Newman. The Smith normal form. *Linear algebra and its applications*, 254(1-3):367–381, 1997.

[19] J. Pitman and W. Tang. Tree formulas, mean first passage times and Kemeny's constant of a Markov chain. *Bernoulli*, 24(3):1942–1972, 2018.

[20] A. Povolotsky, V. Priezzhev, and R. Shcherbakov. Dynamics of Eulerian walkers. *Physical review E*, 58(5):5449, 1998. Publisher: APS.

[21] V. B. Priezzhev, D. Dhar, A. Dhar, and S. Krishnamurthy. Eulerian walkers as a model of self-organized criticality. *Physical Review Letters*, 77(25):5079, 1996. Publisher: APS.

[22] L. Tóthmérész. Algorithmic aspects of rotor-routing and the notion of linear equivalence. *Discrete Applied Mathematics*, 236:428–437, 2018. Publisher: Elsevier.

[23] L. Tóthmérész. Rotor-routing reachability is easy, chip-firing reachability is hard. *European Journal of Combinatorics*, 101:103466, 2022.

[24] T. van Aardenne-Ehrenfest and N. G. de Bruijn. Circuits and trees in oriented linear graphs. *Classic papers in combinatorics*, pages 149–163, 1987.

[25] D. B. Wilson and J. G. Propp. How to get an exact sample from a generic markov chain and sample a random spanning tree from a directed graph, both within the cover time. In *SODA*, pages 448–457. Citeseer, 1996.

[26] V. Yanovski, I. A. Wagner, and A. M. Bruckstein. A Distributed Ant Algorithm for Efficiently Patrolling a Network. *Algorithmica*, 37(3):165–186, Nov. 2003.

A Additional examples

A.1 Standard rotor-routing and flows

A maximal rotor walk is processed on the graph G_2 of Fig. 3. with starting configurations (ρ, σ) where $\rho(v_2) = a_{2,4}$, $\rho(v_3) = a_{3,4}$, $\rho(v_4) = a_{4,2}$ together with $\sigma = 3v_2 + 6v_3 + 3v_4$.

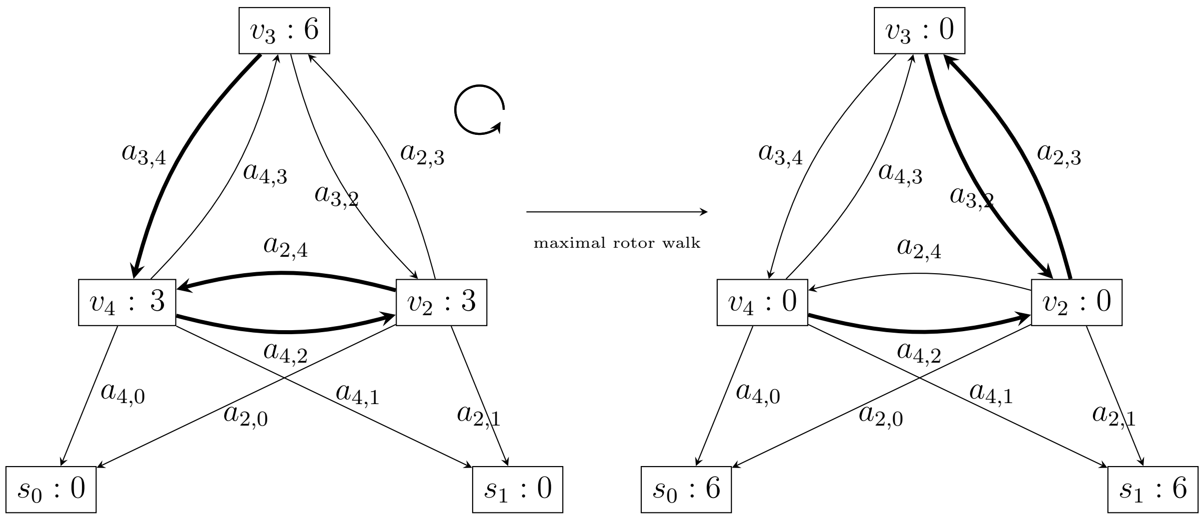


Figure 17: Initial configurations (σ, ρ) and result of a maximal rotor walk (σ', ρ') in G_2.

The walk ends in (ρ', σ'), with $\rho'(v_2) = a_{2,3}$, $\rho'(v_3) = a_{3,2}$, $\rho'(v_4) = a_{4,2}$ together with $\sigma' = 6s_0 + 6s_1$, as depicted on Fig. 17.

Let us state the equations satisfied by flows for (σ, ρ, σ'):

$$
\begin{aligned}
f(a_{4,0}) + f(a_{2,0}) &= 6 && \text{(flow at } s_0) \\
f(a_{4,1}) + f(a_{2,1}) &= 6 && \text{(flow at } s_1) \\
f(a_{3,2}) + f(a_{4,2}) + 3 &= f(a_{2,0}) + f(a_{2,1}) + f(a_{2,3}) + f(a_{2,4}) && \text{(flow at } v_2) \\
f(a_{2,3}) + f(a_{4,3}) + 6 &= f(a_{3,2}) + f(a_{3,4}) && \text{(flow at } v_3) \\
f(a_{2,4}) + f(a_{3,4}) + 3 &= f(a_{4,0}) + f(a_{4,1}) + f(a_{4,2}) + f(a_{4,3}) && \text{(flow at } v_4) \\
f(a_{2,4}) \geqslant f(a_{2,0}) &\geqslant f(a_{2,1}) \geqslant f(a_{2,3}) \geqslant f(a_{2,4}) - 1 && \text{(rotor at } v_2) \\
f(a_{3,4}) \geqslant f(a_{3,2}) &\geqslant f(a_{3,4}) - 1 && \text{(rotor at } v_3) \\
f(a_{4,2}) \geqslant f(a_{4,3}) &\geqslant f(a_{4,0}) \geqslant f(a_{4,1}) \geqslant f(a_{4,2}) - 1 && \text{(rotor at } v_4)
\end{aligned}
$$

We give the values of the run for (ρ, σ) (left of Fig. 18) and a flow (which is not the run) for (ρ, σ, σ') (right of Fig. 18).

A.2 Computing routing vectors

We consider here an example of linear routing in a cyclic GRM multigraph, and show how to compute a routing vector. Namely, we consider the cyclic GRM version of the standard rotor-routing between (ρ, σ) and (ρ', σ') in the cyclic GRM Multigraph corresponding to G_2, as in A.1. We know that the solution is unique by Corollary 26 and corresponds to the flow given in Fig.18 (left). For every arc $a_{i,j}$, we denote by $f_{i,j}$ the unique face that has $a_{i,j}$ as a tail.

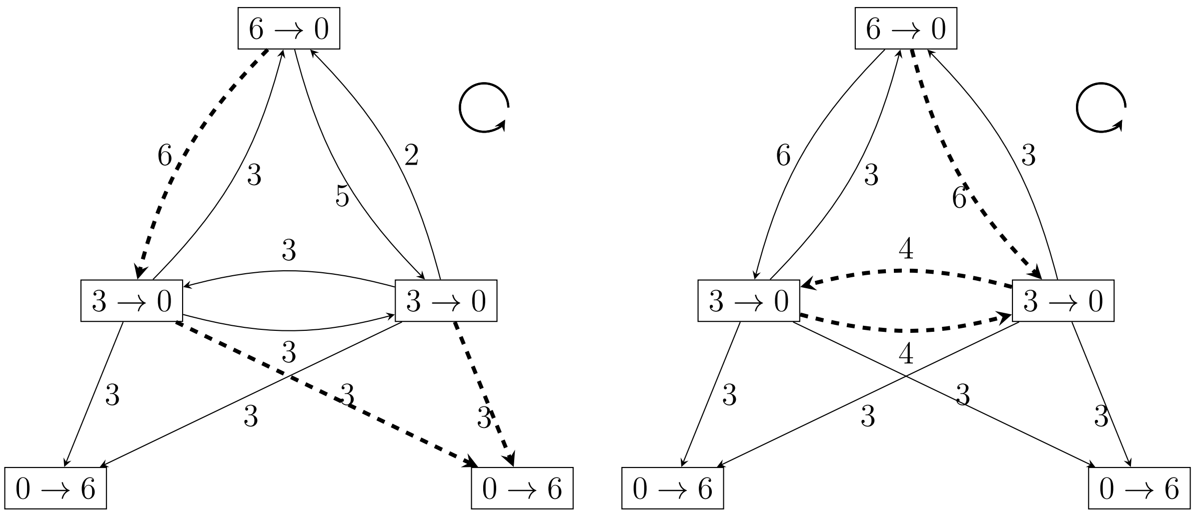


Figure 18: On the left, the values on each arc of the run corresponding to the maximal rotor walk of Fig. 17. The dashed arcs correspond to the last routed particle on every vertex and contain no cycles, which characterizes the run by Theorem 5. On the right, the values of a flow, which also certifies final configuration σ', but is not the run.

Let us form the matrix L of $\mathcal{L}$ in the canonical basis of C_F and $C_A \times C_V$. The first 10 lines correspond to ∂^A and the last 5 to $\partial^V \circ \mathrm{tail}$. We write only nonzero coefficients.

$$L = \begin{array}{c} \\ a_{2,0} \\ a_{2,1} \\ a_{2,3} \\ a_{2,4} \\ a_{3,2} \\ a_{3,4} \\ a_{4,0} \\ a_{4,1} \\ a_{4,2} \\ a_{4,3} \\ s_0 \\ s_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccccccccc} f_{2,0} & f_{2,1} & f_{2,3} & f_{2,4} & f_{3,2} & f_{3,4} & f_{4,0} & f_{4,1} & f_{4,2} & f_{4,3} \end{array} \\ \left(\begin{array}{cccc|cc|cccc} -1 & & & 1 & & & & & & \\ 1 & -1 & & & & & & & & \\ & 1 & -1 & & & & & & & \\ & & 1 & -1 & & & & & & \\ \hline & & & & -1 & 1 & & & & \\ & & & & 1 & -1 & & & & \\ \hline & & & & & & -1 & & & 1 \\ & & & & & & 1 & -1 & & \\ & & & & & & & 1 & -1 & \\ & & & & & & & & 1 & -1 \\ \hline 1 & & & & & & 1 & & & \\ & 1 & & & & & & 1 & & \\ -1 & -1 & -1 & -1 & 1 & & & & 1 & \\ & & 1 & & -1 & -1 & & & & 1 \\ & & & 1 & & 1 & -1 & -1 & -1 & -1 \end{array}\right) \end{array}$$

We want to solve $L \cdot \phi = (\rho' - \rho, \sigma' - \sigma)$ with

$$(\rho' - \rho, \sigma' - \sigma) = (0, 0, 1, -1|1, -1|0, 0, 0, 0|6, 6, -3, -6, -3)^T.$$

Since the solution is unique, the system can be solved in $\mathbb{Q}$, and it can be checked that

$$\phi = (3,3,2,3|5,6|3,3,3,3)^T$$

is the unique solution to this system. Note that this routing vector, in the context of cyclic GRM multigraphs, corresponds exactly to the run obtained in Fig. 18 (left) in the context of standard rotor-routing.

B Smith normal form

Integer linear systems can be solved by Gaussian elimination, but the so-called *Smith normal form* is a useful tool to understand the results. We use the following [18]:

Proposition 31. *Let A be a $n \times m$ integer matrix. There exist invertible matrices $S \in GL_n(\mathbb{Z}), T \in GL_m(\mathbb{Z})$, such that the product SAT is of the form*

$$\begin{pmatrix} \alpha_1 & 0 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & \alpha_2 & 0 & & & & \\ 0 & 0 & \ddots & & \vdots & & \vdots \\ \vdots & & & \alpha_r & & & \\ 0 & & \cdots & & 0 & \cdots & 0 \\ \vdots & & & & \vdots & & \vdots \\ 0 & & \cdots & & 0 & \cdots & 0 \end{pmatrix}$$

where diagonal elements satisfy $\alpha_i \geqslant 1$ for $1 \leqslant i \leqslant r$ and α_i is a divisor of α_{i+1} for $1 \leqslant i \leqslant r-1$. Moreover r is the rank of A, and all α_i are uniquely determined by these properties, since

$$\alpha_i = \left| \frac{d_i(A)}{d_{i-1}(A)} \right|$$

where $d_i(A)$ is the gcd of all $i \times i$ minors of A, and $d_0(A) = 1$.

Coefficients $\alpha_1, \alpha_2, \ldots, \alpha_r$ are called the **invariant factors** of A. We use this result to compute the solution of a system of linear diophantine equations.

Lemma 32. *Let A be an $n \times m$ integral matrix. Let $D = SAT$ be the normal Smith form of A and $\alpha_1, \alpha_2, \cdots, \alpha_r$ be the invariant factors of A. Let x be an $m \times 1$ integral vector and b an $n \times 1$ integral vector, the system of linear equations $Ax = b$ admits an integral solution if and only if*

$$c_i = \begin{cases} 0 \mod (\alpha_i) & \text{if } 1 \leqslant i \leqslant r \\ 0 & \text{if } r < i \leqslant n \end{cases}$$

where $c = Sb$. In this case, the set of solutions is obtained by all the vectors $x = Ty$ with $y \in \mathbb{Z}^m$ of dimension $(m, 1)$, where

$$y_i = c_i/\alpha_i \text{ if } 1 \leqslant i \leqslant r.$$

Proof. We have the following equivalences:

$$\begin{aligned} & Ax = b \\ \Leftrightarrow\ & SAT(T^{-1}x) = Sb \\ \Leftrightarrow\ & Dy = c \text{ with } y = T^{-1}x \text{ and } c = Sb \end{aligned}$$

In particular, since T has an integer inverse, there is a solution to $Ax = b$ if and only if there is one to $Dy = c$. There is a solution to the last equation if and only if $c_i = 0$ for $r < i \leqslant n$ and $c_i = 0 \mod (\alpha_i)$ for $1 \leqslant i \leqslant r$. In this case, a particular solution is $\bar{y} = (c_1/\alpha_1, \ldots, c_r/\alpha_r, 0, \ldots, 0)^\intercal$, and one obtains a solution to $Ax = b$ by choosing $\bar{x} = T\bar{y}$. The other solutions are of the form $y = (c_1/\alpha_1, \ldots, c_r/\alpha_r, y_{r+1}, \ldots, y_n)^\intercal$ for all choices of $y_{r+1}, \ldots, y_n$, and x is obtained by Ty. □