A Quasi-Polynomial Time Algorithm for Multi-Arrival on Tree-Like Multigraphs

Ebrahim Ghorbani
Hamburg University of Technology, Institute for Algorithms and Complexity, Hamburg, Germany

Jonah Leander Hoff
Hamburg University of Technology, Institute for Algorithms and Complexity, Hamburg, Germany

Matthias Mnich
Hamburg University of Technology, Institute for Algorithms and Complexity, Hamburg, Germany

Abstract

Propp machines, or rotor-router models, are a classic tool to simulate random systems in forms of Markov chains by deterministic systems. To this end, the nodes of the Markov chain are replaced by switching nodes, which maintain a queue over their outgoing arcs, and a particle sent through the system traverses the top arc of the queue which is then moved to the end of the queue and the particle arrives at the next node. A key question to answer about such systems is whether a single particle can reach a particular target node, given as input an initial configuration of the queues at all switching nodes. This question was introduced by Dohrau et al. (2017) under the name of ARRIVAL. A major open question is whether ARRIVAL can be solved in polynomial time, as it is known to lie in $\mathsf{NP} \cap \mathsf{co\text{-}NP}$; yet the fastest known algorithm for general instances takes subexponential time (Gärtner et al., ICALP 2021).

We consider a generalized version of ARRIVAL introduced by Auger et al. (RP 2023), which requires routing multiple (potentially exponentially many) particles through a rotor graph. The MULTI-ARRIVAL problem is to determine the particle configuration that results from moving all particles from a given initial configuration to sinks. Auger et al. showed that for path-like rotor graphs with a certain uniform rotor order, the problem can be solved in polynomial time.

Our main result is a quasi-polynomial-time algorithm for MULTI-ARRIVAL on tree-like rotor graphs for arbitrary rotor orders. Tree-like rotor graphs are directed multigraphs which can be obtained from undirected trees by replacing each edge by an arbitrary number of arcs in either or both directions. For trees of bounded contracted height, such as paths, the algorithm runs in polynomial time and thereby generalizes the result by Auger et al.. Moreover, we give a polynomial-time algorithm for MULTI-ARRIVAL on tree-like rotor graphs without parallel arcs.

2012 ACM Subject Classification Theory of computation → Randomness, geometry and discrete structures

Keywords and phrases Arrival, Rotor-routing, Tree-like Multigraph, Path-Like Multigraph, Fixed-Parameter Tractability

Digital Object Identifier 10.4230/LIPIcs.STACS.2025.39

Funding Funded by the project "Hamburg Quantum Computing", co-financed by ERDF and Fonds of the Hamburg Ministry of Science, Research, Equalities and Districts (BWFGB).

1 Introduction

In 2017, Dohrau et al. [3] introduced the ARRIVAL problem, which they intuitively defined as follows:

> "Suppose that a train is running along a railway network, starting from a designated origin, with the goal of reaching a designated destination. The network, however, is of a special nature: every time the train traverses a switch, the switch will change its position immediately afterwards. Hence, the next time the train traverses the

42nd International Symposium on Theoretical Aspects of Computer Science (STACS 2025).
Editors: Olaf Beyersdorff, Michał Pilipczuk, Elaine Pimentel, and Nguyễn Kim Thắng;
Article No. 39; pp. 39:1–39:19
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

same switch, the other direction will be taken, so that directions alternate with each traversal of the switch. Given a network with origin and destination, what is the complexity of deciding whether the train, starting at the origin, will eventually reach the destination?"

One may solve Arrival by running the train iteratively according to the outlined rules until reaching a destination. This may require a number of steps which is exponential in the number of switches. Gärtner et al. [6] devised an algorithm for Arrival which runs in subexponential time. It is further known that Arrival lies in $\mathsf{NP} \cap \mathsf{co\text{-}NP}$ [3], which motivated the conjecture that the problem is polynomial-time decidable. As of now, this conjecture is still open, despite some effort [4, 5, 6].

The routing behaviour in Arrival is also known as *switching networks*, *rotor routing*, *Propp machines*, and *Eulerian walkers* [4, 7, 10]. We adopt the nomenclature of rotor routing and refer to the trains as *particles*. In rotor routing, particles are *routed* from each node along its outgoing arcs in a fixed cyclic order called the *rotor order*. A directed multigraph equipped with a rotor order at each of its nodes is referred to as a *rotor graph*. Arrival can be seen as determining the unique *particle configuration* resulting from routing a single particle until it reaches a sink. As the complexity of Arrival in general is unkown, recent efforts focus on identifying families of instances on which Arrival can be solved in polynomial-time. One such family are *tree-like* rotor graphs, which are rotor graphs whose underlying undirected simple graphs are trees. Note that the class of tree-like rotor graphs is richer than its name may suggest: they can contain arcs in opposite direction, as well as parallel arcs, between the same two nodes. Tree-like rotor graphs are directed multi-graphs that can be obtained from undirected trees by replacing edge undirected edge $\{u, v\}$ by any number $a_{uv} \geq 0$ of arcs (u, v) and any number $a_{vu} \geq 0$ of arcs (v, u), where a_{uv} and a_{vu} can be different (not both of them should be zero). On tree-like rotor graphs, Arrival was shown to be solvable in polynomial time [1]. More recently, the case of routing multiple (potentially exponentially many) particles on *path-like* rotor graphs has been studied. Multi-Arrival refers to the problem of determining the particle configuration resulting from a routing that moves all particles from a given initial configuration to the sinks. This resulting configuration has been proved to be unique [2]. Auger et al. [2] showed that for path-like rotor graphs with a certain uniform rotor order, Multi-Arrival can be solved in polynomial-time. Results by Gärtner et al. [6] show that polynomial-time algorithms for Multi-Arrival for a certain family $\mathcal{F}$ of rotor graphs yield polynomial-time algorithms for Arrival for rotor graphs that belong to $\mathcal{F}$ after deleting constantly many nodes.

Our Contributions. Our main contributions are algorithms for Multi-Arrival on tree-like rotor graphs. To state them, we introduce the notion of *contracted height* $\mathsf{ch}(R)$ of a simple rooted tree R as the minimum height over all trees obtained from R by contracting, for each node v of R, one of the arcs to its children. We now state our main results informally, where $\kappa(T)$ is the maximum possible $\mathsf{ch}(\langle T \rangle)$ over all choices of a root, where $\langle T \rangle$ is the simple undirected underlying tree of T.

▶ **Theorem 1** (Informal). *Multi-Arrival can be solved on tree-like rotor graph T with $|A(T)|$ arcs in time $\mathcal{O}^*\big(\log^{\kappa(T)+1} |A(T)|\big)$, where $\mathcal{O}^*$ hides factors polynomial in the input. In particular,* Multi-Arrival *can be solved in polynomial time if $\kappa(T) = \mathcal{O}\big(\frac{\log |A(T)|}{\log(\log |A(T)|)}\big)$.*

We further show that if $\kappa(T) = \mathcal{O}(1)$ (which includes path-like rotor graphs), Multi-Arrival can be solved in time polynomial in $|\hat{A}(T)|$, where $\hat{A}(T)$ represents the *succinct* encoding of the arcs, only recording the number of consecutive parallel arcs. Therefore, our results widely generalize the previous algorithms by Auger et al. [2] for path-like rotor graphs with uniform rotor order.

As an important corollary, we deduce that Multi-Arrival on tree-like rotor graphs is fixed-parameter tractable when parameterized by $\kappa(T)$. That is, Multi-Arrival can be solved in time $f(\kappa(T)) \cdot m^{\mathcal{O}(1)}$ on tree-like rotor graphs of size m. Such fixed-parameter algorithms are considered to be advantageous over so-called XP-algorithms, where the degree of the polynomial depends on the parameter. Indeed, for the related parameter "feedback vertex set size" $\mathsf{fvs}(G)$, which measures the node deletion distance of the rotor graph G to an acyclic digraph, Gärtner et al. [6] provided an XP-algorithm with run time $m^{\mathcal{O}(\mathsf{fvs}(G))}$.

Finally, we show:

▶ **Theorem 2** (Informal). Multi-Arrival *is solvable in polynomial time for tree-like rotor graphs without parallel arcs.*

2 Preliminaries

2.1 Directed Multigraphs

Throughout, we consider *directed multigraphs* G with finite node set $V(G)$ and arc set $A(G)$. An arc with tail v and head w is said to be *from* v *to* w. A directed multigraph may have *parallel* arcs which share the same head-tail pair, and *self-loops* which are from and to the same node.

For a given node $v \in V(G)$, we denote by $A^+(v)$ the set of arcs whose tail is v and define $d^+(v) = |A^+(v)|$. Let $N^+(v)$ be the set of out-neighbours of v and let $N^-(v)$ be the set of in-neighbours of v. By $d(v, w)$ we denote the number of arcs from v to w. For a directed multigraph G, let $V^+(G)$ be the set of nodes with positive out-degree and let $S_0(G)$ be the set of the *sinks*, that is, the nodes with out-degree 0. The *underlying undirected simple graph* of G is the undirected simple graph $\langle G\rangle$ with node set $V(G)$ which contains an edge $\{v, w\}$ whenever there is an arc between v and w. We call G *tree-like* (*path-like*) if $\langle G\rangle$ is a tree (path). A *rooted tree* is obtained from an undirected tree by designating one node as its *root* and orienting all edges away from the root.

2.2 Rotor-Routing

Let G be a directed multigraph. For any node $v \in V^+(G)$, a *rotor order* at v is a cyclic permutation of $A^+(v)$. A *rotor order* for G is a permutation θ of $A(G)$ such that, for each $v \in V^+(G)$, the restriction of θ to $A^+(v)$ is a rotor order at v. The directed multigraph G combined with a rotor order θ is a *rotor graph*.

A *rotor configuration* of G is a mapping from $\rho : V^+(G) \to A(G)$ such that $\rho(v) \in A^+(v)$ for each $v \in V^+(G)$. Let $\mathcal{R}_G$ be the set of rotor configurations of G. A *particle configuration* of G is a mapping from $\sigma : V(G) \to \mathbb{Z}$. Let $\mathbb{Z}^{V(G)}$ be the set of particle configurations of G.

For node v, denote by $\boldsymbol{v} \in \mathbb{Z}^{V(G)}$ the function which is 1 at v, and 0 elsewhere. For example, $\sigma' = \sigma + 3\boldsymbol{v}$ means $\sigma'(v) = \sigma(v) + 3$ and $\sigma'(u) = \sigma(u)$ for $u \neq v$.

A *rotor-particle configuration* of G is a pair (ρ, σ) of a rotor configuration $\rho \in \mathcal{R}_G$ and a particle configuration $\sigma \in \mathbb{Z}^{V(G)}$. *Routing* at $v \in V^+(G)$ on (ρ, σ) is the operation of moving a particle from v along the arc $\rho(v)$ and then changing the rotor configuration at v to $\theta(\rho(v))$. The resulting rotor-particle configuration $(\rho', \sigma') = \mathrm{rout}^{\boldsymbol{v}}(\rho, \sigma)$ is then formally defined as

$$\rho'(u) = \begin{cases} \theta(\rho(u)) & u = v, \\ \rho(u) & u \neq v, \end{cases}$$

and $\sigma' = \sigma + \mathrm{head}(\rho(v)) - \mathrm{tail}(\rho(v))$. Routing decomposes into two operators. First, a *move* at v, denoted by $\phi(\rho; \boldsymbol{v}) = \mathrm{head}(\rho(v)) - \mathrm{tail}(\rho(v))$. Second, a *turn* at v, denoted by $\theta^{\boldsymbol{v}}$, where $\theta^{\boldsymbol{v}}(a) = \theta(a)$ for each $a \in A^+(v)$ and $\theta^{\boldsymbol{v}}(a) = a$ for each $a \notin A^+(v)$. We see that

$$\mathrm{rout}^{\boldsymbol{v}}(\rho, \sigma) = (\theta^{\boldsymbol{v}} \circ \rho, \sigma + \phi(\rho; \boldsymbol{v})) = (\rho', \sigma') \ .$$

This kind of routing is referred to as *move-then-turn*, as opposed to *turn-then-move*. It is easy to see that both kinds of routing are isomorphic by advancing or reverting the rotor configuration. We will use this isomorphism in the proof of Proposition 5.

We define $\theta^{-\boldsymbol{v}} = (\theta^{\boldsymbol{v}})^{-1}$ and $\phi(\rho; -\boldsymbol{v}) = -\phi(\theta^{-\boldsymbol{v}} \circ \rho; \boldsymbol{v})$; the *inverse* of $\mathrm{rout}^{\boldsymbol{v}}$ is defined as

$$\mathrm{rout}^{-\boldsymbol{v}}(\rho', \sigma') = (\theta^{-\boldsymbol{v}} \circ \rho', \sigma' + \phi(\rho'; -\boldsymbol{v})) = (\rho, \sigma) \ .$$

Given distinct nodes $v \neq u$, observe that $\mathrm{rout}^{\boldsymbol{v}}$ and $\mathrm{rout}^{\boldsymbol{u}}$ commute, as the move and turn of v depend on and affect ρ only at component v. Hence, compositions of the routing operators are characterized by the number of times each node is routed.

A *routing vector* (or simply a *routing*) of G is a mapping from $r : V^+(G) \to \mathbb{Z}$. We denote by rout^r the operator resulting from composing all $(\mathrm{rout}^{\boldsymbol{v}})^{r(v)}$ for $v \in V^+(G)$ in arbitrary order. Similarly, we let θ^r denote the composition of all $(\theta^{\boldsymbol{v}})^{r(v)}$ for $v \in V^+(G)$. Extending the move operator $\phi(\rho; \boldsymbol{v})$ is more involved. We do so by interpreting ϕ to be the *displacement* of particles when routing. As such, for $k \in \mathbb{N}$, we define $\phi(\rho; k\boldsymbol{v}) = \sum_{i=0}^{k-1} \phi(\theta^{i\boldsymbol{v}} \circ \rho; \boldsymbol{v})$ and $\phi(\rho; -k\boldsymbol{v}) = -\sum_{i=1}^{k} \phi(\theta^{-i\boldsymbol{v}} \circ \rho; \boldsymbol{v})$. Then we define $\phi(\rho; r) = \sum_{v \in V^+(G)} \phi(\rho; r(v)\boldsymbol{v})$. It is now straightforward to see that

$$\mathrm{rout}^r(\rho, \sigma) = (\theta^r \circ \rho, \sigma + \phi(\rho; r)) \ .$$

For the sake of convenience, we extend these notations to sinks. Each routing $r \in \mathbb{Z}^{V^+(G)}$ is identified with the mapping in $\mathbb{Z}^{V(G)}$ such that $r(s) = 0$ for all $s \in S_0$. Furthermore, routing sinks has no effect, i.e., $\theta^{\boldsymbol{s}} = \mathrm{id}$ and $\phi(\rho; \boldsymbol{s}) = \mathbf{0}$.

2.3 Arrival and Rotor-Routing Games

In the Arrival game we start with a rotor-particle configuration $(\rho, \boldsymbol{u})$ where $u \in V^+(G)$ is the initial location of the particle. We then repeatedly route the node on which the particle is located. This makes the particle walk through the rotor graph until it ends up on a sink, at which point the game terminates. To ensure termination, throughout this article we require G to be *stopping*, meaning that every node has a directed path to some sink.

We now introduce the *rotor-routing game*, which generalizes Arrival. We say that routing node v is *legal* on (ρ, σ) if $\sigma(v) > 0$. A *routing sequence* $(v_0, \dots, v_{k-1}) \in (V^+(G))^k$ is *legal* on (ρ_0, σ_0) if each routing of v_i is legal on (ρ_i, σ_i) where $(\rho_{i+1}, \sigma_{i+1}) = \mathrm{rout}^{\boldsymbol{v_i}}(\rho_i, \sigma_i)$. Finally, a non-negative routing r is *legal* if there is a *corresponding* legal routing sequence $(v_0, \dots, v_{k-1})$, which means that $r(v) = \left|\{i \in \{0, \dots, k-1\} \mid v_i = v\}\right|$ for each $v \in V^+(G)$.

Such a legal routing r is *maximal* on (ρ, σ) if $\sigma' = \sigma + \phi(\rho; r)$ is non-positive on $V^+(G)$. That is, a routing r is maximal if no node can be legally routed after routing r.

In the rotor-routing game, at each step an arbitrary legal node is routed until all particles are on sinks. It is easy to see that in Arrival, the legal routing sequence is unique, but in rotor-routing at any point there may be multiple nodes that can be legally routed.

An important consequence of routing a node behaving independent of the configuration of other nodes is that every legal routing sequence eventually terminates with the same corresponding maximal legal routing.

▶ **Lemma 3** ([9, Lemma 3.9]). *For all $(\rho, \sigma) \in \mathcal{R}_G \times \mathbb{Z}^{V(G)}$ with $\sigma \geq \mathbf{0}$, there is a unique maximal legal routing r on (ρ, σ).*

It follows that the maximal legal routing is an upper bound (component-wise) of all legal routings. We denote by $(\rho', \sigma') = \mathrm{rout}_L^\infty(\rho, \sigma)$ the rotor-particle configuration resulting from the unique maximal legal routing on (ρ, σ).

We are now ready to formally define the problems discussed in this paper through the notion of routings.

▶ **Problem** (ARRIVAL [3])**.** *Given* $(\rho, \boldsymbol{u}) \in \mathcal{R}_G \times \mathbb{Z}^{V(G)}$*, compute* σ' *for* $(\rho', \sigma') = \mathrm{rout}_L^\infty(\rho, \boldsymbol{u})$*.*

The following generalization of ARRIVAL was introduced by Hoang [8]:

▶ **Problem** (LEGAL MULTI-ARRIVAL [8])**.** *Given* $(\rho, \sigma) \in \mathcal{R}_G \times \mathbb{Z}^{V(G)}$ *with* $\sigma \geq \mathbf{0}$*, compute* σ' *for* $(\rho', \sigma') = \mathrm{rout}_L^\infty(\rho, \sigma)$*.*

An even more general scenario was introduced by Auger et al. [1]. Namely, if we do not require routings to be legal, for arbitrary σ any routing can be extended such that $\sigma'(v) = 0$ for all $v \in V^+(G)$. Such a rotor-particle configuration is called *fully routed*. We denote by $\mathrm{rout}^\infty(\rho, \sigma)$ the set of fully routed rotor-particle configurations obtainable by routing (ρ, σ).

▶ **Proposition 4** ([1, Theorem 2])**.** *For all* $(\rho, \sigma) \in \mathcal{R}_G \times \mathbb{Z}^{V(G)}$ *there is a routing* r *such that with* $(\rho', \sigma') = \mathrm{rout}^r(\rho, \sigma)$ *one has* $\sigma'(v) = 0$ *on all* $v \in V^+(G)$*. Furthermore, for any two such routings* r_1 *and* r_2*, resulting in* (ρ^1, σ^1) *and* (ρ^2, σ^2)*, respectively, we have that* $\sigma^1 = \sigma^2$*.*

This leads to the following problem, which is our main concern in this paper:

▶ **Problem** (MULTI-ARRIVAL [1])**.** *Given* $(\rho, \sigma) \in \mathcal{R}_G \times \mathbb{Z}^{V(G)}$*, compute* σ' *for arbitrary* $(\rho', \sigma') \in \mathrm{rout}^\infty(\rho, \sigma)$*.*

For $\sigma \geq \mathbf{0}$, after applying the maximal legal routing it holds that $\sigma'(v) = 0$ for $v \in V^+(G)$. It follows that $\mathrm{rout}_L^\infty(\rho, \sigma) \in \mathrm{rout}^\infty(\rho, \sigma)$ and that LEGAL MULTI-ARRIVAL reduces to MULTI-ARRIVAL. Conversely, finding full routings reduces to finding a maximal legal routing in G and in $G^{(-1)}$, where $G^{(-1)}$ is obtained from G by reversing the rotor order i.e. using θ^{-1}.

▶ **Proposition 5.** *Let* $(\rho, \sigma) \in \mathcal{R}_G \times \mathbb{Z}^{V(G)}$ *and* $\sigma = \sigma_1 - \sigma_2$ *such that* $\sigma_1, \sigma_2 \geq \mathbf{0}$*. Then given the maximal legal routing* r_1 *on* (ρ, σ_1) *in* G *and the maximal legal routing* r_2 *on* $(\theta^{r_1 - \mathbf{1}} \circ \rho, \sigma_2)$ *in* $G^{(-1)}$*, we have that* $r = r_1 - r_2$ *is a full routing on* (ρ, σ) *in* G*.*

Proof. We define $g \colon \mathcal{R}_G \times \mathbb{Z}^{V(G)} \to \mathcal{R}_G \times \mathbb{Z}^{V(G)}$ by $g(\rho, \sigma) = (\theta^{-1} \circ \rho, -\sigma)$. The function g transforms rotor-particle configurations such that negative move-then-turn routings in G correspond to positive turn-then-move routings in $G^{(-1)}$. We let $\theta' = \theta^{-1}$ and ϕ' denote the move and displacement operators in $G^{(-1)}$.

▷ Claim. It holds that $\mathrm{rout}_G^r(\rho, \sigma) = g^{-1}\big(\mathrm{rout}_{G^{(-1)}}^{-r}(g(\rho, \sigma))\big)$.

Proof. We show that $\mathrm{rout}_G^{\boldsymbol{v}}(\rho, \sigma) = g^{-1}\big(\mathrm{rout}_{G^{(-1)}}^{-\boldsymbol{v}}(g(\rho, \sigma))\big)$, from which the claim follows. Recall that $\mathrm{rout}_{G^{(-1)}}^{-\boldsymbol{v}}(\rho, \sigma) = \big(\theta'^{-\boldsymbol{v}} \circ \rho, \sigma - \phi(\theta'^{-\boldsymbol{v}} \circ \rho; \boldsymbol{v})\big)$. First, for the turn operator we see that $\theta'^{-\boldsymbol{v}} \circ \theta^{-1} = \theta^{-1} \circ \theta^{\boldsymbol{v}}$. Then, for the move operator we derive that

$$-\phi(\theta'^{-\boldsymbol{v}} \circ \theta^{-1} \circ \rho; \boldsymbol{v}) = -\phi(\rho; \boldsymbol{v}) \ .$$

Consequently, it holds that

$$\begin{aligned} g^{-1}\big(\mathrm{rout}_{G^{(-1)}}^{-\boldsymbol{v}}(g(\rho, \sigma))\big) &= g^{-1} \circ \mathrm{rout}_{G^{(-1)}}^{-\boldsymbol{v}}(\theta^{-1} \circ \rho, -\sigma) \\ &= g^{-1} \circ \big(\theta^{-1} \circ \theta^{\boldsymbol{v}} \circ \rho, -\sigma - \phi(\rho; \boldsymbol{v})\big) \\ &= \big(\theta^{\boldsymbol{v}} \circ \rho, \sigma + \phi(\rho; \boldsymbol{v})\big) = \mathrm{rout}_G^{\boldsymbol{v}}(\rho, \sigma) \ . \end{aligned}$$

This proves the claim. ◁

We next find that $\sigma + \phi(\rho; r_1 - r_2) = (\sigma_1 + \phi(\rho; r_1)) - (\sigma_2 + \phi'(\theta^{r_1 - \mathbf{1}} \circ \rho; r_2))$. Since r_1 is maximal on (ρ, σ_1) with $\sigma_1 \geq \mathbf{0}$ in G, it follows that $\sigma_1 + \phi(\rho; r_1)$ is 0 on $V^+(G)$. Similarly, as r_2 is maximal on $(\theta^{r_1 - \mathbf{1}} \circ \rho, \sigma_2)$, with $\sigma_2 \geq \mathbf{0}$ in $G^{(-1)}$ we have that $\sigma_2 + \phi'(\theta^{r_1 - \mathbf{1}} \circ \rho; r_2)$ is also 0 on $V^+(G)$. It follows that r is a full routing. ◀

Our approach finds maximal legal routings which enables us to solve both Legal Multi-Arrival and Multi-Arrival on tree-like rotor graphs.

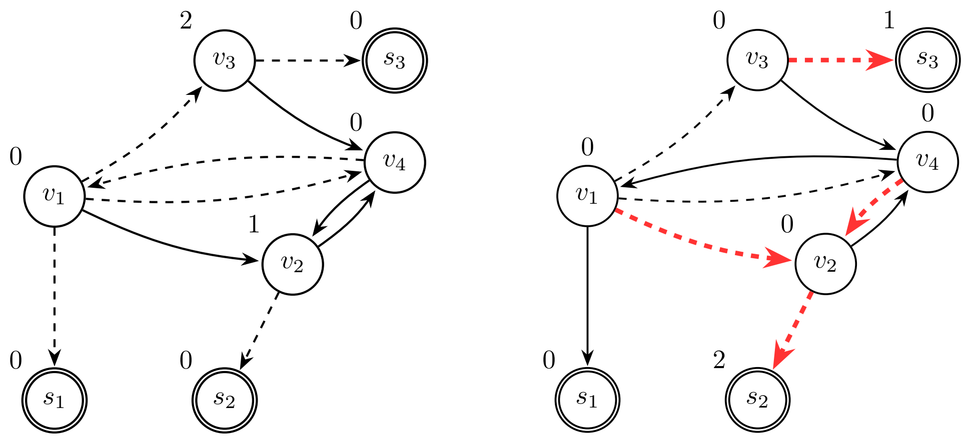
3 Routing Decompositions

In this section we show how to decompose maximal legal routings into legal routings whose most recently used arcs are directed towards sinks. We shall always denote the maximal legal routing by $\hat{r}$, assuming some (ρ, σ) to be fixed.

3.1 Destination Graphs

Given a rotor configuration ρ and routing $r \geq \mathbf{0}$, the *destination graph* $G[\rho; r]$ is a subgraph of G whose nodes consist of all those that either sent or received particles during routing r. For each node $v \in \text{supp}(r)$, that is $r(v) \neq 0$, the destination graph $G[\rho; r]$ contains only the outgoing arc that the last particle traversing v was sent over when routing r. That is, $G[\rho; r]$ is the subgraph of G induced by the arcs $\{\theta^{-1}(\rho'(v)) \mid v \in \text{supp}(r)\}$, where $\rho' = \theta^r \circ \rho$ is the rotor configuration after routing r. Each node $v \in \text{supp}(r)$ has one outgoing arc in $G[\rho; r]$ and every other node in $G[\rho; r]$ has no outgoing arcs.

Furthermore, the destination graph $G[\rho; \hat{r}]$ of the maximal legal routing is acyclic and the connected components are directed subtrees rooted at sinks. Figure 1 depicts a rotor graph with the rotor order being clockwise, and a corresponding destination graph.



(a) On (ρ, σ). **(b)** On (ρ', σ').

Figure 1 (a) A rotor graph G with a rotor-particle configuration (ρ, σ), where solid lines represent ρ, labels on the nodes represent the particle distribution $\sigma = 1\boldsymbol{v}_2 + 2\boldsymbol{v}_3$. A possible maximal legal routing sequence is $v_2 \to v_4 \to v_2 \to s_2$, $v_3 \to v_4 \to v_1 \to v_2 \to v_4 \to v_2 \to s_2$ and $v_3 \to s_3$, with corresponding maximal legal routing $\hat{r} = 1\boldsymbol{v}_1 + 4\boldsymbol{v}_2 + 2\boldsymbol{v}_3 + 3\boldsymbol{v}_4$. (b) Thick red lines are the arcs in the destination graph $G[\rho; \hat{r}]$ for the maximal legal routing $\hat{r}$. Note that $G[\rho; \hat{r}]$ does not contain s_1.

3.2 Compensated Routings

Any legal routing r must satisfy the condition that, for each node v, the number of particles sent out of v during the routing process must not exceed the number of particles initially present at v and those sent into v during the routing. This establishes one of the main constraints for a routing to be legal. We call such a routing "compensated".

Formally, we say that a routing r is *compensated* at v with respect to (ρ, σ) if

$$\sigma(v) + \sum_{u \in N^-(v)} \phi_v^u(\rho; r(u)) \geq \sum_{w \in N^+(v)} \phi_w^v(\rho; r(v)) = r(v),$$

where $\phi_w^v(\rho; j)$ denotes the number of particles sent to w when routing j particles out of v.

A routing is called *compensated* if it is compensated at each $v \in V^+(G)$. Equivalently, r is compensated if and only if $\sigma + \phi(\rho; r) \geq \mathbf{0}$. Note that compensated routings are not necessarily legal.

The following proposition, which is a special case of a theorem by Tóthmérész [11, Theorem 3.4], provides a characterization of compensated routings that are legal. For sake of completeness, we include a simplified proof.

▶ **Proposition 6.** *Given a routing $r \geq \mathbf{0}$ and rotor-particle configuration (ρ, σ) with $\sigma \geq \mathbf{0}$. Let $(\rho', \sigma') = \mathrm{rout}^r(\rho, \sigma)$. The routing r is legal on (ρ, σ) if and only if, r is compensated on (ρ, σ) and each cycle in $G[\rho; r]$ contains some node v such that $\sigma'(v) > 0$. In particular, if $G[\rho; r]$ is acyclic, legality and compensation are equivalent.*

Proof. Suppose that r is legal. So it is compensated. If $G[\rho; r]$ contains a cycle C, then given some corresponding legal routing sequence, there is some node $v \in V(C)$ that is routed last in C. Let $w \in V(C)$ denote the successor of v in C. Then a particle is sent from v to w, but w is not routed afterwards. Hence, as the particle distribution is non-negative during the legal routing sequence, we have that $\sigma'(w) > 0$.

Conversely, assume that r is compensated on (ρ, σ) and each cycle in $G[\rho; r]$ contains some node v such that $\sigma'(v) > 0$. Let $r^* \leq r$ be a legal routing with $(\rho^*, \sigma^*) = \mathrm{rout}^{r^*}(\rho, \sigma)$ obtained by routing until for each node v either $\sigma^*(v) = 0$ or $r^*(v) = r(v)$.

Since r^* is legal, it suffices to show that $r^* = r$. Assume, for the sake of contradiction, that $r^* \neq r$. We then consider the set $B = \{v \in V^+(G) \mid r^*(v) < r(v)\}$ of nodes that still need to be routed (if any).

It follows that $\sigma^*(v) = 0$ for each $v \in B$. In the remaining routing $r^* - r$, each particle sent out from a node in B must be replaced afterward; otherwise, we would end up with a node $v \in B$ such that $\sigma'(v) < 0$, which is impossible since r is compensated. This implies that during the routing $r^* - r$, no particle from B is sent outside of B; otherwise, that particle could not be replaced in B as no node outside of B is routed in $r^* - r$. It follows that in the subgraph H of $G[\rho; r]$ induced by B, the nodes have out-degree of 1, implying that H contains a cycle. This contradicts the assumption that every cycle in $G[\rho; r]$ contains a node v such that $\sigma'(v) > 0$. ◀

3.3 s-Directed Routings

The compensated routings r are those for which each $v \in V^+(G)$ satisfies an inequality that relates $r(v)$ to $r(u)$ for all in-neighbours u of v. Enforcing this inequality is much simpler than enforcing the stricter property of legality. If restricted to routings with acyclic destination graph, these properties coincide. For finding the maximal legal routing $\hat{r}$, we may restrict the search to only those routings. But the destination graph being acyclic is also a non-trivial property. Instead, we consider an even more restricted case in which the destination graph is oriented towards some fixed sink s. In tree-like rotor graphs, this implies a unique destination graph (up to parallel arcs), which can be enforced by an appropriate parametrization of routings. But first, we need to show that these restricted routings still allow us to recover $\hat{r}$.

Given a rotor configuration ρ and sink s, we say that node v is *s-directed* in a routing r if $v \in \mathrm{supp}(r)$ and there is a path in $G[\rho; r]$ from v to s. Routing r is s-directed if every $v \in \mathrm{supp}(r)$ is s-directed in r. Our aim, to be established in Theorem 9, is to decompose $\hat{r}$ into a set of s-directed routings r_s for each sink s. The idea is that each such r_s need only match the number of particles sent into s by $\hat{r}$. We denote by $\phi_w(\rho; r)$ the number of particles sent to w when routing r.

► **Lemma 7.** *Let (ρ, σ) be a rotor-particle configuration with $\sigma \geq \mathbf{0}$ and $b \geq \mathbf{0}$ be a legal routing. Then, for each sink s, there is an s-directed legal routing $r \geq \mathbf{0}$ such that $r \leq b$ and $\phi_s(\rho; r) = \phi_s(\rho; b)$.*

Proof. We construct a series of legal routings $b = r_1 \geq r_2 \geq \cdots$ with $A_1 \subset A_2 \subset \cdots$ denoting the s-directed nodes of the respective routing, such that each r_{i+1} equals r_i on A_i and each $v \in A_i$ has the same in-flow in r_{i+1} as in r_i.

Suppose that we constructed r_i and that there is flow from $V^+(G) \setminus A_i$ to A_i when routing r_i. Consider an arbitrary legal routing sequence $(v_0, \ldots, v_{m-1})$ corresponding to r_i with $(\rho_0, \sigma_0) = (\rho, \sigma)$ and $(\rho_{\ell+1}, \sigma_{\ell+1}) = \mathrm{rout}^{\boldsymbol{v_\ell}}(\rho_\ell, \sigma_\ell)$.

Let j be maximal such that $v_j \notin A_i$ and $\mathrm{head}(\rho_j(v_j)) \in A_i$. That is, the j^{th} routing step is the last time a particle is sent into A_i from $V^+(G) \setminus A_i$. Now, we remove each routing step $\ell > j$ with $v_\ell \notin A_i$. Let r_{i+1} be the corresponding routing vector. Then r_{i+1} must be legal since for $\ell > j$ we have $v_\ell \in A_i$ and v_ℓ has in the ℓ-th routing step the same number of particles as before the removals. By construction r_{i+1} satisfies the required conditions and $A_i \cup \{v_j\} \subseteq A_{i+1}$ as v_j sent its last particle towards A_i in r_{i+1}.

Otherwise, suppose there is no flow from $V^+(G) \setminus A_i$ to A_i when routing r_i. We set r to be r_i on A_i, and zero on $V^+(G) \setminus A_i$. Then r is an s-directed legal routing with $r \leq r_1 = b$. Since $s \in A_1 \subseteq A_2 \subseteq \ldots$ we have $\phi_s(\rho; r_1) = \phi_s(\rho; r_2) = \cdots = \phi_s(\rho; r)$. This procedure is finite as it terminates once $\mathrm{supp}(b) \subseteq A_i$ and each step increases the size of A_i. ◄

Next, we show that in the maximal legal routing $\hat{r}$ the s-directed nodes already route only as much as is required to achieve the flow towards s. We therefore derive that a corresponding s-directed routing matches $\hat{r}$ on those nodes.

► **Lemma 8.** *Let (ρ, σ) be a rotor-particle configuration with $\sigma \geq \mathbf{0}$. If $r \geq \mathbf{0}$ is an s-directed legal routing such that $\phi_s(\rho; r) = \phi_s(\rho; \hat{r})$, then $r(v) = \hat{r}(v)$ for each s-directed node of $\hat{r}$.*

Proof. Suppose u is s-directed in $\hat{r}$ such that $r(u) \neq \hat{r}(u)$. We denote by v the successor on the path from u to s in $G[\rho; \hat{r}]$. As $\hat{r}$ is maximal and r is legal, it must be that $r(u) < \hat{r}(u)$. Since the last particle sent from u in $\hat{r}$ is towards v we have $\phi_v^u(\rho; r(u)) < \phi_v^u(\rho; \hat{r}(u))$. In particular, $\phi_v(\rho; r) < \phi_v(\rho; \hat{r})$. If $v \neq s$, then as $\hat{r}$ routes v for each received particle and r receives less particles on v, it follows that $r(v) < \hat{r}(v)$. Now, v is also s-directed in $\hat{r}$ with $r(v) \neq \hat{r}(v)$. We repeat the argument along the path from u to s in $G[\rho; \hat{r}]$ until we end up with $v = s$. Then $\phi_s(\rho; r) < \phi_s(\rho; \hat{r})$ contradicts the assumption that $\phi_s(\rho; r) = \phi_s(\rho; \hat{r})$. ◄

► **Theorem 9.** *Let (ρ, σ) be a rotor-particle configuration with $\sigma \geq \mathbf{0}$. For each $s \in S_0$, let r_s be an optimal solution of the following optimization problem:*

$$\text{maximize} \quad \phi_s(\rho; r) \quad \text{subject to} \quad r \text{ is an } s\text{-directed compensated routing.}$$

Then $\hat{r}$ is given by $\hat{r}(v) = \max\{r_s(v) \mid s \in S_0\}$ for all $v \in V^+(G)$.

Proof. Let $s \in S_0$. As r_s is s-directed, it has an acyclic destination graph. Since it is also compensated, it follows from Proposition 6 that r_s is legal. In particular, as $\hat{r}$ is maximal we have $r_s \leq \hat{r}$ and $\phi_s(\rho; r_s) \leq \phi_s(\rho; \hat{r})$. Now, as shown by Lemma 7 there is some legal s-directed routing r'_s such that $\phi_s(\rho; r'_s) = \phi_s(\rho; \hat{r})$. It follows from r_s maximizing $\phi_s(\rho; r)$ that $\phi_s(\rho; r_s) = \phi_s(\rho; \hat{r})$.

For each $v \notin \text{supp}(\hat{r})$ the claim holds as $0 \leq r_s(v) \leq \hat{r}(v) = 0$ for each $s \in S_0$. Let $v \in \text{supp}(\hat{r})$. Then as $G[\rho; \hat{r}]$ is acyclic and each node in $G[\rho; \hat{r}]$ has out-degree at most one, there is some u that the unique path starting at v ends on. If $u \notin S_0$, then $u \notin \text{supp}(\hat{r})$. But then u receives a particle in $\hat{r}$ but is not routed, which contradicts the maximality of $\hat{r}$. Thus, $u \in S_0$ and v is u-directed in $\hat{r}$. Consequently, due to Lemma 8 we find that $\hat{r}(v) = r_u(v) = \max\{r_s(v) \mid s \in S_0\}$. ◀

4 Algorithms for Multi-Arrival on Tree-Like Rotor Graphs

In the following, let T be a tree-like rotor graph. Henceforth assume, without loss of generality, that the induced subgraph of T on $V^+(T)$ is strongly connected, and each sink to have exactly one in-neighbour. Indeed, if T were not strongly connected, then we can solve Multi-Arrival in some ordering of the strongly connected components as each component has no particles flowing to the prior. Furthermore, if some sink had multiple in-neighbours, we split it into one sink per in-neighbor.

4.1 Relative Rotor Subgraphs

With the above assumptions on T, each $v \in V^+(T)$ has a unique successor on its paths to s in T. We denote by $N_s^+(v)$ the *successor* on the paths from v to s in T, and by $N_s^-(v)$ the set of *predecessors* whose successor is v on their respective paths towards s. When a routing is s-directed, its destination graph can be seen to be an s-rooted directed subtree of T.

Throughout this section, we fix a sink $s \in S_0$ and a rotor-particle configuration (ρ, σ) with $\sigma \geq \mathbf{0}$. For the sake of convenience, we leave this dependence on s and (ρ, σ) implicit in the introduced notation. Further, we simplify the notation and let $\phi(r)$ denote $\phi(\rho; r)$, and similarly extend this simplification to ϕ_w^v and ϕ_v.

We proceed to show that s-directed routings can be characterized by their restrictions to nested rotor subgraphs. For any node $v \in V^+(T)$ with $w = N_s^+(v)$ let T_v be the rotor graph obtained from T as follows. We remove the arcs between v and w, and then, remove the nodes not reachable from v. Finally, we add w again with only the original arcs from v to w (but not the reverse arcs). In T_v, node w is thus a sink. See Figure 2 for an example.

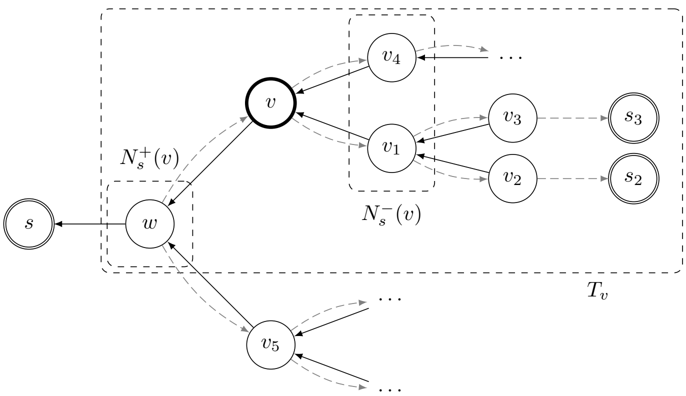


Figure 2 Example of a relative subtree T_v. Grey dashed lines show the arcs oriented away from s. Note that, e.g., the rotor graph T_{v_2} contains v_1 as a sink, which therefore has no arcs from v_1 to v_2.

For any integers $x, y \in \mathbb{N}$, we denote by $H_v(x, y)$ the set of w-directed compensated routings r in T_v on rotor-particle configuration $(\rho, \sigma + y\boldsymbol{v})$ such that $\phi^v_w(r(v)) = x$.

Furthermore, we define $h_v(y) = \max\{x \in \mathbb{N} \mid H_v(x, y) \neq \emptyset\}$.

Note that for a given y, from Lemma 7 it follows that $h_v(y)$ is the flow from v to w that is legally achievable in T_v when dispersing an additional y particles on v.

▶ **Lemma 10.** *Given $v \in V^+(T)$ and $x, y \in \mathbb{N}$ such that $H_v(x, y) \neq \emptyset$, we have $H_v(x', y) \neq \emptyset$ for any $0 \leq x' \leq x$. In particular, $H_v(x, y) \neq \emptyset$ if and only if $0 \leq x \leq h_v(y)$.*

Proof. Let $r \in H_v(x, y)$ and let b be some legal routing such that $x' = \phi_w(\rho; b) \leq \phi_w(\rho; r) = x$ in T_v, where $w = N^+_s(v)$. Note that such a b exists as we may truncate a legal routing sequence of r up to the x'-th routing towards w. Then, using Lemma 7 with b on T_v, we obtain an w-directed compensated routing r' with $x' = \phi_w(\rho; r')$. Hence, $r' \in H_v(x', y)$. As the zero routing is always compensated and s-directed, the second claim follows as well. ◀

Let $w = N^+_s(v)$ and $u \in N^-_s(v)$. We define the function $q_v \colon \mathbb{N} \to \mathbb{N}$ as the minimum number of routings required to send x particles from v to w. Formally, $q_v(x) = \min\{k \in \mathbb{N} \mid \phi^v_w(\rho; k) = x\}$. It is also convenient to specify the notation $q^v_u(x) = \phi^v_u(q_v(x))$, which is the number of particles that v sends to u as a by-product.

The following crucial lemma will enable us to recursively construct routings in $H_v(x, y)$.

▶ **Lemma 11.** *Let $v \in V^+(T)$ and $w = N^+_s(v)$. Consider a routing $r \geq \mathbf{0}$ on T_v and for $u \in N^-_s(v)$ let r_u denote the restriction of r on $V^+(T_u)$. Then, $H_v(0, y) = \{\mathbf{0}\}$, and for $x > 0$,*

$$r \in H_v(x, y)$$

if and only if

$$r(v) = q_v(x), \tag{1}$$

and there exist non-negative integers x_u, for $u \in N^-_s(v)$, such that

$$\sum_{u \in N^-_s(v)} x_u \geq r(v) - y - \sigma(v), \tag{2}$$

$$r_u \in H_u\big(x_u, \phi^v_u(r(v))\big), \text{ for each } u \in N^-_s(v) \quad . \tag{3}$$

In particular, $H_v(x, y) \neq \emptyset$ if and only if

$$\sum_{u \in N^-_s(v)} h_u(q^v_u(x)) \geq q_v(x) - y - \sigma(v) \quad . \tag{4}$$

Proof. Let $r \in H_v(0, y)$. Then $r(v) = 0$, as $\phi^v_w(r(v)) = 0$ and r is w-directed. It follows that v is not in $T_v[\rho; r]$, and since every path from $V^+(T_v)$ to w in T_v is through v, it must be that $\mathrm{supp}(r) = \emptyset$.

Now, for the case $x > 0$, assume that $r \in H_v(x, y)$ and therefore $r(v) > 0$. Thus, the last routing of r at v should be towards w, which is equivalent to (1).

Set $x_u = \phi^u_v(r_u(u)) = \phi^u_v(r(u))$. Then we see that (2) is a rearrangement of r being compensated at v. For (3), consider $u \in N^-_s(v)$. As the nodes of T_u are w-directed in r and their paths towards w in $T_v[\rho; r]$ go through v, it follows that r_u is v-directed in T_u. It remains to show that r_u being compensated in T_v is equivalent to r_u being compensated in T_u. With $y_u = \phi_u^v(r(v))$, note that r_u needs to be compensated on $(\rho, \sigma + y_u \boldsymbol{u})$ in T_u. Clearly, the only non-trivial case is the compensation at u, which follows from the in-flow being equal in both cases:

$$\phi_u(r) = \phi_u(r_u) + \phi_u^v(r(v)) + \sigma(u) = \phi_u(r_u) + \sigma(u) + y_u \ .$$

Consequently, $r_u \in H_u(x_u, y_u)$.

Now suppose that $r \geq \mathbf{0}$ is a routing that satisfies (1)–(3). Notably, as $x > 0$ we have $q_v(x) > 0$. As mentioned above, (1) is equivalent to v being w-directed in r. From (3) it follows that $x_u = \phi_v^u(r(u))$, which together with (2) implies that r is compensated at v. Also, r is w-directed and compensated on $V^+(T_v) \setminus \{v\}$, because by (3), each restriction r_u is v-directed and compensated. ◀

The following properties of the functions h_v are needed in our argument.

▶ **Lemma 12.** *For any $v \in V^+(T)$, define $f_v(x) = q_v(x) - \sigma(v) - \sum_{u \in N_s^-(v)} h_u(q_u^v(x))$ for all x. Then the following properties hold for the functions f_v and h_v:*
- **(i)** $h_v(y) = \max\{x \in \mathbb{N} \mid f_v(x) \leq y\}$,
- **(ii)** $h_v(y) \leq h_v(y+1) \leq h_v(y) + 1$, *for* $y \in \mathbb{N}$,
- **(iii)** $f_v(x+1) \geq f_v(x) + 1$, *for* $x \in \mathbb{N}$.

Proof. (i) We know that $h_v(y)$ equals the largest x with $H_v(x, y) \neq \emptyset$, which, by Lemma 11, is the largest x satisfying (4), that is, the largest x such that $f_v(x) \leq y$.

(ii) Let $\hat{r}_y$ denote the maximal legal routing in T_v on $(\rho, \sigma + y\boldsymbol{v})$. Then, by definition of $h_v(y)$ and Lemma 7, we see that $h_v(y) = \phi_w(\rho; \hat{r}_y)$. Thus, with $h_v(y+1) = \phi_s(\rho; \hat{r}_{y+1})$, we see that $h_v(y+1) = h_v(y) + 1$ if that additional particle ends up in w and $h_v(y+1) = h_v(y)$ otherwise.

(iii) Being integer-valued function, it suffices to show that f_v is strictly increasing.

Observe that $q_v(x+1) - q_v(x)$ is the number of arcs until and including the next that has head w. But $q_u^v(x+1) - q_u^v(x)$ is then the number of those arcs that have head u. Consequently, $q_v(x+1) - q_v(x) > \sum_{u \in N_s^-(v)} \big(q_u^v(x+1) - q_u^v(x)\big)$, where by (ii), the right-hand side is as large as $\sum_{u \in N_s^-(v)} \big(h(q_u^v(x+1)) - h(q_u^v(x))\big)$. It then follows that $f_v(x+1) > f_v(x)$. ◀

4.2 Multi-Arrival for Tree-Like Multigraphs

In this subsection, we prove the main result of the paper, Theorem 1. To do so, we first need to specify how a rotor graph is computationally represented.

Suppose we wanted to represent $A^+(v)$ and θ^v. We could take in a sequence of nodes $w_1, \ldots, w_k$ and let each index i identify an outgoing arc with head w_i. Arc with label i is taken to be before $i+1$ in θ^v with index $k+1$ wrapping around to 1. We can improve upon this by merging consecutive duplicates $w_i = w_{i+1}$ by associating with each w_i the number of times that entry is repeated. We denote by $\hat{A}$ the reduced arc set of the rotor graph obtained by removing arcs in A that vanish due to this encoding of consecutive parallel arcs. This encoding is called *succinct* [4, 11]. Hence, $|\hat{A}|$ is polynomial in the input size, while $|A|$ may be exponential in the input size.

For a rooted simple tree R, its *contracted height* $\mathsf{ch}(R)$ is defined as the minimum height over all trees which are obtained from R by contracting, for each node v of R, one of the arcs to its children.

▶ **Lemma 13.** *For any rooted simple tree R, it holds that $\mathsf{ch}(R) \leq \log(1 + \ell(R))$, where $\ell(R)$ is the number of nodes of R with at least two children.*

Proof. We proceed by induction on the height of R; recall that the height of a rooted tree is the maximum length of a path from its root to any of its other nodes. If the height of R is zero, the assertion holds trivially. Thus, assume now that the height of R is at least 1. Let $R_1, \dots, R_k$ be the connected components of R after removing its root, with $\mathsf{ch}(R_j) = \max\{\mathsf{ch}(R_i) \mid i = 1, \dots, k\}$. If there is a unique such j, then $\mathsf{ch}(R) = \mathsf{ch}(R_j)$ since the edge to that subtree can be contracted, and we are thus done by the induction hypothesis.

Otherwise, there is another such j, say j'. We may assume that $\ell(R_j) \leq \ell(R_{j'})$. Then

$$\mathsf{ch}(R) \leq 1 + \mathsf{ch}(R_j) \leq \log(2 + 2\ell(R_j)) \leq \log(1 + \ell(R)) \ . \qquad ◀$$

Let R'_u be the tree $\langle T[V^+] \rangle$ rooted at u. We define

$$k(T) = 1 + \max\{\mathsf{ch}(R'_u) \mid u \in N^-(S_0)\} \ .$$

By Lemma 13, $k(T) \leq 1 + \log(n_{>2}(\langle T \rangle) + 1)$, where $n_{>2}(\langle T \rangle)$ denotes the number of nodes in $\langle T \rangle$ that have more than two neighbors.

Notice that $k(T) \leq \kappa(T)$, where $\kappa(T)$ is the upper bound on the contracted heights of $\langle T \rangle$ as mentioned in introduction for the sake of stating an informal version of the main result.

We can thus now state our main result formally.

▶ **Theorem 1.** Multi-Arrival *can be solved on any tree-like rotor graph T with arc set A in time $\mathcal{O}\Big(|S_0||\hat{A}| \log^k(|A|)\Big)$, where $k = k(T) \leq 1 + \log\big(n_{>2}(\langle T \rangle) + 1\big)$. In particular,* Multi-Arrival *can be solved in time polynomial in $|A|$ if $k = \mathcal{O}\left(\frac{\log |A|}{\log(\log |A|)}\right)$. Furthermore, if $k = \mathcal{O}(1)$ (which includes path-like rotor graphs as then $k = 1$),* Multi-Arrival *can be solved in time polynomial in $|\hat{A}|$.*

Furthermore, in the procedure, a full routing for Multi-Arrival *is obtained.*

When k is fixed, we have that $\log^k(|A|) \leq f(k)|A|$ for some computable function f, and thus the run time of our algorithm would be at most $f(k)|A|^3$. Therefore, we conclude:

▶ **Corollary 14.** Multi-Arrival *on tree-like rotor graphs is fixed-parameter tractable with parameter k.*

One of the main ingredients of our approach is an affine approximation for the function $h_v(y)$, which is provided by the following lemma:

▶ **Lemma 15.** *For any $v \in V^+(T)$, there is an affine function $\overline{h}_v \colon \mathbb{R} \to \mathbb{R}$ such that*

$$|\overline{h}_v(y) - h_v(y)| \leq 2|A(T_v)|, \quad y \in \mathbb{N} \ . \tag{5}$$

A set of these functions $\{\overline{h}_v \mid v \in V^+(T)\}$ can be determined in time $\mathcal{O}(|\hat{A}|)$.

Proof. We first observe that for $x > 0$, it holds that

$$q_v\big(x + d(v,w)\big) = q_v(x) + d_+(v),$$
$$q_u^v\big(x + d(v,w)\big) = q_u^v(x) + d(v,u) \ .$$

These properties suggest the following approximations:

$$\overline{q}_v(x) = \big(d_+(v) - q_v(d(v,w))\big) + \frac{x d_+(v)}{d(v,w)} \quad \text{to approximate } q_v(x), \text{ and}$$
$$\overline{q}_u^v(x) = \big(d(v,u) - q_u^v(d(v,w))\big) + \frac{x d(v,u)}{d(v,w)} \quad \text{to approximate } q_u^v(x) \ .$$

Let $x = \ell d(v,w) + t \geq d(v,w)$ with $0 \leq t < d(v,w)$. Then

$$|q_v(x) - \overline{q}_v(x)| = \left| \big(q_v(d(v,w)+t) - q_v(d(v,w))\big) - \frac{td_+(v)}{d(v,w)} \right| < d_+(v), \tag{6}$$

$$|q_u^v(x) - \overline{q}_u^v(x)| = \left| \big(q_u^v(d(v,w)+t) - q_u^v(d(v,w))\big) - \frac{td(v,u)}{d(v,w)} \right| < d(v,w) \ . \tag{7}$$

Similarly, (6) and (7) hold for $x < d(v,w)$ as then $q_v(x)$ and $\overline{q}_v(x)$ lie within an interval of size $d_+(v)$, and $q_u^v(x)$ and $\overline{q}_u^v(x)$ lie within an interval of size $d(v,w)$. Now, define inductively

$$\overline{h}_v(y) = \max\{x \in \mathbb{R} \mid \overline{f}_v(x) \leq y\},$$
$$\overline{f}_v(x) = \overline{q}_v(x) - \sigma(v) - \sum_{u \in N_s^-(v)} \overline{h}_u\big(\overline{q}_u^v(x)\big) \ .$$

▷ Claim. For any $x, y, \delta \in \mathbb{R}$ with $\delta > 0$, it holds that

$$\overline{f}_v(x+\delta) \geq \overline{f}_v(x) + \delta, \tag{8}$$
$$\overline{h}_v(y+\delta) \leq \overline{h}_v(y) + \delta \ . \tag{9}$$

Proof. We first observe that for each node v, (9) follows from (8). Let $\overline{h}_v(y) = x$. So $\overline{f}_v(x) = y$. Then $\overline{f}_v(x+\delta) \geq \overline{f}_v(x) + \delta = y + \delta$. As $\overline{f}_v$ is strictly increasing, we must have $\overline{h}_v(y+\delta) \leq x + \delta = \overline{h}_v(y) + \delta$.

Now, we establish (8) by induction. We have that

$$\overline{f}_v(x+\delta) - \overline{f}_v(x) = \overline{q}_v(x+\delta) - \overline{q}_v(x) - \sum_{u \in N_s^-(v)} \Big(\overline{h}_u\big(\overline{q}_u^v(x+\delta)\big) - \overline{h}_u\big(\overline{q}_u^v(x)\big)\Big) \ .$$

If $N_s^-(v) = \emptyset$, then $\overline{f}_v(x+\delta) - \overline{f}_v(x) \geq \frac{\delta d_+(v)}{d(v,w)} \geq \delta$. Next, assume that $N_s^-(v) \neq \emptyset$ and (8) holds for all $u \in N_s^-(v)$. Then by the above argument, (9) also holds for $u \in N_s^-(v)$, implying that

$$\overline{h}_u\big(\overline{q}_u^v(x+\delta)\big) - \overline{h}_u\big(\overline{q}_u^v(x)\big) = \overline{h}_u\left(\overline{q}_u^v(x) + \frac{\delta d(v,u)}{d(v,w)}\right) - \overline{h}_u\big(\overline{q}_u^v(x)\big) \leq \frac{\delta d(v,u)}{d(v,w)} \ .$$

It follows that $\overline{f}_v(x+\delta) - \overline{f}_v(x) \geq \frac{\delta d_+(v)}{d(v,w)} - \sum_{u \in N_s^-(v)} \frac{\delta d(v,u)}{d(v,w)} = \delta$, as required. ◁

▷ Claim 16. If for all $x \in \mathbb{Z}$, $|\overline{f}_v(x) - f_v(x)| \leq L$, then $|\overline{h}_v(y) - h_v(y)| \leq L+1$ for all $y \in \mathbb{Z}$.

Proof. Suppose that $\overline{h}_v(y) = x$. Then $\overline{f}_v(\lfloor x \rfloor) \leq \overline{f}_v(x) \leq y$ and $\overline{f}_v(\lfloor x \rfloor + 1) > y$. Thus, $f_v(\lfloor x \rfloor + 1 + L) \geq f_v(\lfloor x \rfloor + 1) + L \geq \overline{f}_v(\lfloor x \rfloor + 1)$, that is $f_v(\lfloor x \rfloor + 1 + L) > y$. That means $h_v(y) \leq \lfloor x \rfloor + L$. On the other hand, $f_v(\lfloor x \rfloor - L) \leq f_v(\lfloor x \rfloor) - L \leq \overline{f}_v(\lfloor x \rfloor) \leq y$. So $h_v(y) \geq \lfloor x \rfloor - L$, and thus $\big|h_v(y) - \lfloor x \rfloor\big| \leq L$ which implies that $|h_v(y) - \overline{h}_v(y)| \leq L+1$. ◁

Now, we complete the proof by induction. If $N_s^-(v) = \emptyset$, then $\overline{h}_v(y) = \frac{d(v,w)}{d_+(v)}(y + \sigma(v))$ is an affine function that satisfies (5), because of (6) and Claim 16.

So assume that $N_s^-(v) \neq \emptyset$ and the assertion holds for $\overline{h}_u(y)$ for every $u \in N_s^-(v)$. The property of being an affine function follows directly from the definition of $\overline{h}_v(y)$.

For an integer x, we have:

$$\big|h_u\big(q_u^v(x)\big) - \overline{h}_u\big(\overline{q}_u^v(x)\big)\big| \leq \big|h_u\big(q_u^v(x)\big) - \overline{h}_u\big(q_u^v(x)\big)\big| + \big|\overline{h}_u\big(q_u^v(x)\big) - \overline{h}_u\big(\overline{q}_u^v(x)\big)\big| \ . \tag{10}$$

The induction hypothesis implies that

$$\left|h_u\big(q^v_u(x)\big) - \overline{h}_u\big(q^v_u(x)\big)\right| \le 2|A(T_u)| \ .$$

Combining (7) and (9), we see that

$$\left|\overline{h}_u\big(q^v_u(x)\big) - \overline{h}_u\big(\overline{q}^v_u(x)\big)\right| \le d(v,u) \ .$$

Thus, the right-hand side of (10) is at most $d(v,u) + 2|A(T_u)|$. It follows that for every $x \in \mathbb{Z}$, it holds that

$$|\overline{f}_v(x) - f_v(x)| \le d_+(v) + \sum_{u \in N^-_s(v)} \big(d(v,u) + 2|A(T_u)|\big) \ .$$

Then applying Claim 16, for every $y \in \mathbb{Z}$ we have that

$$|\overline{h}_v(y) - h_v(y)| \le 1 + d_+(v) + \sum_{u \in N^-_s(v)} \big(d(v,u) + 2|A(T_u)|\big) \le 2|A(T_v)| \ .$$

The inductive constructing of $\overline{h}_v$ for each $v \in V^+(T)$ can be done in time $\mathcal{O}(|\hat{A}^+(v)|)$. Therefore, the set $\{\overline{h}_v \mid v \in V^+(T)\}$ can be constructed in time $\mathcal{O}(|\hat{A}^+|)$. ◀

In following, we let $k_v = 1 + \mathsf{ch}(T'_v)$ where T'_v is the rooted simple tree $\langle T_v\rangle - (S_0 \setminus \{s\})$ rooted at w.

▶ **Lemma 17.** *Let t be the unique in-neighbour of s. Then the maximum $\phi_s(r)$ over every s-directed compensated routing r can be obtained in time $\mathcal{O}\big(|\hat{A}|\log^{k_t}(|A|)\big)$.*

Proof. We first determine the set of affine functions $\{\overline{h}_v \mid v \in V^+(T)\}$ in time $\mathcal{O}(|\hat{A}|)$ using Lemma 15. For given $x, y \in \mathbb{N}$, let $\mathcal{T}_v$ be the time complexity required to *query* $H_v(x,y)$, that is to return a routing from the set or to confirm that it is empty.

We will prove that

$$\mathcal{T}_v = \mathcal{O}\big(|\hat{A}(T_v)|\log^{k_v - 1}(|A(T_v)|)\big). \tag{11}$$

Once we established this, for a given x we can determine if there is some $r \in H_t(x,0)$, that is an s-directed compensated routing r_s such that $\phi_s(r_s) = x$, within time $\mathcal{T}_t$.

Moreover, by employing an exponential search using the approximation $\overline{h}_t(0)$, we only need to check the non-emptiness of $H_t(x,0)$ for $\mathcal{O}(\log|A|)$ distinct values of x. This process enables us to compute

$$h_t(0) = \max\{\phi_s(r) \mid s\text{-directed compensated routing } r\}$$

in time $\mathcal{O}(\mathcal{T}_t \log|A|) = \mathcal{O}\big(|\hat{A}|\log^k(|A|)\big)$, which completes the proof.

So it remains to prove (11). To do so, we first establish the following claim.

▷ Claim 18. For $v \in V^+(T)$ and any choice of $u^* \in N^-_s(v)$, it holds that

$$\mathcal{T}_v = \mathcal{O}\Big(|\hat{A}^+(v)| + \mathcal{T}_{u^*} + \sum_{u \in N^-_s(v)\setminus\{u^*\}} \mathcal{T}_u \log|A(T_u)|\Big) \ .$$

Proof. Given $x, y \in \mathbb{N}$, our objective is to either find a routing $r \in H_v(x,y)$ or confirm that the set is empty. We may assume $x > 0$ as otherwise we simply return $r = \mathbf{0}$.

For each $u \in N^-_s(v)$, let $y_u = q^v_u(x)$.

First, we determine $r(v) = q_v(x)$. Then, for each $u \in N_s^-(v) \setminus \{u^*\}$, we determine $\overline{h}_u(y_u)$.

Using the induced bound of that approximation on $h_u(y_u)$, we perform an exponential search to obtain $x_u = h_u(y_u)$ and $r_u \in H_u(x_u, y_u)$.

To perform this, due to Lemma 15, we require at most $1 + \log |A(T_u)|$ queries to H_u. As for x_{u^*}, we set

$$z = r(v) - \sigma(v) - y - \sum_{u \in N_s^-(v) \setminus \{u^*\}} x_u \ .$$

If $z \leq 0$, we let $x_{u^*} = 0$ and r_{u^*} be the zero routing which belongs to $H_{u^*}(0, y_{u^*})$. Otherwise, we let $x_{u^*} = z$ and query $r_{u^*} \in H_{u^*}(x_{u^*}, y_{u^*})$. If no such r_{u^*} exists, we terminate and assert $H_v(x, y) = \emptyset$.

Finally, we return r as given by $r(v)$ and whose restrictions to $V(T_u)$ is given by r_u for every $u \in N_s^-(v)$.

Now, the procedure takes the claimed time if $r(v)$ and $\{y_u \mid u \in N_s^-(v)\}$ can be calculated in time $\mathcal{O}(|\hat{A}^+(v)|)$. This can be done using a straightforward procedure employing two Euclidean divisions and two iterations through the encoding $\hat{A}^+(v)$.

As for correctness, if we failed to obtain r_{u^*}, we must have $x_{u^*} > h_{u^*}(y_{u^*})$. Hence:

$$0 > r(v) - \sigma(v) - y - \sum_{u \in N_s^-(v)} h_u(y_u),$$

which by Lemma 11 implies $H_v(x, y) = \emptyset$. Otherwise, r satisfies every condition of Lemma 11 and thus $r \in H_v(x, y)$. ◁

To complete the proof of (11) we now proceed by induction. If $N_s^-(v) = \emptyset$, we have $\mathcal{T}_v = \mathcal{O}(|\hat{A}^+(v)|)$ which satisfies the hypothesis as $k_v = 1$. Otherwise, by the induction hypothesis, for each $u \in N_s^-(v)$, we have that $\mathcal{T}_u = \mathcal{O}\big(|\hat{A}(T_u)| \log^{k_u - 1}(|A(T_u)|)\big)$.

Now, using Claim 18, we obtain

$$\mathcal{T}_v = \mathcal{O}\Big(|\hat{A}^+(v)| + |\hat{A}(T_{u^*})| \log^{k_{u^*}-1}(|A(T_{u^*})|) \ + \sum_{u \in N_s^-(v) \setminus \{u^*\}} |\hat{A}(T_u)| \log^{k_u}(|A(T_u)|)\Big) \ .$$

We choose $u^* \in N_s^-(v)$ such that $k_{u^*} = \max\{k_u \mid u \in N_s^-(v)\}$. Then

$$k_v = \max\{k_{u^*}\} \cup \{k_u + 1 \mid u^* \neq u \in N_s^-(v)\},$$

from which (11) follows. ◀

Proof of Theorem 1. By Proposition 5, it suffices to show that Legal Multi-Arrival can be solved in the claimed time.

Our algorithm for Theorem 1 finds, for every $s \in S_0$, an s-directed compensated routing r_s which maximizes $\phi_s(r)$. This by Theorem 9 allows us to obtain the maximal legal routing $\hat{r}$ given by $\hat{r}(v) = \max\{r_s(v) \mid s \in S_0\}$ for $v \in V^+(T)$.

Hence, for each $s \in S_0$ we need only show that r_s can be obtained in time $\mathcal{O}\big(|\hat{A}| \log^k(|A|)\big)$, which is established by Lemma 17.

If for some constant $C > 0$, we have $k \leq \frac{C \log |A|}{\log(\log |A|)}$, then it is seen that $\log^k(|A|) \leq |A|^C$, and thus the run time of the algorithm is $\mathcal{O}\Big(|A|^{\max(1,C)}\Big)$.

Further, if for some constant $C > 0$, we have $k \leq C$, by the fact that $\log(|A|) \leq |\hat{A}|$, the run time is $\mathcal{O}\big(|\hat{A}|^{C+2}\big)$. ◀

4.3 Multi-Arrival for Tree-like Simple Graphs

In this subsection, we focus on tree-like rotor graphs without parallel arcs, for which we prove that Multi-Arrival can be solved in polynomial time.

▶ **Theorem 2.** Multi-Arrival *can be solved on tree-like rotor graphs T with arc set A, sink set S_0 and without parallel arcs in time $\mathcal{O}\big(|S_0||A|^3\big)$. Furthermore, in the procedure, a full routing is obtained.*

We now employ a dynamic program that given bounds $b_1 \leq b_2$ that contain some unknown s-directed compensated routing r, constructs lower bounds of $\{h_v \mid v \in V^+(T)\}$. Then, as r is contained in those bounds, these lower bounds allow us to obtain an s-directed compensated routing $r' \geq r$ in time proportional to the size of these bounds.

▶ **Lemma 19.** *Let T be a tree-like rotor graph, and let b_1, b_2 be two vectors such that there exists an s-directed compensated routing r with $b_1 \leq r \leq b_2$. Then we can construct an s-directed compensated routing r' such that $r' \geq r$ in time $\mathcal{O}(\Delta|\hat{A}|)$ where $\Delta = 1 + \max\big\{\phi^v_w(b_2(v)) - \phi^v_w(b_1(v)) \mid v \in V^+(T) \text{ and } w = N^+_s(v)\big\}$.*

Proof. First, we inductively construct for each $v \in V^+(T)$ a strictly increasing function $\hat{f}_v : I_v \to \mathbb{N}$ where $I_v = \{\phi^v_w(b_1(v)), \dots, \phi^v_w(b_2(v))\}$ and a function $\hat{h}_v : \mathbb{N} \to \mathbb{N}$. The function $\hat{h}_v$ will serve as a lower bound for h_v.

Suppose $\hat{h}_u$ has been constructed for each $u \in N^-_s(v)$. We define $\hat{f}_v$ and $\hat{h}_v$ as follows:

$$\hat{f}_v(x) = q_v(x) - \sigma(v) - \sum_{u \in N^-_s(v)} \hat{h}_u\big(q^v_u(x)\big), \quad \text{for } x \in I_v,$$

$$\hat{h}_v(y) = \begin{cases} 0 & y < \min \hat{f}_v, \\ \max\{x \in I_v \mid \hat{f}_v(x) \leq y\} & \min \hat{f}_v \leq y. \end{cases}$$

It is clear that $\hat{f}_v$ is strictly increasing and $\hat{h}_v$ is an increasing function by arguments analogous to that of Lemma 12. Similarly, as by induction we have $\hat{h}_u \leq h_u$ for all $u \in N^-_s(v)$, it follows that $\hat{h}_v \leq h_v$.

Next, we will show that

$$\hat{h}_v\big(\phi^w_v(r(w))\big) \geq \phi^v_w(r(v)) \quad . \tag{12}$$

To this end, suppose that, by induction, $\hat{h}_u\big(\phi^v_u(r(v))\big) \geq \phi^u_v(r(u))$ for $u \in N^-_s(v)$. Then as r is a compensated routing, we have:

$$\phi^w_v(r(w)) \geq r(v) - \sigma(v) - \sum_{u \in N^-_s(v)} \phi^u_v(r(u)) \quad .$$

Since the function $\phi^v_w(\cdot)$ is increasing and $b_1(v) \leq r(v) \leq b_2(v)$, we have $\phi^v_w(r(v)) \in I_v$. Also, $q_v(\phi^v_w(r(v))) = r(v)$, as r is s-directed.

Consequently,

$$\begin{aligned} \hat{f}_v\big(\phi^v_w(r(v))\big) &= r(v) - \sigma(v) - \sum_{u \in N^-_s(v)} \hat{h}_u\big(\phi^v_u(r(v))\big) \\ &\leq r(v) - \sigma(v) - \sum_{u \in N^-_s(v)} \phi^u_v(r(u)) \leq \phi^w_v(r(w)) \quad . \end{aligned}$$

From this, by the definition of $\hat{h}_v$, (12) readily follows.

Next, we inductively construct an s-directed routing r' and begin by setting $r'(s) = r(s) = 0$. We assume by induction that $r'(w)$ is defined and satisfies $r'(w) \geq r(w)$.

Now, if $w \neq s$ and $r'(w) = r(w) = 0$, we set $r'(v) = 0$. From Lemma 11 we see that $r(v) = 0$. Otherwise, we define

$$r'(v) = q_v\Big(\hat{h}_v\big(\phi_v^w(r'(w))\big)\Big) \ . \tag{13}$$

From $r'(w) \geq r(w)$ and (12) it follows that $\hat{h}_v\big(\phi_v^w(r'(w))\big) \geq \hat{h}_v\big(\phi_v^w(r(w))\big) \geq \phi_w^v(r(v))$. Taking q_v from the left and right sides of this inequality, we get $r'(v) \geq r(v)$.

Since $r' \geq r$, it remains to show that r' is s-directed and compensated. We accomplish this by showing that the conditions of Lemma 11 are satisfied by r'. Let $v \in V^+(T)$ and assume for each $u \in N_s^-(v)$ we have $r'_u \in H_u\big(\phi_v^u(r'(u)), \phi_u^v(r'(v))\big)$ where r'_u denotes the restriction of r' to T_u. We will show that r'_v belongs to $H_v\big(\phi_w^v(r'(v)), \phi_v^w(r'(w))\big)$. By construction, $\phi_w^v(r'(v)) > 0$ if and only if $r'_v(v) > 0$. The case $r'_v(v) = 0$ is trivial as then we see that $r'_v = \mathbf{0}$. Otherwise, we need to show that (1)–(3) of Lemma 11 hold. By the assumption, (3) is already satisfied.

Taking $\phi_w^v(\cdot)$ from the both sides of (13), we obtain $\phi_w^v(r'(v)) = \hat{h}_v\big(\phi_v^w(r'(w))\big)$. So $r'(v)$ satisfies (1).

Finally, by definition of $\hat{h}_v$ and r', we have:

$$\begin{aligned}\phi_v^w(r'(w)) &\geq \hat{f}_v\big(\phi_w^v(r'(v))\big) \\ &= r'(v) - \sigma(v) - \sum_{u \in N_s^-(v)} \hat{h}_u\big(\phi_u^v(r'(v))\big) = r'(v) - \sigma(v) - \sum_{u \in N_s^-(v)} \phi_v^u(r'(u)) \ .\end{aligned}$$

Consequently, (2) holds as well.

Hence, the procedure returns the desired s-directed compensated routing $r' \geq r$.

Note that $\hat{h}_v$ is increasing. Hence, it is straightforward to see that we can construct $\hat{h}_v$ by iterating through $x \in \{\phi_w^v(b_1(v)), \ldots, \phi_w^v(b_2(v))\}$ while keeping track of the index x_u for each $u \in N_s^-(v)$ that corresponds to the maximal $\hat{f}_u(x_u) \leq \phi_u^v(q_v(x))$. Hence, we can construct $\hat{h}_v$ in time $\mathcal{O}\big(|\hat{A}^+(v)|\Delta + |N_s^-(v)|\Delta\big)$. In total, including the construction of r', we thus take time at most $\mathcal{O}(\Delta|\hat{A}|)$. ◀

Proof of Theorem 2. We proceed analogously to the proof of Theorem 1. It suffices to show how to obtain r_s for each sink $s \in S_0$.

We will demonstrate that for the simple tree-like rotor graph T, we can obtain bounds $b_1 \leq b_2$ in time $\mathcal{O}(|A|)$, which contain an s-directed compensated routing r which maximizes $\phi_s(r)$. Additionally, we show that the parameter Δ, defined as in Lemma 19, satisfies $\Delta \leq \mathcal{O}(|A|^2)$. We then plug these into Lemma 19, which allows us to construct the desired routing r_s in time $\mathcal{O}(|A|^3)$. Note that, since T is simple, we have $|\hat{A}| = \mathcal{O}(|A|)$. The remainder of the proof is therefore devoted to finding the required vectors b_1 and b_2.

Let $\{t\} = N_s^-(s)$ and let r denote the component-wise maximal routing in $H_t(h_t(0), 0)$. We first construct, for each $v \in V^+(T)$, the affine approximations $\overline{h}_v$ given in Lemma 15. To this end, define $b_1(s) = b_2(s) = 0$. Now, for $v \in V^+(T)$ and $w = N_s^+(v)$, assume we have constructed $b_1(w) \leq r(w) \leq b_2(w)$. Hence:

$$h_v\big(\phi_v^w(b_1(w))\big) \leq h_v\big(\phi_v^w(r(w))\big) \leq h_v\big(\phi_v^w(b_2(w))\big) \ .$$

Since r is maximal we have that $\phi_w^v(r(v)) = h_v\big(\phi_v^w(r(w))\big)$. From r being s-directed it follows that

$$q_v\big(h_v(\phi_v^w(b_1(w)))\big) \leq r(v) \leq q_v\big(h_v(\phi_v^w(b_2(w)))\big) \ .$$

Hence, using $\overline{h}_v$ with $C_v = 2|A(T_v)|$ denoting a bound on the error, we define:

$$b_1(v) = q_v\Big(\overline{h}_v(\phi_v^w(b_1(w))) - C_v\Big) \leq r(v) \leq q_v\Big(\overline{h}_v(\phi_v^w(b_2(w))) + C_v\Big) = b_2(v) \ .$$

Our next task, is to show that for each $v \in V^+(T)$, it holds that

$$|b_1(v) - b_2(v)| \leq 4d_+(v)|A| \operatorname{dist}(s, v) \ . \tag{14}$$

We prove this relation by induction starting at t. The base case is trivial, as

$$|b_1(t) - b_2(t)| = \Big|q_t\Big(\overline{h}_t(0) - C_t\Big) - q_t\Big(\overline{h}_t(0) + C_t\Big)\Big| \leq 4d_+(t)|A| \ .$$

Assume that (14) holds for v; we now prove it for $u \in N_s^-(v)$. Since T is simple, we have that $q_u(x) - q_u(x') = d_+(u)(x - x')$ and

$$|\phi_u^v(x) - \phi_u^v(x')| \leq \left\lceil \frac{|x - x'|}{d_+(v)} \right\rceil .$$

Furthermore, as $\overline{h}_v$ is a contraction (see (9)), we obtain:

$$\begin{aligned} |b_1(u) - b_2(u)| &= d_+(u)\Big(\big|\overline{h}_v(\phi_u^v(b_1(v))) - \overline{h}_v(\phi_u^v(b_2(v)))\big| + 2C_u\Big) \\ &\leq d_+(u)\Big(|\phi_u^v(b_1(v)) - \phi_u^v(b_2(v))| + 2C_u\Big) \\ &\leq d_+(u)\left(\frac{|b_1(v) - b_2(v)|}{|d_+(v)|} + 2C_u + 1\right) \\ &\leq 4d_+(u)\left(\operatorname{dist}(s, v)|A| + |A(T_u)| + 1\right) \leq 4d_+(u) \operatorname{dist}(s, u)|A| \ . \end{aligned}$$

By (14) it holds that for each $v \in V^+(T)$, $|\phi_w^v(b_1(v)) - \phi_w^v(b_2(v))| = \mathcal{O}(|A|^2)$, implying that the error Δ, defined as in Lemma 19, satisfies $\Delta \leq \mathcal{O}(|A|^2)$. This completes the proof. ◀

5 Conclusion

Our main result is a quasi-polynomial time algorithm for Multi-Arrival on tree-like multigraphs. We established the polynomial-time solvability of Multi-Arrival on path-like multigraphs, and on tree-like multigraphs with bounded contracted height. Thereby, we extended the polynomial-time algorithm by Auger et al., which was restricted to path-like rotor graphs with certain uniform rotor order, to a much wider class of instances.

The notion of contracted height, as we introduce it in this paper, is related to several other classic notions of height on trees, such as topological height. Nonetheless, we have not found any direct reference of this concept in the literature. It may therefore be interesting to explore its relationship to those well-known concepts of height further.

The complexity of Arrival in general graphs remains widely open. As a next step, it could be valuable to improve the quasi-polynomial time algorithms for Multi-Arrival on tree-like rotor graphs to a polynomial-time algorithm.

References

1 David Auger, Pierre Coucheney, and Loric Duhazé. Polynomial time algorithm for ARRIVAL on tree-like multigraphs. In *Proc. MFCS 2022*, volume 241 of *Leibniz Int. Proc. Inform.*, pages Art. No. 12, 14. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPIcs.MFCS.2022.12.

2 David Auger, Pierre Coucheney, Loric Duhazé, and Kossi Roland Etse. Generalized ARRIVAL problem for rotor walks in path multigraphs. In *Proc. Reachability problems 2023*, volume 14235 of *Lecture Notes Comput. Sci.*, pages 183–198. Springer, 2023. doi:10.1007/978-3-031-45286-4_14.

3 Jérôme Dohrau, Bernd Gärtner, Manuel Kohler, Jiří Matoušek, and Emo Welzl. ARRIVAL: a zero-player graph game in $\mathsf{NP} \cap \mathsf{coNP}$. In *A journey through discrete mathematics*, pages 367–374. Springer, Cham, 2017. doi:10.1007/978-3-319-44479-6_14.

4 John Fearnley, Martin Gairing, Matthias Mnich, and Rahul Savani. Reachability switching games. *Log. Methods Comput. Sci.*, 17(2):Paper No. 10, 29, 2021. doi:10.23638/LMCS-17(2:10)2021.

5 Bernd Gärtner, Thomas Dueholm Hansen, Pavel Hubácek, Karel Král, Hagar Mosaad, and Veronika Slívová. ARRIVAL: Next Stop in CLS. In *Proc. ICALP 2018*, volume 107 of *Leibniz Int. Proc. Informatics*, pages 60:1–60:13, 2018. doi:10.4230/LIPICS.ICALP.2018.60.

6 Bernd Gärtner, Sebastian Haslebacher, and Hung P. Hoang. A subexponential algorithm for ARRIVAL. In *Proc. ICALP 2021*, volume 198 of *Leibniz Int. Proc. Inform.*, pages 69:1–69:14. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPIcs.ICALP.2021.69.

7 Giuliano Pezzolo Giacaglia, Lionel Levine, James Propp, and Linda Zayas-Palmer. Local-to-global principles for the hitting sequence of a rotor walk. *Electron. J. Combin.*, 19(1):Paper 5, 23, 2012. doi:10.37236/12.

8 Phuc Hung Hoang. *On Two Combinatorial Reconfiguration Problems: Reachability and Hamiltonicity*. PhD thesis, ETH Zurich, 2022.

9 Alexander E. Holroyd, Lionel Levine, Karola Mészáros, Yuval Peres, James Propp, and David B. Wilson. Chip-firing and rotor-routing on directed graphs. In *In and out of equilibrium. 2*, volume 60 of *Progr. Probab.*, pages 331–364. Birkhäuser, Basel, 2008. doi:10.1007/978-3-7643-8786-0_17.

10 Vyatcheslav B. Priezzhev, Deepak Dhar, Abhishek Dhar, and Supriya Krishnamurthy. Eulerian walkers as a model of self-organized criticality. *Phys. Rev. Lett.*, 77(25):5079, 1996. doi:10.1103/PhysRevLett.77.5079.

11 Lilla Tóthmérész. Rotor-routing reachability is easy, chip-firing reachability is hard. *European J. Combin.*, 101:Paper No. 103466, 9, 2022. doi:10.1016/j.ejc.2021.103466.