Competitive Packet Routing with Priority Lists

TOBIAS HARKS, Department of Mathematics, University of Augsburg
BRITTA PEIS, DANIEL SCHMAND, BJOERN TAUER, and LAURA VARGAS KOCH,
School of Business and Economics, RWTH Aachen University

In competitive packet routing games, the packets are routed selfishly through a network and scheduling policies at edges determine which packets are forwarded first if there is not enough capacity on an edge to forward all packets at once. We analyze the impact of *priority lists* on the worst-case quality of pure Nash equilibria. A priority list is an ordered list of players that may or may not depend on the edge. Whenever the number of packets entering an edge exceeds the inflow capacity, packets are processed in list order. We derive several new bounds on the price of anarchy and stability for global and local priority policies. We also consider the question of the complexity of computing an optimal priority list. It turns out that even for very restricted cases, i.e., for routing on a tree, the computation of an optimal priority list is APX-hard.

CCS Concepts: • **Theory of computation** → **Network optimization**; *Problems, reductions and completeness*; • **Applied computing** → Operations research;

Additional Key Words and Phrases: Packet routing, Nash equilibrium, price of anarchy, priority policy, complexity

ACM Reference format:
Tobias Harks, Britta Peis, Daniel Schmand, Bjoern Tauer, and Laura Vargas Koch. 2018. Competitive Packet Routing with Priority Lists. *ACM Trans. Econ. Comput.* 6, 1, Article 4 (March 2018), 26 pages.
https://doi.org/10.1145/3184137

4

1 INTRODUCTION

A fundamental combinatorial optimization problem that has received considerable attention in the past is *packet routing* in graphs (cf. [4, 15, 18, 19, 22, 26]). We are given a set of packets, which may, for example, correspond to unit-sized messages/bits in a communication network. Originated at possibly different start nodes, the goal is to transfer all packets as fast as possible to their respective destination nodes. It is assumed that each edge is equipped with a capacity (or bandwidth) and a travel time. A prominent variant is *discrete store-and-forward packet routing*, where every node can store arbitrarily many packets, but only a limited number can enter an edge simultaneously at each discrete time step. This model has been used widely to analyze routing problems in computers or other clock-driven chips; see [18].

Authors' addresses: T. Harks, Department of Mathematics, University of Augsburg, Germany, 86135 Augsburg; email: tobias.harks@math.uni-augsburg.de; B. Peis, School of Business and Economics, RWTH Aachen University, Germany, Kackertstraße 7, 52072 Aachen; email: britta.peis@oms.rwth-aachen.de; D. Schmand, School of Business and Economics, RWTH Aachen University, Germany, Kackertstraße 7, 52072 Aachen; email: daniel.schmand@oms.rwth-aachen.de; B. Tauer, School of Business and Economics, RWTH Aachen University, Germany, Kackertstraße 7, 52072 Aachen; email: bjoern.tauer@oms.rwth-aachen.de; L. V. Koch, School of Business and Economics, RWTH Aachen University, Germany, Kackertstraße 7, 52072 Aachen; email: laura.vargas@oms.rwth-aachen.de.

https://doi.org/10.1145/3184137

Table 1. Inefficiency of Packet Routing Games with Global Priority Lists

	Price of Stability		Price of Anarchy	
	Lower Bnd.	Upper Bnd.	Lower Bnd.	Upper Bnd.
Single commod.	1	1	$1+\frac{n-1}{2\Delta}$	$1+\frac{n-1}{2\Delta}$
Multicommod.	$\Theta(\sqrt{\frac{n}{\delta}})$	$1+\frac{n-1}{2\delta}$	$1+\frac{n-1}{2\Delta}$	$\min\{n, 1+\frac{(n-1)\Delta}{2\delta}\}$

The results are proven in Theorems 3.1 and 3.2, Propositions 3.3 and 3.6, and Corollary 3.5.

In many settings, it may be expensive or impossible to regulate network traffic by a central authority in order to implement an optimal assignment of routes. In this work, we focus on *selfish* or *competitive* packet routing using the discrete store-and-forward packet routing model. We are given a multi- or single-commodity network, where the commodities are specified by a source and a sink node. Each commodity represents one or several rational players, who route selfishly one packet from their source to their sink through the network. Each edge of the network is endowed with an integral travel time and an integral capacity. The capacity of the edge defines the number of players that may *enter* the edge simultaneously. For each edge, we are given a priority list (i.e., an ordered list of the players) to resolve conflicts whenever more than capacity many players seek to enter that edge at the same point in time. Here we assume that all players are ready to start right from the beginning (i.e., there are no release dates). If a player with release date r_i exists, we can extend the graph G by vertex s_i' and edge (s_i', s_i) with capacity one and travel time r_i and ignore the release date in the new graph. Note that this may change a single-commodity network into a multicommodity network if the release dates differ. The players aim to minimize their respective arrival times at the sink. Since the outcome of this competitive situation intrinsically depends on the priority lists employed on the edges, the problem of finding *good* priority lists renders into a coordination mechanism design problem. See [6] for the first landmark paper and several follow-ups [1, 5, 7, 13].

1.1 Our Contribution

In this article, we explore properties of selfish discrete store-and-forward packet routing with *priority-based* scheduling policies. We consider local priority lists and global priority lists. In the case of local priority lists, the predefined order of players may be different among edges. For global priority lists, it is the same for all edges. We obtain the following results, where n denotes the number of players and $\text{dist}(i)$ is the length of a shortest s_i-t_i-path. We write $\delta = \min_{i\in N} \text{dist}(i)$ and $\Delta = \max_{i\in N} \text{dist}(i)$, where N is the player set.

Price of Anarchy/Stability. To measure the efficiency of a PNE, the price of anarchy (PoA for short) and the price of stability (PoS) are widely used concepts. The price of anarchy is the quotient of the costs of the players in a worst PNE and the costs of the players in an optimal solution. The price of stability is the quotient of the costs of the players in a best PNE and the costs of the players in an optimal solution. For global priority lists and multicommodity instances, we show that the price of stability is upper bounded by $1 + \frac{n-1}{2\delta}$ and the price of anarchy by $\min\{n, 1 + \frac{(n-1)\Delta}{2\delta}\}$. For global priority lists and single-commodity games (i.e., all packets travel from a common source to a common sink), the PoS is one, while we derive a tight bound of $1 + \frac{n-1}{2\Delta}$ for the PoA for these games. Note that in single-commodity games, we have $\delta = \Delta$. An overview of the results for global priority lists is given in Table 1.

For local priority lists and multicommodity games, we derive that the PoA is between $\frac{\tilde{D}}{4}$ and $6\tilde{D}^2 - \tilde{D}$. Here, $\tilde{D}$ is a kind of *dilation* of the graph, i.e., the maximal length of an s_i-t_i-path, where

Table 2. Inefficiency of Packet Routing Games with Local Priority Lists, if Equilibria Exist

	Price of Stability		Price of Anarchy	
	Lower Bnd.	Upper Bnd.	Lower Bnd.	Upper Bnd.
Single commod.	?	?	$\frac{\tilde{D}}{4}$	$6\tilde{D}^2 - \tilde{D}$
Multicommod.	$\Theta(\sqrt{\tilde{D}})$	$6\tilde{D}^2 - \tilde{D}$	$\frac{\tilde{D}}{4}$	$6\tilde{D}^2 - \tilde{D}$

The results are proven in Theorem 3.7 and Remarks 3.9 and 3.10.

edges with travel time 0 contribute 1 to the path. Note that $\delta \leq \Delta \leq D \leq \tilde{D}$, where D denotes the dilation, i.e., the maximal length of an s_i-t_i-path with respect to the travel times. This result on the PoA is obtained via adapting the primal-dual technique introduced by Kulkarni and Mirrokni [17]. Note that the network model in [17] is different than ours in the sense that it allows for different weights and sizes of the packets. We refer for a further comparison to Section 1.2. While the result in [17] holds for a very specific local scheduling policy only, namely, *Highest Density First*, our result even applies to *arbitrary* local priority lists, due to the special network structure in our model. As a byproduct of applying the primal-dual technique, we obtain the same bounds even for coarse correlated equilibria that are guaranteed to exist.

Note that these bounds do only depend on $\tilde{D}$ and not on n, and thus are constant for any given network. An overview of the results for local priority lists is given in Table 2.

We also present refined upper and lower bounds depending on the minimal edge capacity $u_{\min}$, which match the previously described bounds for $u_{\min} = 1$. Since the main construction of the proofs do not change for $u_{\min} > 1$, we only sketch these proofs.

Computational Complexity. We then turn to the question of computing *optimal* priority lists, that is, priority lists that induce the best possible social optima or Nash equilibria. Here a social optimum is a vector of paths that minimizes the sum of arrival times of all players. It turns out that in single-commodity instances, the sum of arrival times in a social optimum is independent of the priority lists (see Proposition 2.8). Additionally, we show that all global priority lists guarantee a PoS of 1 in single-commodity networks (see Theorem 3.1). The complexity of computing priority lists that minimize the PoA in single-commodity instances remains open. For multicommodity graphs, the problem of defining global as well as local priority lists for minimizing the cost of any Nash equilibrium or of any social optimum is APX-hard. Note that this is the first hardness result for the underlying coordination mechanism design problem and complements several approximability results for the tree case recently derived by Bhattacharya et al. [2]. Technically, we adapt a construction of Peis et al. [22], where it is shown that the problem to compute a schedule minimizing the makespan (the latest arrival of any packet) is APX-hard. The question of computing an optimal local priority list with respect to minimizing the inefficiency of Nash equilibria for single-commodity instances remains open.

We finally derive several further hardness results for our model: in multicommodity games with local priority lists, it is NP-hard to decide if there is a pure Nash equilibrium. Moreover, it is NP-hard to compute a best response in single-commodity games with local priority lists. These results are obtained by adapting the reduction described in Hoefer et al. [12], which is used to prove hardness in a more general setting. An overview of the results for local priority lists is illustrated in Table 3. An open question is the complexity of computing a pure Nash equilibrium in a single-commodity game with local priority lists. Attention should be paid to the fact that there exist competitive packet routing games with local priority lists that improve the PoA in contrast to any

Table 3. Computational Complexity in Packet Routing Games with a Local Priority List

	PNE	Best Response	Opt. Priority List	Social Opt.
Single commod.	?	NP-hard	?	P
Multicommod.	NP-hard	NP-hard	APX-hard	no PTAS

The results are proven in Propositions 2.8 and 4.2 and Theorems 4.1 and 4.4.

Table 4. Computational Complexity in Packet Routing Games with a Global Priority List

	PNE	Best Response	Opt. Priority Lists	Social Opt.
Single commod.	P	P	$O(1)$	P
Multicommod.	P	P	APX-hard	no PTAS

The results are proven in Proposition 2.8, Theorems 4.1 and 4.4, and Observation 4.3.

global priority list. For example, there exists a local priority list such that the PoA of the nth Braess graph is equal to 1, whereas it is $\frac{n+1}{2}$ for every global priority list and travel time 1 on the source-sink paths; see Figure 4. For global priority lists, we get an efficient Dijkstra-type algorithm for computing a best response and, thus, a pure Nash equilibrium. Again, an overview of the results for global priority lists is depicted in Table 4.

1.2 Related Work

Competitive Routing over Time. Hoefer et al. [12] considered weighted network congestion games in the continuous-time setting. In their model, the edges represent machines with predefined speed. Each job has a weight, and the time needed to traverse an edge is given by the product of speed and weight. In contrast to our model, traversal times in [12] might be fractional, but zero travel times are not allowed. Furthermore, the type of capacity constraints is different: while [12] capacitates the total weight of players using an edge at the same point in time, our model capacitates the number of players that may enter an edge simultaneously. Roughgarden and Tardos [24] initiated the study of the price of anarchy in nonatomic network games. Hoefer et al. analyzed four different scheduling policies: *FIFO*, *nonpreemptive global ranking*, *preemptive global ranking,* and *fair Time-Sharing*. They showed that in the case of a global priority list, at least one equilibrium exists and it can be computed efficiently by iteratively and greedily routing the players with respect to the global order. Further results include the nonexistence of equilibria for the FIFO scheduling policy and the complexity of computing equilibria and best responses.

The model of Kulkarni and Mirrokni [17] is a generalization of the model of Hoefer et al. [12] with two main differences. First, each packet has a size as well as a weight. The size determines how long it takes to traverse an edge and the weight denotes the contribution of this packet to the social cost. Observe that the authors also exclude edges with traversal time 0 for a packet by their definition of the processing time as size divided by speed of an edge. The second difference to our model is that Kulkarni and Mirrokni assume that the strategy space of a packet is a *subset* of the simple paths from its source to its sink, whereas it is the set of *all* simple source-sink paths in [12] and in our model. Kulkarni and Mirrokni consider a variant of the robust price of anarchy, which is the worst-case ratio of the social cost of a coarse-correlated equilibrium and that of a social optimum [17]. A general framework to bound the robust price of anarchy via LP-Duality or Fenchel Duality is introduced by [17]. For the *Highest Density First* policy, they derive an upper bound of $4D^2$ for the robust price of anarchy. Here, D is the dilation of the graph, i.e., the length of a longest s_i-t_i-path. They also show a lower bound of $\frac{D}{16}$.

Flows over Time. Noncompetitive packet routing can be interpreted as a special variant of *flows over time* (also known under the name *dynamic flows*) as introduced by Ford and Fulkerson in their seminal paper [10]. In fact, (noncompetitive) packet routing is exactly the problem to find an integral multicommodity flow satisfying unit demands of minimum time horizon. For an introduction to flows over time, we refer to Skutella [25].

Koch and Skutella introduced in [16] a game-theoretic variant of flows over time. In their model, a continuum of players route selfishly from a source to a sink through a network and flow enters an edge in a continuous fashion. They showed the existence of equilibria and analyzed the price of anarchy for their model; see also Cominetti et al. for a constructive proof for the existence and uniqueness of equilibria in [8]. Koch et al. [14] introduced discrete time steps for a single-commodity model. One could see our work as an extension to unsplittable flow particles and multicommodity networks.

2 PRELIMINARIES

An instance of a competitive packet routing game is a tuple (N, G, u, τ, π) consisting of a directed graph $G = (V, E)$ with integral travel times $\tau_e \in \mathbb{Z}_{\geq 0}$ denoting the time needed to traverse an edge $e \in E$. Additionally, each edge $e \in E$ is endowed with a capacity $u_e \in \mathbb{Z}_{>0}$ denoting the number of players that can enter an edge e simultaneously at *each* integral time step. Note that this is independent of the travel times, even for $\tau_e = 0$. Allowing edges with travel time $\tau_e = 0$ seems to be useful for many applications. For example, this gives us the possibility to model a throughput capacity of a node v by the following standard transformation. Replace v by uncapacitated nodes v' and v'' with a capacitated edge $e = (v', v'')$ with $\tau_e = 0$ and connect all incoming edges of v to v' instead of v and all outgoing edges of v with v'' instead of v. The existence of $\tau_e = 0$ edges in the model is very important for this transformation. The set of players in competitive packet routing games is denoted by $N = \{1, \ldots, n\}$. Each player $i \in N$ is associated with a source s_i and sink t_i, inducing a strategy space $\mathcal{P}_i \subseteq 2^E$ consisting of all possible simple paths in G linking the respective source and sink. We call an instance *single commodity* if all players start at the same node and have the same sink, i.e., $s_i = s_j$ and $t_i = t_j$ for all players $i, j \in N$. But also in this case we assume, as common in game theory, that the encoding size of the game is proportional to n.

Depending on the chosen strategies, there might be more than u_e-many players seeking to enter an edge at the same integral time step $\theta \in \mathbb{Z}_{\geq 0}$. To resolve such conflicts, we use priority lists to define a total ordering of all players on each edge.

Definition 2.1. A *priority list* $\pi_e : N \to \{1, \ldots, n\}$ is a permutation of all players. For players seeking to traverse edge e at time $\theta \in \mathbb{Z}_{\geq 0}$, the u_e players of highest priority according to list π_e may enter and travel along edge e, while the remaining players need to wait (at least) one time step. A priority list $\pi = (\pi_e)_{e \in E}$ is called *global* if $\pi_e = \pi_{e'}$ for all edges e and e'; otherwise, it is called *local*.

Since all results hold for unit capacity, we improve readability by the following preprocessing. Given an instance (N, G, u, τ, π), we replace each edge e with capacity $u_e > 1$ by u_e-many parallel edges with unit capacity. Observe that at most n players can enter an edge at each time step, w.l.o.g. $u_e \leq n$. Since we define each player to be encoded with his or her source and his or her sink even in the single-commodity case, the network remains polynomial after the preprocessing. Note that this enlarges the strategy space of the players by allowing them to wait for entering an edge if this edge is used by at least one other player. Each player $i \in N$ selects one path P_i from $\mathcal{P}_i$ with the

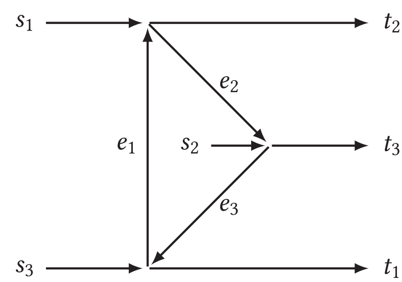

Fig. 1. A graph showing that the mapping of paths to arrival times is not necessarily well defined.

goal to minimize the time when his or her packet entirely reaches his or her sink t_i. This time not only depends on the length $\tau(P_i) = \sum_{e \in P_i} \tau_e$ of path P_i but also on the time the packet needs to wait at intermediate nodes due to interferences with players of higher priority. Given a strategy profile or state $P = (P_1, \ldots, P_n)$, we denote by $C_i(P)$ (or C_i if the context is clear) the time needed for player i's packet to entirely reach sink t_i. We define

$$C_i(P) = \sum_{e \in P_i} (\tau_e + w_{i,e}(P)), \tag{1}$$

where $w_{i,e}(P)$ is the waiting time for player i on the entry of edge e under profile P. The *social cost* of state $P = (P_1, \ldots, P_n)$ is the sum of all players' costs, i.e., $C(P) = \sum_{i \in N} C_i(P)$. We call a profile P *(socially) optimal* if it minimizes the social cost $C(P)$ over the set of all possible profiles. State P is a *pure Nash equilibrium (PNE)* if $C_i(P) \leq C_i(P_{-i}, P'_i)$ holds for each player $i \in N$ and each alternative strategy $P'_i \in \mathcal{P}_i$. Here, as usual, state (P_{-i}, P'_i) is obtained from P by replacing strategy P_i with P'_i.

In the definition of the arrival time of player i (cf. Equation (1)), we implicitly assumed that the values $C_i(P), i \in N$, for a given state $P = (P_1, \ldots, P_n)$ are actually well defined. We show in the following example that directed cycles of length 0 might be harmful.

Example 2.2. Consider the graph depicted in Figure 1. We are given three players, and player i travels from s_i to t_i for $i \in \{1, 2, 3\}$. In this example, each player has exactly one strategy and the priority lists on the edges e_1, e_2, and e_3 are given as $\pi_{e_1} = (2, 3)$, $\pi_{e_2} = (3, 1)$, $\pi_{e_3} = (1, 2)$. The travel times on the edges e_1, e_2, e_3 are equal to 0, whereas all other edges have travel time 1. The capacities of all edges are equal to 1. Now, there is no feasible integral flow over time respecting both the capacity constraints and the priority lists. Therefore, it is not possible to map each player $i \in N$ to a real-valued arrival time $C_i(P)$.

A necessary condition for a well-defined game is to have a function mapping strategy profiles to players' costs. The observation above motivates the exclusion of directed 0-cycles, that is, cycles C of length $\tau(C) = \sum_{e \in C} \tau_e = 0$. The following proposition shows that it is sufficient to exclude directed 0-cycles. That is, given any strategy profile P, the embedding of players to arrival times $C_i(P), i \in N$, is well defined, as long as directed 0-cycles are excluded. Thus, such a function exists.

Proposition 2.3. *Given an instance without directed 0-cycles, we can use a Dijkstra-like algorithm to map given paths $P = (P_1, \ldots, P_n)$ to a flow over time, thus to arrival times $C_i(P)$ in polynomial time.*

Proof. The idea is to adapt Dijkstra's algorithm [9] as follows. For each node v, we additionally define a list containing the arrival times of the players at v by the following procedure: initialize all source nodes s_i with arrival time 0 for every player starting at s_i. Use a priority queue of nodes sorted by the smallest arrival time of any player at that node. In each step, extract all nodes of minimal arrival time from the queue. Consider the graph H induced by these nodes and the corresponding edges of length $\tau_e = 0$. Repeatedly choose a node without incoming edges. Note that such a node needs to exist, since graph G is assumed to be free of 0-cycles. For each chosen node, route all players with minimal arrival time being able to depart according to the priority lists of the outgoing edges and delete the arrival time of the routed players. Add the arrival time of the routed players to the next node on their paths and reintroduce the node into the priority queue, if necessary. If the node is already in the priority queue, we possibly change its order in the queue. Now, delete the current node from H and go on with the next node without incoming edge. If H is empty, continue with the next nodes from the queue. □

Note that networks with paths with total travel time equal to 0 can lead to instances where any waiting time increases the arrival time by an arbitrarily large factor. One could address this problem by only analyzing additive gaps in the objective function. In contrast to this, we are interested in the analysis of relative gaps and assume in the following that there are no paths $P \in \mathcal{P}_i$ with total travel time equal to 0.

Using the relationship between packet routing and integral flows over time [25], we show that at least in the single-commodity setting, a social optimum can be computed via earliest arrival flows. Gale showed [11] that earliest arrival flows do always exist in single-commodity networks.

Definition 2.4. Let $s, t \in V$. An *integral s-t-flow over time* is a set of functions $f_e : \mathbb{Z}_{\geq 0} \to \mathbb{Z}_{\geq 0}$ for all $e \in E$ satisfying the following two constraints:

$$f_e(\theta) \leq u_e \qquad \forall e \in E, \theta \in \mathbb{Z}_{\geq 0} \tag{2}$$

$$\sum_{e \in \delta^-(v)} \sum_{\theta=0}^{\xi-\tau_e} f_e(\theta) \geq \sum_{e \in \delta^+(v)} \sum_{\theta=0}^{\xi} f_e(\theta) \qquad \forall \xi \in \mathbb{Z}_{\geq 0}, v \in V \setminus \{s, t\}. \tag{3}$$

The first inequality constrains the capacity and the second one represents weak flow conservation. If Equation (3) is fulfilled with equality, the flow is said to satisfy strong flow conservation. Here $\delta^-(v)$ is the set of all edges that enter vertex v and $\delta^+(v)$ is the set of all edges that leave vertex v. An integral s-t-flow over time fulfills the *earliest arrival property* if it maximizes the amount of flow arriving at the sink t for every integral time step. An integral s-t-flow over time that fulfills the earliest arrival property is called an *integral earliest arrival s-t-flow*.

Lemma 2.5 (Wilkinson [28]). *A cycle-free integral earliest arrival s-t-flow with flow value n and strong flow conservation can be computed in polynomial time in n by an adapted successive shortest-path algorithm.*

Remark 2.6. In the setting of Wilkinson, the successive shortest-path algorithm is pseudo-polynomial since the demand n is encoded as a number in the input. Since we define each player to be encoded with his or her source and his or her sink even in the single-commodity case, the algorithm runs in polynomial time in our model.

Wilkinson's algorithm computes a cycle-free integral earliest arrival s-t-flow with strong flow conservation. In the following, we describe how to obtain an earliest arrival state from the output of Wilkinson's algorithm. An earliest arrival state is a strategy profile such that for every point in time, the number of players that have already reached the destination is maximized.

First, calculate a path decomposition in the sense that there are n paths that contain one unit of flow. Notice that all paths are cycle-free. Next, identify each path with one player in the game. For each used outgoing edge of s, rename the players that use that edge according to the order given by the local priority list. Observe that these paths P_i are feasible strategies. Thus, we have constructed a feasible strategy profile P for the competitive packet routing game.

It remains to show that this state is indeed an earliest arrival state. We sketch this by arguing that at each point in time on all edges, the number of packets according to P coincides with the amount of flow of Wilkinson's output. Suppose there is an edge $e = (u, v)$ and a point in time θ where this is not the case. Among all these occurrences, choose one with minimal time and minimal distance between u and s. We distinguish two cases, either $u = s$ or $u \neq s$. We start with the latter. The number of packets on all incoming edges of u is the same as the amount of flow of Wilkinson's output. Wilkinson's output fulfills the strong flow conservation, and thus there is no intermediate storage of flow. So the only way to deviate is to decrease the flow on e at time θ. This means a packet has to wait before entering edge e at time θ. This is a contradiction to the fact that the output of Wilkinson's algorithm is a feasible flow.

If $u = s$, the flows coincide as well. Wilkinson's flow as well as state P send the same amount of flow into edge e in total. Due to the preprocessing, every edge in the strategy set has unit capacity and is thus fully used in both cases from the beginning on.

Lemma 2.7. *An earliest arrival state in a single-commodity competitive packet routing game is a social optimum, and vice versa.*

Proof. Let P be a social optimum and P' be an earliest arrival state. Let $N_\theta^P = |\{i : C_i(P) \leq \theta\}|$ denote the number of players with arrival time less than or equal to θ. Note that inequality $\sum_{i \in N} C_i(P) \leq \sum_{i \in N} C_i(P')$ can be equivalently written as

$$\sum_{\theta \in \mathbb{Z}_{>0}} \left(N_\theta^P - N_{\theta-1}^P\right)\theta \leq \sum_{\theta \in \mathbb{Z}_{>0}} \left(N_\theta^{P'} - N_{\theta-1}^{P'}\right)\theta.$$

Adding $\sum_{\theta \in \mathbb{Z}_{>0}} (N_{\theta-1}^P - N_\theta^{P'})\theta$ on both sides of the inequality yields

$$\sum_{\theta \in \mathbb{Z}_{>0}} \left(N_\theta^P - N_\theta^{P'}\right)\theta \leq \sum_{\theta \in \mathbb{Z}_{>0}} \left(N_{\theta-1}^P - N_{\theta-1}^{P'}\right)\theta = \sum_{\theta \in \mathbb{Z}_{>0}} \left(N_\theta^P - N_\theta^{P'}\right)\theta + \sum_{\theta \in \mathbb{Z}_{>0}} \left(N_{\theta-1}^P - N_{\theta-1}^{P'}\right).$$

It follows that $\sum_{\theta \in \mathbb{Z}_{>0}} (N_{\theta-1}^P - N_{\theta-1}^{P'}) \geq 0$. On the other hand, $N_{\theta-1}^P - N_{\theta-1}^{P'} \leq 0$ for all θ due to the earliest arrival property of P'. As a consequence, $N_\theta^P = N_\theta^{P'}$ for all θ. This completes the proof. □

We conclude that we can compute a social optimum via the adapted successive shortest-path algorithm presented by Wilkinson.

Proposition 2.8. *A social optimum of a single-commodity competitive packet routing instance can be computed in polynomial time even for local priority lists.*

While this proposition shows that a socially optimal profile can be computed in polynomial time in single-commodity games, we prove in Section 4 that the computational complexity of computing a social optimum becomes APX-hard in multicommodity games, even when restricted to global priority lists.

The following example shows that a PNE does not necessarily exist even in very simple two-player games:

Example 2.9. Consider the network shown in Figure 2 with three nodes $V = \{s, u, t\}$, two parallel edges e_1, e_2 linking s and u, and one edge e_3 linking u and t, all edges of unit capacity and unit travel time. Suppose we are given two players, both with source s and sink t. Now, if the priorities are chosen such that player 1 has priority on the two s-leaving edges, i.e., $\pi_{e_1} = \pi_{e_2} = (1, 2)$, and player 2 has priority on the t-entering edge, i.e., $\pi_{e_3} = (2, 1)$, then the resulting packet routing game does not admit a PNE. In this game, player 1 tries to choose the same path as player 2 and player 2 always

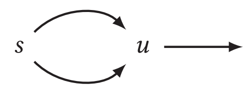

Fig. 2. A network without pure Nash equilibrium.

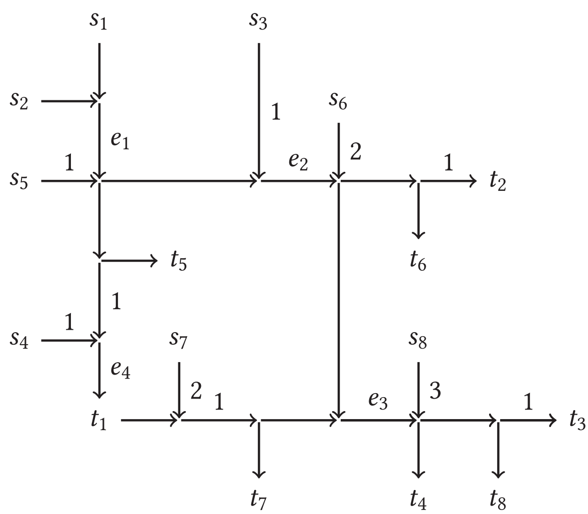

Fig. 3. A graph showing that there are games where edge-dependent priority lists may induce lower social cost than global priority lists.

chooses the free path. It follows that already this very simple two-player single-commodity game does not admit a PNE.

We have seen that not all priority lists guarantee pure Nash equilibria in competitive packet routing games. However, for games with global priority lists, a pure Nash equilibrium can be guaranteed to exist. This can be seen easily by letting players play best responses one by one according to the priority list (see Observation 4.3, similarly observed by Hoefer et al. [12]).

Certainly, the social cost of a profile highly depends on the chosen priority lists $\pi_e, e \in E$. In fact, the restriction to global priority lists might lead to higher social cost and the gap might be arbitrarily large.

Observation 2.10. *There is a competitive packet routing game where the difference of the social cost for any global priority list to the minimal social cost achievable with local priority lists is arbitrarily large.*

Proof. Consider the graph depicted in Figure 3. All edges have unit capacity and travel time 0 if not denoted otherwise in the graph. There is one player starting at s_i and going to t_i for $i \in \{1, 2, 3, 4\}$ and a large constant k of players starting at s_i and going to t_i for $i \in \{5, 6, 7, 8\}$. Since all paths are fixed, the social cost depends only on the waiting times. If and only if player 1 goes in front of player 2 on edge e_1, player 2 in front of player 3 on edge e_2, player 3 in front of player 4 on edge e_3, and player 4 in front of player 1 on edge e_4, there is no collision with the players starting at s_5, s_6, s_7, s_8. In this case, players 1, 2, 3, and 4 wait 1 unit of time and the other players wait $4 \cdot \sum_{i=0}^{k-1} i$ units of time in total. Since the priorities from above cannot be satisfied from a global scheduling rule, each global scheduling rule would lead to a collision of player 1, 2, 3, or 4 with one of the sets of k players, which gives, no matter which priority we choose, an additional waiting time of k. □

3 INEFFICIENCY OF PURE NASH EQUILIBRIA

In this section, we examine the inefficiency of pure Nash equilibria. For the case of global priority lists, we find bounds depending on the number of players n and the length of the shortest paths δ and Δ. We have $\delta = \Delta$ for the single-commodity case. We show a tight bound of 1 for the PoS for single-commodity games, a multicommodity game with PoS in $\Theta(\sqrt{\frac{n}{\delta}})$, and an upper bound of $1 + \frac{n-1}{2\delta}$ by using the idea of so-called blocking times. We achieve a tight bound of $1 + \frac{n-1}{2\Delta}$ for the PoA in the single-commodity case. The PoA in the multicommodity case is upper bounded by $\min\{n, 1 + \frac{(n-1)\Delta}{2\delta}\}$. An overview of the results for global priority lists is depicted in Table 1.

For local priority lists, it turns out that it is much harder to find bounds on the price of anarchy and price of stability. We use a technique introduced in [3] and [17] to prove that the PoA is between $\frac{\tilde{D}}{4}$ and $6\tilde{D}^2 - \tilde{D}$. Note that this bound is constant for any given network, i.e., independent of the number of players.

3.1 Global Priority Lists

Theorem 3.1. *In single-commodity competitive packet routing games with global priorities, the price of stability is equal to 1.*

Proof. Consider a single-commodity instance with global priority list π. Note that the choice of the priority list does not matter since all players have the same start and target node. Up to renaming, suppose $\pi = (1, \ldots, n)$. Observe that a social optimum P fulfills the earliest arrival property due to Lemma 2.7. Note that $C_i(P) \leq C_j(P)$ whenever $i \leq j$. We claim that such a profile is a pure Nash equilibrium. For the sake of contradiction, suppose there exists one player with an improving move. Among all players with an improving move, we choose one of smallest index, say, k. If k decreases his or her cost by switching his or her strategy from path P_k to P'_k, the arrival time of player k decreases, while the arrival time of players $\{1, \ldots, k-1\}$ stays the same. However, this is a contradiction to the fact that a socially optimal profile admits the earliest arrival property. □

Theorem 3.1 shows that there always exists a socially optimal pure Nash equilibrium, as long as we restrict to global priority lists and single-commodity games.

In the following, we show that the price of anarchy for this case is at most $1 + \frac{n-1}{2\Delta}$, where Δ is the distance of s and t. This yields an upper bound on the price of anarchy of $\frac{n+1}{2}$ independent of Δ. There is also a matching lower-bound example for all $\Delta \geq 1$.

Theorem 3.2. *For all single-commodity competitive packet routing games with global priority lists and n players, the price of anarchy is upper bounded by $1 + \frac{n-1}{2\Delta}$, where Δ is the distance of s and t in the given graph. There is a matching lower-bound example.*

Proof. We first prove the upper bound. Let $P = (P_1, \ldots, P_n)$ be any pure Nash equilibrium. We start by showing that the cost of the pure Nash equilibrium P is at most

$$C(P) \leq n \cdot \Delta + \frac{n(n-1)}{2}.$$

Since $\sum_{i=1}^{n} (\Delta + i - 1) = n \cdot \Delta + \frac{n(n-1)}{2}$, it suffices to show that the arrival time of the ith player in the priority list does not exceed $\Delta + i - 1$ time units. Suppose this is not the case—i.e., there are players arriving later. Clearly, the first player in the priority list needs Δ time units. Now, let j be the player who is the first one arriving late in the order of the priority list. It follows that player j

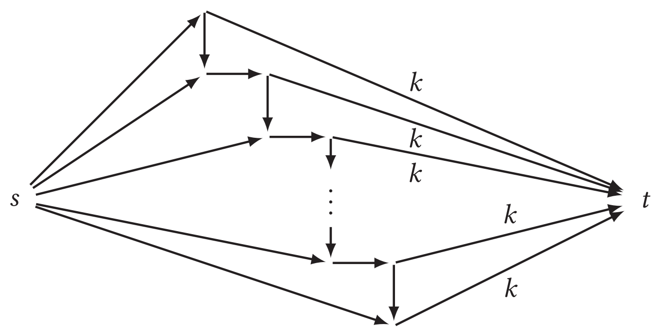

Fig. 4. A game with price of anarchy equal to $1 + \frac{n-1}{2\Delta}$.

has the following improving move: he or she can start at source s and follow player 1. If he or she has to wait at the entry of any edge, he or she follows the player entering the edge directly ahead of him or her. Player j can only be delayed by players arriving on time. By using this strategy, he or she arrives at sink t at the latest 1 time unit after the last player being able to delay him or her. Hence, he or she can guarantee to reach sink t by time $\Delta + i - 1$.

Observe that no player can arrive before time unit Δ. So, the cost of the optimal solution P^* can be lower bounded by $n\Delta$. This yields the following bound:

$$\text{PoA} \leq \frac{C(P)}{C(P^*)} \leq \frac{n \cdot \Delta + \frac{n(n-1)}{2}}{n\Delta} = \frac{\Delta + \frac{n-1}{2}}{\Delta} = 1 + \frac{n-1}{2\Delta}.$$

This completes the proof for the upper bound on the PoA and is well defined since $\Delta \geq 1$.

For showing the tightness of the result, consider the example depicted by the graph of Figure 4. This Braess-like graph topology has been used before to show lower bounds on the PoA in other settings; e.g., see [17]. The travel times τ_e are depicted next to the edges, where edges without a label have $\tau_e = 0$. We define $u_e = 1$ for all edges $e \in E$. Note that there are n direct paths (i.e., those without vertical edges), each with travel time k. Note that the length of every s-t-path is k, and thus $\Delta = k$. In an optimal solution, all players pick a direct path, resulting in an arrival time of Δ for each player, and an optimal social cost of $n\Delta$. However, the profile in which all players use the path containing all vertical edges is a pure Nash equilibrium as well. The ith player in the order of the priority list arrives at time $\Delta + i - 1$ and has no incentive to deviate, since there is no possible path he or she can use without waiting for all players of higher priority. Thus, there is a pure Nash equilibrium of social cost equal to

$$\sum_{i=1}^{n} (\Delta + i - 1) = n\Delta + \frac{n(n-1)}{2}.$$

As a consequence, we obtain

$$\text{PoA} = \frac{n\Delta + \frac{n(n-1)}{2}}{n\Delta} = 1 + \frac{n-1}{2\Delta},$$

which completes the proof. □

For an arbitrary lower bound $u_{\min}$ on the minimal edge capacity in the original network (without preprocessing), where $\frac{n}{u_{\min}}$ is integral, we can derive a tight bound of

$$\text{PoA} = 1 + \frac{1}{2\Delta}\left(\frac{n}{u_{\min}} - 1\right).$$

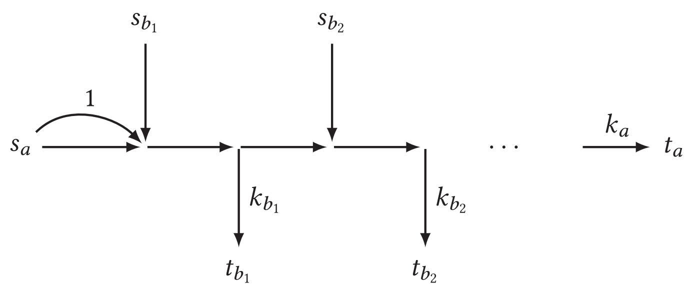

Fig. 5. A game with price of stability in $\Theta(\sqrt{\frac{n}{\delta}})$.

We omit the proof as it follows the same argumentation as above, except that for deriving the upper bound, we additionally use the fact that $u_{\min}$ players can arrive simultaneously at each point in time. For the lower bound, we modify the example slightly by replacing every edge by an edge with capacity $u_{\min}$.

For multicommodity competitive packet routing games, earliest arrival flows do not necessarily exist [25]. It turns out that the price of stability might not even be constant in this more general setting. We provide an example where the price of stability is in $\Theta(\sqrt{\frac{n}{\delta}})$.

Proposition 3.3. *There is a multicommodity competitive packet routing game with a global priority list and price of stability in* $\Theta(\sqrt{\frac{n}{\delta}})$.

Proof. Consider the game illustrated by the graph in Figure 5. There are a players with source s_a and sink t_a, and b players each with individual source s_{bi} and sink t_{bi}, $i \in \{1, \ldots, b\}$. All edges have unit capacity. The travel time is 0 if not depicted otherwise in Figure 5. We consider a global priority list in which all horizontal players have priority over the vertical players.

In the only pure Nash equilibrium, all horizontal players choose the direct path. Thus, their arrival times sum up to $\sum_{i=1}^{a}(k_a + i - 1)$. The vertical players need to wait until all horizontal players have passed. Thus, their arrival times sum up to $\sum_{j=1}^{b}(a + k_{b_j})$. In an optimal solution, all horizontal players choose the longer path with the first edge of travel time 1. Therefore, their cost is $\sum_{i=1}^{a}(k_a + i)$, while the cost of the vertical players decreases to $\sum_{j=1}^{b} k_{b_j}$. Due to these considerations, the price of stability of this game can be expressed by

$$\begin{aligned} \text{PoS} &= \frac{ak_a - a + \frac{a(a+1)}{2} + ba + \sum_{j=1}^{b} k_{b_j}}{ak_a + \frac{a(a+1)}{2} + \sum_{j=1}^{b} k_{b_j}} \\ &= 1 + \frac{a(b-1)}{ak_a + \frac{a(a+1)}{2} + \sum_{j=1}^{b} k_{b_j}}. \end{aligned}$$

Choosing $k_a = k_{b_1} = \cdots = k_{b_b}$, we can lower bound the PoS by

$$\text{PoS} \geq 1 + \frac{2a(b-1)}{2n\delta + a(a-1)}.$$

If we replace b by $n - a$, we can differentiate the expression with respect to a. Taking the root, we get $a = -2\sqrt{\delta} + \sqrt{\delta}\sqrt{4\delta + 2n - 2}$. By substitution, we get a price of stability that is in $\Theta(\sqrt{\frac{n}{\delta}})$. □

For an arbitrary $u_{\min} > 1$, we also choose $k_a = k_{b_1} = \cdots = k_{b_b}$. We can bound the PoS by

$$1 + \frac{\frac{ab}{u_{\min}} - a}{n\delta + \frac{a}{2}(\frac{a}{u_{\min}} - 1)}.$$

Analogously to the calculations above, we get a price of stability that is in $\Theta(\sqrt{\frac{n}{\delta u_{\min}}})$.

Next we present bounds for the price of stability and the price of anarchy by estimating the waiting times of the players in an equilibrium state.

We call a best response P_i of player i to P_{-i} a *greedy best response* if he or she reaches every intermediate node of his or her path P_i as early as possible. The notion of greedy best responses has been introduced in [12].

For a socially optimal profile P^* and a profile P, in which every P_i is a greedy best response, we show the following: the term $\frac{C(P)}{C(P^*)}$ in multicommodity competitive packet routing games with global priority lists is upper bounded by $1 + \frac{n-1}{2\delta}$. Such a profile P, in which every P_i is a greedy best response, always exists (see Observation 4.3). This means that the price of stability is also upper bounded by $1 + \frac{n-1}{2\delta}$ in this case.

THEOREM 3.4. *Let $P = (P_1, \ldots, P_n)$ be a profile in a competitive packet routing game with a global priority list such that every P_i is a greedy best response and let P^* be a socially optimal profile, i.e., a profile minimizing the sum of arrival times. Then the following holds:*

$$\frac{C(P)}{C(P^*)} \leq 1 + \frac{n-1}{2\delta}.$$

PROOF. We can assume up to renaming that the global priority list is equal to $\pi = (1, \ldots, n)$. Let $P = (P_1, \ldots, P_n)$ be a pure Nash equilibrium, where every P_i is a greedy best response to P_{-i}. Let P'_i be a shortest s_i-t_i-path with respect to the travel times τ_e for every player i, i.e., $\sum_{e \in P'_i} \tau_e = \text{dist}(i)$. Let P^* be a profile minimizing the sum of arrival times. By definition of the pure Nash equilibrium, we get the following bound on the inefficiency:

$$\frac{C(P)}{C(P^*)} \leq \frac{\sum_{i \in N} C_i(P)}{\sum_{i \in N} \text{dist}(i)} \leq \frac{\sum_{i \in N} C_i(P_{-i}, P'_i)}{\sum_{i \in N} \text{dist}(i)}$$
$$= \frac{\sum_{i \in N} \sum_{e \in P'_i} \left(\tau_e + w_{i,e}(P_{-i}, P'_i)\right)}{\sum_{i \in N} \text{dist}(i)} = 1 + \frac{\sum_{i \in N} \sum_{e \in P'_i} w_{i,e}(P_{-i}, P'_i)}{\sum_{i \in N} \text{dist}(i)}.$$

We claim that the waiting time of player i is bounded by $i - 1$, which we prove later. This yields

$$\frac{C(P)}{C(P^*)} \leq 1 + \frac{\sum_{i \in N} \sum_{e \in P'_i} w_{i,e}(P_{-i}, P'_i)}{n\delta} \leq 1 + \frac{\sum_{i \in N} (i-1)}{n\delta} = 1 + \frac{n-1}{2\delta}.$$

It remains to show that the waiting time of player i on P'_i in the profile (P_{-i}, P'_i) is in fact bounded by $i - 1$.

For player i and any player $j < i$, we introduce the notion of *blocking times* $\beta_i[j]$. We define $\beta_i[j]$ to be the collection of points in time such that $t \in \beta_i[j]$ if and only if player i cannot be sent into path P'_i at time t without being delayed in the profile $(P_1, \ldots, P_j, P'_i)$. This means we restrict the game to the players $\{1, \ldots, j, i\}$.

Note that, if $|\beta_i[i-1]| \leq i - 1$, then there is a number $k \in \{0, \ldots, i-1\}$ that is no blocking time, i.e., $k \notin \beta_i[i-1]$. This means player i could be sent into P'_i at time k without being delayed by the players in $\{1, \ldots, i-1\}$. Since players $\{i+1, \ldots, n\}$ have a lower priority than player i, they can influence the strategy and waiting time of neither player i nor of players in $\{1, \ldots, i-1\}$. Since player i is starting at time 0 and his or her chosen path is free at time k, he or she waits at most $k \leq i - 1$ time units. It remains to show that $|\beta_i[i-1]| \leq i - 1$.

By induction on j, we prove $|\beta_i[j]| \leq j$ for all $j \in \{0, \ldots, i-1\}$.

Case $j = 0$. Certainly, for $j = 0$, player i will never be blocked, since i is the only player in the restricted game. Thus, $|\beta_i[0]| = 0$.

Step $j - 1 \rightarrow j$. For contradiction, assume $|\beta_i[j]| > j$. Since $|\beta_i[j-1]| \leq j - 1$ by induction hypothesis, the introduction of player j to the game adds at least two new blocking times for player i. Remember that j has a lower priority than players $\{1, \ldots, j-1\}$. A new blocking time k means that player i sent into path P_i' at time k would collide with player j under $(P_1, \ldots, P_j, P_i')$. Now consider the lowest blocking time $k \in \beta_i[j] \setminus \beta_i[j-1]$. Since $k \notin \beta_i[j-1]$, player j passes through edges in $P_j \cap P_i'$ without being delayed due to the fact that P_j is a greedy best response to P_{-j} and P_i' is a shortest path with respect to τ_e. That contradicts that j induces a second blocking time, which finishes the proof. □

The following corollary follows immediately due to the fact that the described profile P is a pure Nash equilibrium.

Corollary 3.5. *The price of stability in a competitive packet routing game with a global priority list is upper bounded by* $1 + \frac{n-1}{2\delta}$.

We can derive a refined upper bound on the PoS depending on $u_{\min}$ of $1 + \frac{n-1}{2\delta u_{\min}}$. We adapt the induction above to prove $|\beta_i[j]| < \frac{j}{u_{\min}}$. Therefore, we do the base cases $j = 0, \ldots, u_{\min} - 1$ and afterward the induction step $j \rightarrow j + u_{\min}$. Additionally to the arguments above, we use the fact that $u_{\min}$ players are necessary to block an edge at any time step. The bound for $|\beta_i[j]|$ directly implies the refined upper bound.

In the following, we show that the price of anarchy of multicommodity competitive packet routing games is upper bounded by $\min\{n, 1 + \frac{(n-1)\Delta}{2\delta}\}$. Note that it is not possible to use the technique from above to bound the PoA, because the profile that contains the greedy best responses is only an upper bound for a PNE with the lowest cost. So to bound the social cost for every possible equilibrium, we need a different approach.

Proposition 3.6. *The price of anarchy in every multicommodity competitive packet routing game with a global priority list is upper bounded by*

$$\min\left\{n, 1 + \frac{(n-1)\Delta}{2\delta}\right\}.$$

Proof. We assume w.l.o.g. that the players are numbered according to the priority list. We start by estimating the arrival time of player i. Every player can always choose a shortest s_i-t_i-path. Observe that the waiting time of player i is upper bounded by the maximum completion time of players $1, \ldots, i-1$. So the arrival time is at most

$$C(i) \leq \max\{C(1), C(2), \ldots, C(i-1)\} + \text{dist}(i).$$

With this observation, we can prove the following claim.

Claim. *The completion time $C(i)$ of player i is upper bounded by $C(i) \leq \sum_{j=1}^{i}$ dist(j).*

We prove this claim by induction.

Case $i = 1$. The first player is never delayed by any other player, so he or she can choose a shortest path and $C(1) = \mathrm{dist}(1)$.

Step $i \to i+1$. Due to the argumentation above, the following holds:

$$\begin{aligned} C(i+1) &\le \max\{C(1), C(2), \ldots, C(i)\} + \mathrm{dist}(i+1) \\ &\le \max\left\{\sum_{j=1}^{1}\mathrm{dist}(j), \sum_{j=1}^{2}\mathrm{dist}(j), \ldots, \sum_{j=1}^{i}\mathrm{dist}(j)\right\} + \mathrm{dist}(i+1) \\ &\le \sum_{j=1}^{i}\mathrm{dist}(j) + \mathrm{dist}(i+1) = \sum_{j=1}^{i+1}\mathrm{dist}(j), \end{aligned}$$

and the claim follows. In a PNE profile $P = (P_1, \ldots, P_n)$, the sum of completion times is upper bounded by

$$C(P) = \sum_{i=1}^{n} C(i) \le \sum_{i=1}^{n}\sum_{j=1}^{i}\mathrm{dist}(j) = \sum_{i=1}^{n}(n-i+1)\mathrm{dist}(i).$$

We can estimate the cost of the optimal solution by $C(OPT) \ge \sum_{i=1}^{n}\mathrm{dist}(i)$. So the price of anarchy is upper bounded by

$$\mathrm{PoA} \le \frac{\sum_{i=1}^{n}(n-i+1)\mathrm{dist}(i)}{\sum_{i=1}^{n}\mathrm{dist}(i)}.$$

There are two possibilities to bound this expression. It depends on the parameters n, Δ, and δ which of the two is smaller. We have

$$\begin{aligned} \frac{\sum_{i=1}^{n}(n-i+1)\mathrm{dist}(i)}{\sum_{i=1}^{n}\mathrm{dist}(i)} &= 1 + \frac{\sum_{i=1}^{n}(n-i)\mathrm{dist}(i)}{\sum_{i=1}^{n}\mathrm{dist}(i)} \\ &\le 1 + \frac{\sum_{i=1}^{n}(n-i)\Delta}{\sum_{i=1}^{n}\delta} \\ &= 1 + \frac{(n-1)\Delta}{2\delta}, \end{aligned}$$

and

$$\frac{\sum_{i=1}^{n}(n-i+1)\mathrm{dist}(i)}{\sum_{i=1}^{n}\mathrm{dist}(i)} \le \frac{\sum_{i=1}^{n} n\cdot\mathrm{dist}(i)}{\sum_{i=1}^{n}\mathrm{dist}(i)} = n,$$

which finishes the proof. □

Observe that this upper bound coincides with the bound on the PoA in single-commodity games for $\Delta = \delta = 1$. Thus, we get a tight example from the lower-bound example in Theorem 3.2 for this special case. Note that, for $\Delta > 1$, the single-commodity PoA bound $1 + \frac{n-1}{2\Delta}$ as well as the multicommodity PoS bound $1 + \frac{n-1}{2\delta}$ is strictly smaller than the multicommodity PoA bound $\min\{n, 1 + \frac{(n-1)\Delta}{2\delta}\}$.

The refined upper bound, assuming $\frac{n}{u_{\min}}$ integral, of the PoA is

$$\min\left\{\frac{n}{u_{\min}}, 1 + \frac{(n-1)\Delta}{2\delta u_{\min}}\right\}.$$

We can achieve this upper bound by changing the proof above as follows. We form subsets S_l of players of cardinality $u_{\min}$. That is, subset S_l consists of the players $\{(l-1)\cdot u_{\min}+1, \ldots, l\cdot u_{\min}\}$.

By using $u_{\min}$, we can upper bound the completion time of a player $i \in S_l$ by

$$C(i) \leq \left(\frac{n}{u_{\min}} - l + 1\right) \operatorname{dist}(i).$$

Here, we use the idea that a player k in S_l can traverse every edge of his or her shortest path without being delayed after at least k players of S_{l-1} arrived at their sink. Using this bound, we derive

$$\text{PoA} \leq \frac{\sum_{i=1}^{n} (\frac{n-i}{u_{\min}} + 1)\operatorname{dist}(i)}{\sum_{i=1}^{n} \operatorname{dist}(i)} \leq \min\left\{\frac{n}{u_{\min}}, 1 + \frac{(n-1)\Delta}{2\delta u_{\min}}\right\},$$

and get the claimed bound.

3.2 Local Priority Lists

All results presented so far deal with global priority lists. For local priority lists, we derive the following upper bound on the price of anarchy.

Theorem 3.7. *In instances in which pure Nash equilibria exist, the price of anarchy is upper bounded by* $6\tilde{D}^2 - \tilde{D}$ *for all priority lists, where*

$$\tilde{D} = \max_{i \in N} \max_{P_j \in \mathcal{P}_i} \sum_{e \in P_j} \max\{\tau_e, 1\}.$$

In a related model of competitive routing games, Kulkarni and Mirrokni [17] prove a bound on the robust price of anarchy of $4D^2$, where D denotes the length of the longest feasible path of any player. For the differences in the two models, see Section 1.2 and note that $\delta \leq \Delta \leq D \leq \tilde{D}$. Observe that an instance of a competitive packet routing game, in the special case where no 0 travel times exist, can be fit into the model defined by Kulkarni and Mirrokni by replacing any edge of length τ_e by τ_e edges of length 1. This is necessary due to the different capacity constraints. Note that, for the case without 0 travel times, the values $\tilde{D}$ and D coincide.

In contrast to [17], our result holds for *arbitrary* local priority lists, whereas the result of [17] applies only to the *Highest-Density-First* rule.

Proof. We start with the following preprocessing in order to ensure binary travel times and unit capacities. For a given instance, we substitute each edge $e \in E$ of travel time $\tau_e > 1$ by τ_e-many edges of length 1, each with the same priority lists π_e. This results in an instance with travel times $\tau_e \in \{0, 1\}$. The roadmap of the proof is as follows. We will define a primal-dual LP pair with primal feasible solution x, objective value $C_{\text{primal}}(x)$, and dual feasible solution (α, β, ν) with objective function value $C_{\text{dual}}(\alpha, \beta, \nu)$. We introduce some coefficients $c_1 = 2$ and $c_2 = 3\tilde{D}^2 - \frac{\tilde{D}}{2}$ such that

$$C(\bar{P}) \leq c_1 C_{\text{dual}}(\alpha, \beta, \nu) \leq c_1 C_{\text{primal}}(x) \leq c_1 c_2 C(OPT),$$

where $\bar{P}$ denotes any pure Nash equilibrium, x and (α, β, ν) are special primal and dual solutions defined as below, and OPT denotes a social optimal solution.

We start by defining the primal linear program and construct a feasible solution x with

$$C_{\text{primal}}(x) \leq \left(3\tilde{D}^2 - \frac{\tilde{D}}{2}\right) C(OPT).$$

The primal LP is defined as follows:

$$\begin{aligned}
\min \sum_{i\in N}\sum_{P_j\in\mathcal{P}_i}\sum_{e\in P_j}\sum_{\theta\in\mathbb{Z}_{\geq 0}} & x_{eij\theta}\cdot(\theta+\tau_e) & \\
\text{s.t.} \quad \sum_{P_j\in\mathcal{P}_i} x_{ij} &\geq 1 & \forall i\in N\\
\sum_{\theta\in\mathbb{Z}_{\geq 0}} x_{eij\theta} &\geq x_{ij} & \forall e\in E, i\in N, j: P_j\in\mathcal{P}_i\\
\sum_{i\in N}\sum_{P_j\in\mathcal{P}_i} x_{eij\theta} &\leq \frac{1}{2\tilde{D}} & \forall e\in E, \theta\in\mathbb{Z}_{\geq 0}\\
x_{ij}, x_{eij\theta} &\geq 0 & \forall e\in E, \theta\in\mathbb{Z}_{\geq 0}, i\in N, j: P_j\in\mathcal{P}_i.
\end{aligned}$$

We claim that, given any socially optimal solution to a competitive packet routing instance, we can construct a feasible LP-solution x with objective value $C_{\text{primal}}(x) \leq (3\tilde{D}^2 - \frac{\tilde{D}}{2})C(OPT)$. Given a socially optimal profile $OPT = P = (P_1, \ldots, P_n)$ with arrival times $C_i(P), i \in N$, assign $x_{ij} = 1$ if player i chooses path $P_j \in \mathcal{P}_i$, and $x_{ij} = 0$ otherwise. This trivially fulfills the first inequality. Furthermore, assign $x_{eij(2\tilde{D}\theta)} = x_{eij(2\tilde{D}\theta+1)} = \cdots = x_{eij(2\tilde{D}\theta+2\tilde{D}-1)} = \frac{1}{2\tilde{D}}$ if player i enters edge e on his or her selected path P_j at time step θ, and 0 otherwise. It is easy to check that this is a feasible solution for the LP. For bounding the objective function, we observe that setting $x_{eij(2\tilde{D}\theta)} = x_{eij(2\tilde{D}\theta+1)} = \cdots = x_{eij(2\tilde{D}\theta+2\tilde{D}-1)} = \frac{1}{2\tilde{D}}$ results in an objective value of

$$\sum_{k=0}^{2\tilde{D}-1} \frac{1}{2\tilde{D}} \cdot (2\tilde{D}\theta + k + \tau_e) = 2\tilde{D}\theta + \tau_e + \frac{1}{2\tilde{D}} \sum_{k=0}^{2\tilde{D}-1} k = 2\tilde{D}\theta + \tau_e + \frac{2\tilde{D}-1}{2}$$

for an edge e that is entered at time step θ by player i. Let $E_i(P_j, \theta)$ denote the set of edges player i enters on his or her selected path P_j at time step θ. We have that the total objective value of x is

$$\begin{aligned}
&\sum_{i\in N}\sum_{\theta\in\mathbb{Z}_{\geq 0}}\sum_{e\in E_i(P_j,\theta)} \left(2\tilde{D}\theta + \tau_e + \frac{2\tilde{D}-1}{2}\right)\\
&\leq \sum_{i\in N}\sum_{\theta\in\mathbb{Z}_{\geq 0}}\sum_{e\in E_i(P_j,\theta)} \left(2\tilde{D}(\theta + \tau_e) + \frac{2\tilde{D}-1}{2}\right)\\
&\leq \sum_{i\in N}\sum_{\theta\in\mathbb{Z}_{\geq 0}}\sum_{e\in E_i(P_j,\theta)} \left(2\tilde{D}(C_i(P)) + \frac{2\tilde{D}-1}{2}\right)\\
&\leq \sum_{i\in N} \left(\tilde{D}\cdot\left(2\tilde{D}(C_i(P)) + \frac{2\tilde{D}-1}{2}\right)\right)\\
&\leq \sum_{i\in N} \left(\tilde{D}\cdot\left(2\tilde{D}(C_i(P)) + \frac{2\tilde{D}-1}{2}(C_i(P))\right)\right)\\
&\leq \left(3\tilde{D}^2 - \frac{\tilde{D}}{2}\right)\sum_{i\in N} C_i(P) = \left(3\tilde{D}^2 - \frac{\tilde{D}}{2}\right) C(OPT).
\end{aligned}$$

In the remainder of the proof, we show that even the worst PNE has a social cost of at most $2 \cdot C_{\text{dual}}(\alpha, \beta, \nu)$ for a dual feasible solution (α, β, ν) constructed as described below, which yields the desired bound of $6\tilde{D}^2 - \tilde{D}$ on the price of anarchy. Consider the dual LP

$$\begin{aligned} \max \sum_{i\in N} \alpha_i - \frac{1}{2\tilde{D}} \sum_{e\in E} \sum_{\theta\in\mathbb{Z}_{\geq 0}} \beta_{e\theta} & \\ \text{s.t. } \alpha_i - \sum_{e\in P_j} \nu_{eij} \leq 0 & \qquad \forall i \in N, j : P_j \in \mathcal{P}_i \\ \nu_{eij} - \beta_{e\theta} \leq \theta + \tau_e & \qquad \forall e \in E, i \in N, j : P_j \in \mathcal{P}_i, \theta \in \mathbb{Z}_{\geq 0} \\ \alpha_i, \beta_{e\theta}, \nu_{eij} \geq 0 & \qquad \forall e \in E, i \in N, j : P_j \in \mathcal{P}_i, \theta \in \mathbb{Z}_{\geq 0}. \end{aligned}$$

By weak linear programming duality, we know that any feasible dual solution (α, β, ν) has objective value at most $C_{\text{primal}}(x)$. It suffices to show that any pure Nash equilibrium $\bar{P}$ induces a feasible dual solution with objective value at least $\frac{1}{2}\sum_{i\in N} C_i(\bar{P})$.

Let $\bar{P} = (\bar{P}_1, \ldots, \bar{P}_n)$ be any PNE. Define a dual solution (α, β, ν) as follows. For each $i \in N$, let $\alpha_i = C_i(\bar{P})$. Furthermore, for each $\theta \in \mathbb{Z}_{\geq 0}$ and $e \in E$, let $\beta_{e\theta} = |\{i \in N \mid e \in \bar{P}_i,\ \theta \leq C_i(\bar{P})\}|$ denote the number of players who use edge e in their selected path in $\bar{P}$ and have not yet arrived at their sink at time θ. Finally, let $\nu_{eij} = \tau_e + w_{ie}(P_j, \bar{P}_{-i})$ denote the waiting time at the entry of e plus the traversing time τ_e in case player i switches from strategy $\bar{P}_i$ to P_j. Note that the first dual constraint is easily seen to be satisfied by this definition of the dual variables, since $\alpha_i - \sum_{e\in P_j} \nu_{eij} \leq 0$ is equivalent to $C_i(\bar{P}) \leq \sum_{e\in P_j}(\tau_e + w_{ie}(P_j, \bar{P}_{-i})) = C_i(P_j, \bar{P}_{-i})$, which follows by the definition of PNE. The second constraint $\nu_{eij} - \beta_{e\theta} \leq \theta + \tau_e$ can equivalently be written as $w_{ie}(P_j, \bar{P}_{-i}) \leq \theta + \beta_{e\theta}$, which is certainly satisfied, since at each time step θ, any player i, after switching from $\bar{P}_i$ to path P_j, will never wait at the entry of any edge e longer than θ plus the total number of players that have not yet arrived at their sink at time θ. It remains to show that $C(\bar{P}) \leq 2C_{\text{dual}}(\alpha, \beta, \nu)$. We get

$$\sum_{i\in N} \alpha_i - \frac{1}{2\tilde{D}} \sum_{e\in E} \sum_{\theta\in\mathbb{Z}_{\geq 0}} \beta_{e\theta} \geq \sum_{i\in N} C_i(\bar{P}) - \frac{1}{2\tilde{D}} \tilde{D} \sum_{i\in N} C_i(\bar{P}) = \frac{1}{2} C(\bar{P}),$$

which concludes the proof. □

Remark 3.8. Since our proof goes along the same lines as the primal-dual proof technique of Kulkarni and Mirrokni [17], it is not hard to verify that our results also hold for coarse-correlated equilibria, which are guaranteed to exist. This includes the case of correlated equilibria and mixed Nash equilibria; see [23].

Remark 3.9. We get an example with the price of anarchy $\frac{\tilde{D}}{4}$ from the proof of Theorem 3.2.

Remark 3.10. There is an example with PoS in $\Theta(\sqrt{\tilde{D}})$.

Proof. Analogously to Proposition 3.3, we can also bound the price of stability by $\tilde{D}$. We modify the analysis as follows. For every player b, the number of edges of the s_a-t_a-path is increased by 1. For choosing $k_a = k_{b_1} = \cdots = k_{b_b} = 1$ in this example, the total number of edges of this path is $\tilde{D} = b + 2$. If we substitute this and set $a = n - \tilde{D} + 2$ in the formula

$$\text{PoS} = 1 + \frac{a(b-1)}{a + \frac{a(a+1)}{2} + b}$$

and optimize over n, we get a price of stability that is in $\Theta(\sqrt{\tilde{D}})$. □

4 COMPUTATIONAL COMPLEXITY

We now focus on the computational complexity and present different hardness results. We start the section by showing that the problem to design either local or global priority lists to minimize the cost of a social optimum, or the cost of any pure Nash equilibrium, in the arising competitive packet routing game is APX-hard. Second, we show that the calculation of a best response or a Nash equilibrium highly depends on the chosen priority lists. It is NP-hard to decide if there is a best response with cost smaller than k and also if there exists a PNE for local priority lists. For the case of global priority lists, the calculation of a best response and a pure Nash equilibrium can be done in polynomial time. We conclude this section by showing that it is NP-hard to calculate an approximately social optimal state in multicommodity instances, even with global priority lists.

4.1 The Design of Priority Lists

THEOREM 4.1. *Even in graphs that form a tree, i.e., every player has a predefined strategy, designing priority lists that minimize the social cost or the cost of a pure Nash equilibrium is APX-hard both in the case of global or local priority lists.*

PROOF. The proof uses similar ideas as a proof of Peis, Skutella, and Wiese in [22]. Compared to their proof, we have to modify the travel times on the edges and add some additional dummy players with their corresponding subgraphs. However, the analysis is different, since we need to consider a different cost function. We seek to minimize the sum of arrival times, where in [22], the aim is to minimize the makespan.

The idea of the proof is to reduce 3-OCCURRENCE-MAX-3-SAT, which is known to be APX-hard [20], [21]. We are given an instance of 3-OCCURRENCE-MAX-3-SAT with n variables and m clauses. Each clause has exactly three variables and w.l.o.g. we assume that each variable occurs at most two times as a positive and at most two times as a negative literal.

We design the following competitive packet routing instance with unique strategies for the players. The choice of the priority lists corresponds to the assignment of the variables. The cost of the unique strategy profile in the competitive packet routing instance will be equal to $34n + 6m + \#(\text{unsatisfied clauses})$. For every variable x, we introduce four so-called variable players x_1, x_2, $\bar{x}_1$, $\bar{x}_2$, which correspond to the first and second occurrence as a positive or negative variable, respectively. All those players have a predefined path that collides with the paths of the clause players as follows. For each clause, we define three clause players. The path of the clause player corresponding to the ith positive or negative occurrence of a variable x in the 3-OCCURRENCE-MAX-3-SAT instance will meet the path of x_i or $\bar{x}_i$, respectively. Figure 6 shows the part of the graph corresponding to the variable players. The three clause players of one clause c all start at the same node s_c, use the same edge to an intermediate node n_c, and use the last edge of the path of their corresponding variable player. The path of a clause player corresponding to the first occurrence of x as a negated literal, denoted by $\bar{x}_1$, is depicted in Figure 7. In addition to that, we introduce $four$ dummy players per variable in the 3-OCCURRENCE-MAX-3-SAT instance. All these players d_i are ready to enter the last edge of player i for $i \in \{x_1, \bar{x}_1, x_2, \bar{x}_2\}$ at time 5. Their paths are introduced in Figure 8.

Now, we argue that there is a solution to 3-OCCURRENCE-MAX-3-SAT with U unsatisfied clauses if and only if there is a scheduling rule with sum of arrival times equal to $34n + 6m + U$. If there is no collision, the variable players corresponding to the same variable arrive at times 3,3,4,4; the clause players arrive at times $1, 2, 3$; and the dummy players per variable arrive at times 5,5,5,5. We conclude that the sum of arrival times is $34n + 6m$ if there is no collision.

Given a solution of 3-OCCURRENCE-MAX-3-SAT, we construct the scheduling rules as follows. For a variable x that is set to true, we define $\pi = (\bar{x}_1, \bar{x}_2, x_1, x_2)$. If x is set to false, we choose

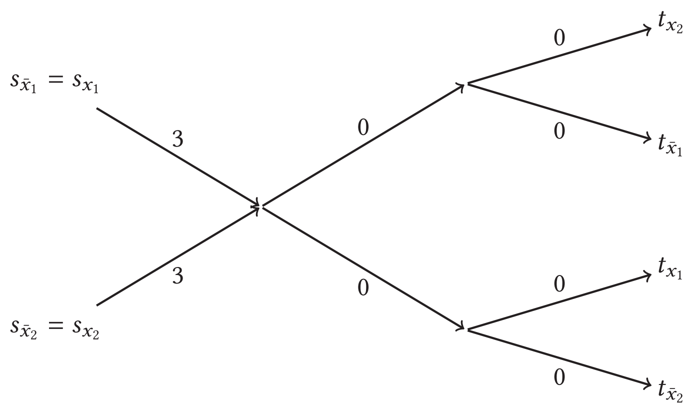

Fig. 6. The part of the graph that can be used by the variable players corresponding to a variable x.

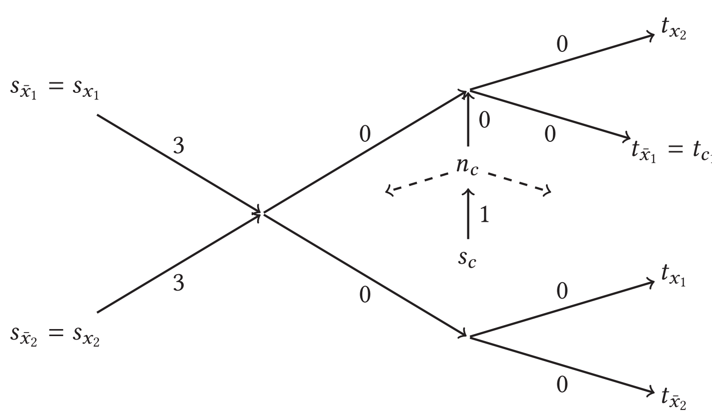

Fig. 7. We introduced the clause player for the first occurrence of $\bar{x}$ in a clause c.

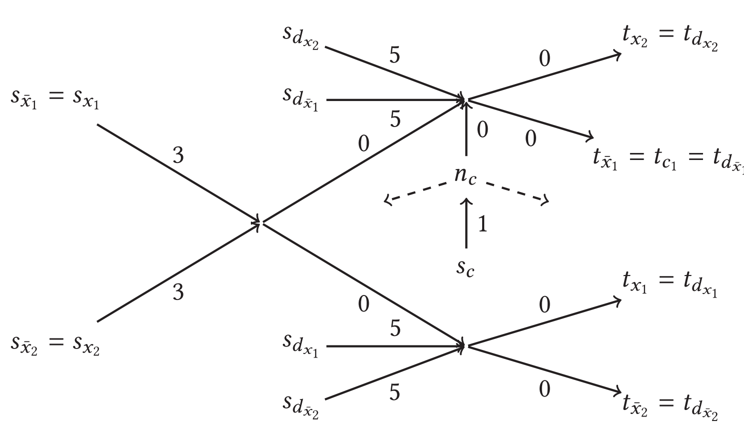

Fig. 8. We introduced the dummy players for variable x.

$\pi = (x_1, x_2, \bar{x}_1, \bar{x}_2)$; that is, the variable players x_1 and x_2 start at time 0, whereas players $\bar{x}_1$ and $\bar{x}_2$ start at time 1. For each satisfied clause, we let a clause player corresponding to one clause-satisfiable variable start last. That means he or she starts at time 2 and uses the edge to his or her sink at time 3. Since the corresponding variable player starts last, he or she will not meet the variable player there. The other two clause players cannot meet a variable player at all and the dummy players cannot collide at all. So we have the designated cost.

It remains to show that we get at most one collision per unsatisfied clause. First, note that if for all variable players in an unsatisfied clause it holds that x_1 and x_2 start at the same time, we definitely get one collision. The only other possibility is to start x_1 and $\bar{x}_2$ first and x_2 and $\bar{x}_1$ second, or the other way round. This means that we get an additional waiting time of 2 because of collisions between them and the variable players arrive at times 3,4,4,5. Thus, the last player collides with the corresponding dummy player and we have three collisions in total. Since the variables can occur at most 3 times, this cannot be cheaper. So, the cost cannot be smaller than $34n + 6m + U$.

We proceed by showing the inapproximability factor. Suppose we cannot approximate 3-OCCURRENCE-MAX-3-SAT beyond a factor of $(1 - \epsilon)$. Let U^* be the number of unsatisfied clauses in an optimal solution. We have that for all computable scheduling rules,

$$\begin{aligned}\sum C_j &\geq 34n + 6m + U \geq 34n + 7m - (1 - \epsilon)(m - U^*) \\ &\geq 34n + 6m + U^* + \epsilon\left(\frac{1}{2}m\right) = \sum C_j^* + \frac{\epsilon}{218}(109m) \\ &\geq \sum C_j^* + \frac{\epsilon}{218}(7m + 34n) \geq \sum C_j^* + \frac{\epsilon}{218}(6m + 34n + U^*) \\ &\geq \left(1 + \frac{\epsilon}{218}\right)\sum C_j^*,\end{aligned}$$

since $n \leq 3m$, which completes the proof. □

4.2 Computing Best Responses and Pure Nash Equilibria

For computing best responses and pure Nash equilibria, the complexity status highly depends on the chosen priority list.

PROPOSITION 4.2. *In competitive packet routing games with local priority lists, it is NP-hard to decide if there is a best response for a player with cost smaller than k even in the single-commodity setting. Moreover, given a game with local priority lists, it is NP-hard to decide if there is a pure Nash equilibrium.*

The proof of this proposition is based on a proof of [12]. The difference is that Hoefer et al. use the FIFO policy with a global tie-breaking rule, where we use local priority lists. Their model allows one to schedule a player with lower priority before another player by slight perturbation of the travel times. This does not work in our model due to the integral time steps and integral travel times, so we need to use local priority lists. In order to show the hardness for computing pure Nash equilibria, we had to modify the graph and introduce an exclusive start node for every player and a gadget with an additional player, which leads to the nonexistence of a pure Nash equilibrium, if necessary.

PROOF. We start by proving that it is NP-hard to decide if there is a best response of a player with cost smaller than k by reducing from 3-SAT.

Assume we are given a 3-SAT instance with clauses $c_1, \ldots, c_m$ and variables $x_1, \ldots, x_n$. Each clause c_j consists of three literals v_{1j}, v_{2j}, v_{3j}. A literal is a variable that is either negotiated or

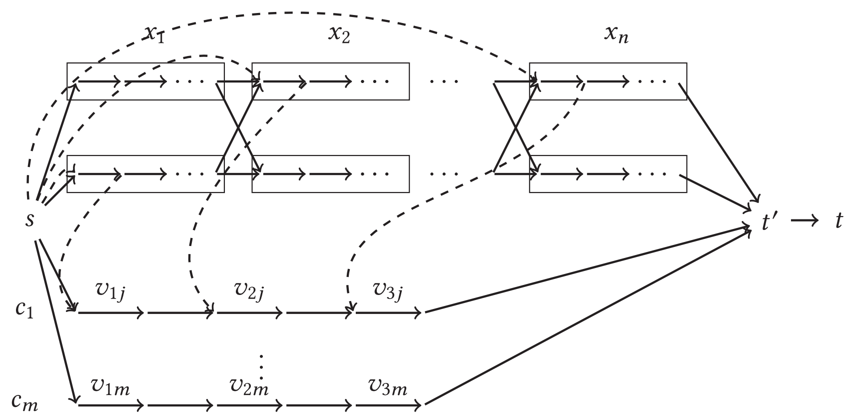

Fig. 9. Graph with the example clause $c_1 = \bar{x}_1 \vee x_2 \vee x_n$.

not. We assume that the variables in a clause are ordered by their index. We construct a single-commodity competitive packet routing game, such that the cost of the best response of a player shows whether the 3-SAT instance is satisfiable. Additionally, the best response gives us the assignment of the variables.

Consider the graph depicted in Figure 9. It contains two rows of n blocks, with $m + 1$ nodes per block. Each block corresponds to a variable and each edge in a block corresponds to a clause. The idea is as follows. If a later-introduced decider player uses a block of the upper row in his or her best response, it means that the corresponding variable is set to true and to false for the second row. All travel times and capacities of the edges in the blocks, the edge from s to the blocks, and the edge from t' to t are one.

Additionally, there is a path for every clause that consists of seven edges. It starts in s and goes to t'. All edges have unit capacity. The travel times of the edges depend on the variables. Let clause j contain the variables i, k, and l, ordered by index. The first edge has travel time $L_{ij} + 2$, where $L_{ij} = (m + 1)(i - 1) + j$. The second, fourth, and sixth edge have travel time 1. They correspond to the first, second, and third variable in c_j. The third edge has travel time $L_{kj} - L_{ij} - 1$, the fifth $L_{lj} - L_{kj} - 1$, and the seventh $n \cdot (m + 1) - L_{jl} - 2$.

Additionally, there are some edges for the variables. If there is a variable $x_i \in c_j$, we define an edge from s to the jth node in the ith block in the upper row. For $\bar{x_i} \in c_j$, we get such an edge to the according node in the lower row. This edge has unit capacity and travel time L_{ij}. This is exactly the number of edges from s to the corresponding node on the block path. Let x_i be at position k in c_j. Then, we introduce an edge from the $(j + 1)$-th node in block i in the corresponding row to the start node of the edge of the kth variable in the path of clause c_j. This edge has unit capacity and unit travel time.

Now, we introduce the $4m + 1$ players of the competitive packet routing game. We have a clause player P^{c_j} for every clause c_j, a variable player P^{ij} for every variable $x_i \in c_j$, and the decider player. This will be the player who chooses a best response.

Next, we fix the priority list. On the edges in the blocks, the decider player is scheduled first. On the edges of the paths of the clauses, the variable players are scheduled first and the clause players are scheduled in front of the decider player. And on the edge (t', t), the clause players are scheduled first and the decider player is scheduled in front of the variable players.

We choose the strategies for all players except for the decider player. Each clause player chooses his or her clause path. Each variable player chooses the direct edge from s to the jth node in the

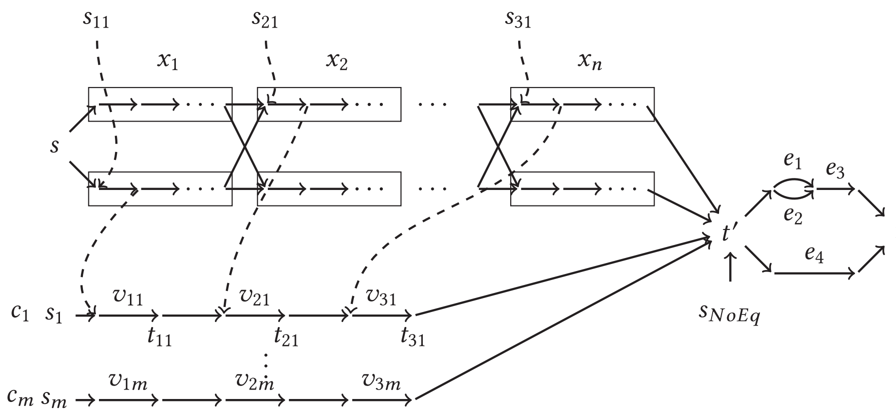

Fig. 10. Modified graph with the example clause $c_1 = \bar{x}_1 \vee x_2 \vee x_n$.

ith block in the according row, the following edge in this block, the edge that leads them to their clause, and the following clause path.

We claim that the 3-SAT problem is satisfiable if and only if there is a best response for the decider player with cost at most $n(m + 1) + 2$.

First, we observe that the decider chooses a path in the blocks. On any other path, he or she needs to wait for other players and arrives later. On the block path, he or she cannot be delayed until he or she reaches t' after $n(m + 1) + 1$ units of time because he or she is scheduled first. At (t', t), he or she could be delayed by a clause player. These players also arrive at t' after $n(m + 1) + 1$ units of time, if they are not delayed. But they can be delayed by the corresponding variable players. Thus, the decider player reaches t after $n(m + 1) + 2$ units of time if and only if all clause players are delayed. This happens if and only if there is a variable player in each clause that does not meet the decider player, which corresponds to a true literal in every clause.

In the following, we prove that also the decision, whether or not there is a PNE, is NP-hard. Consider the graph depicted in Figure 10. It is very similar to the graph above. The difference is that a gadget replaces the last edge (t', t) and we have one additional player. Furthermore, we introduce an exclusive start node for every clause and variable player. The target nodes of the variable players are at the end node of their corresponding variable edge in the clause path. Thus, the strategy of each clause and variable player is unique.

The additional player is called the NoEq player. His or her starting point is s_{NoEq} and his or her target is t. The edge (s_{NoEq}, t') has travel time $n(m + 1) + 1$, so that the NoEq player is at node t' at the same time as the decider player. In this gadget, there are some distinguished edges e_1, e_2, e_3, and e_4. All other edges in the gadget have infinite capacity and 0 travel time. The two parallel edges e_1 and e_2 have unit capacity and travel time 1. The priority list of edges e_1 and e_2 is that the decider player comes first, then the NoEq player, and at last the clause players. The edge e_3 has also unit capacity and travel time 1, and the priority list is NoEq player, decider player, and clause players. On edge e_4, we have unit capacity and unit travel time, and the priority list is as follows. The clause players have the highest priority, followed by the decider player, and the NoEq player has the lowest priority.

Additionally, we double all clause players, such that one unsatisfied clause causes a delay of two for the decider player and thus a change to the upper path in the last gadget. There, the NoEq player and the decider player will block each other and circle with their best responses as in Example 2.9, since the lower path is always more expensive.

It is obvious that computing a best response of cost $\leq k$ in the first graph is equivalent to deciding whether there is a PNE in the second graph. □

The following observation shows that we can compute best responses and a Nash equilibrium in games with global priority lists. The main idea is to use a Dijkstra-like algorithm and the fact that a player is never influenced by players of lower priority. This idea has also been used in [12]. The proofs are obtained by extending the proofs of [12] step by step to edges with 0 travel times.

Observation 4.3. *For games with global priority lists, there is a polynomial-time algorithm to compute greedy best response for any given player and thus a pure Nash equilibrium.*

Proof. Note that a polynomial-time algorithm for the computation of a best response induces a polynomial-time algorithm for the computation of a pure Nash equilibrium as follows. Without loss of generality, we assume that the global priority list is equal to $\pi = (1, \ldots, n)$. The idea is that the players choose their best response in the order of the priority list. Due to the global priority list, a player j is not influenced by the choices of the players $\{j + 1, \ldots, n\}$. By running the polynomial-time algorithm for computing a best response n times, we get a pure Nash equilibrium in polynomial time.

For computing a greedy best response of player j, we can embed players $\{1, \ldots, j - 1\}$ according to the algorithm of Proposition 2.3. No matter which strategy player j will choose, he or she can never influence their embedding, i.e., their arrival time at every intermediate node on the chosen paths. Players $\{j + 1, \ldots, n\}$ are influenced by the choice of player j, but they do not affect the choice of him or her. Thus, we can ignore them. This is why it is possible to compute a greedy best response for player j with an adapted Dijkstra algorithm. Analogously to Dijkstra's algorithm, each node has a label representing the current known shortest possible travel time from the start node to this node. The distance to an adjacent node is the travel time of the connecting edge plus the waiting time for this edge at the given time. It is possible to compute the waiting time since we can embed all the relevant players with the algorithm of Proposition 2.3. We can use a slightly adapted version of the well-known Dijkstra algorithm to compute a greedy best response, since the output of Dijkstra's algorithm has the property that every subpath is also a shortest path. □

4.3 Minimizing the Social Cost

We show that for given (even global) priority lists, there exists no polynomial-time approximation scheme for computing a strategy profile that minimizes the social cost, unless P = NP. The proposition follows by a reduction to the disjoint paths problem in directed acyclic graphs, considered by Tholey [27].

Theorem 4.4. *Given a multicommodity instance with arbitrary priority lists and n players, there is no polynomial-time algorithm approximating a social optimum better than a factor of* $(1 + \frac{1}{n})$*, unless P = NP.*

Proof. The proposition follows by a reduction to the disjoint paths problem in acyclic graphs, which was shown to be NP-hard; see [27]. Assume we are given an instance of a disjoint paths problem with directed graph G, sources $s_1, \ldots, s_n$, and sinks $t_1, \ldots, t_n$. Given this graph G, we define the competitive packet routing instance on G as follows. We set the travel times τ_e to be equal to 0 and the capacities u_e to be equal to 1. We add n nodes $t'_1, \ldots, t'_n$ and n edges $(t_1, t'_1), \ldots, (t_n, t'_n)$ with capacity and travel time equal to 1. We are given n players $i \in \{1, \ldots, n\}$, where player i starts at s_i and travels to t'_i. This is a feasible instance of a competitive packet routing game since the given graph was acyclic. Now, a social optimum has a value of n if and only if there are disjoint paths connecting the respective source and sink nodes; otherwise, we get an objective value of at least $n + 1$. □

ACKNOWLEDGMENTS

We thank the anonymous reviewers for very helpful comments, particularly for the proof of Proposition 3.6. Moreover, we thank Michael Etscheid for his careful proofreading and fruitful discussions, especially concerning Observation 2.10 and Theorem 4.4.

A preliminary version of this paper appeared in 41st International Symposium on Mathematical Foundations of Computer Science, MFCS 2016, August 22-26, 2016 - Kraków, Poland, pages 49:1–49:14.

REFERENCES

[1] Yossi Azar, Lisa Fleischer, Kamal Jain, Vahab S. Mirrokni, and Zoya Svitkina. 2015. Optimal coordination mechanisms for unrelated machine scheduling. *Operations Research* 63, 3 (2015), 489–500.

[2] Sayan Bhattacharya, Janardhan Kulkarni, and Vahab S. Mirrokni. 2014. Coordination mechanisms for selfish routing over time on a tree. In *Automata, Languages, and Programming - 41st International Colloquium (ICALP'14), Part I.* 186–197.

[3] Vittorio Bilò. 2012. A unifying tool for bounding the quality of non-cooperative solutions in weighted congestion games. In *Approximation and Online Algorithms - 10th International Workshop (WAOA'12), Revised Selected Papers.* 215–228.

[4] Costas Busch, Malik Magdon-Ismail, Marios Mavronicolas, and Paul Spirakis. 2006. Direct routing: Algorithms and complexity. *Algorithmica* 45, 1 (2006), 45–68.

[5] Ioannis Caragiannis. 2013. Efficient coordination mechanisms for unrelated machine scheduling. *Algorithmica* 66, 3 (2013), 512–540.

[6] George Christodoulou, Elias Koutsoupias, and Akash Nanavati. 2009. Coordination mechanisms. *Theoretical Computer Science* 410, 36 (2009), 3327–3336.

[7] Richard Cole, José R. Correa, Vasilis Gkatzelis, Vahab S. Mirrokni, and Neil Olver. 2011. Inner product spaces for MinSum coordination mechanisms. In *Proceedings of the 43rd ACM Symposium on Theory of Computing (STOC'11).* 539–548.

[8] Roberto Cominetti, José R. Correa, and Omar Larré. 2011. Existence and uniqueness of equilibria for flows over time. In *Automata, Languages and Programming - 38th International Colloquium (ICALP'11), Part II.* 552–563.

[9] Edsger W. Dijkstra. 1959. A note on two problems in connexion with graphs. *Numerische Mathematik* 1, 1 (1959), 269–271.

[10] Lester R. Ford Jr. and Delbert R. Fulkerson. 1958. Constructing maximal dynamic flows from static flows. *Operations Research* 6, 3 (1958), 419–433.

[11] David Gale. 1958. Transient flows in networks. *Michigan Mathematical Journal* 6 (1958), 59–63.

[12] Martin Hoefer, Vahab S. Mirrokni, Heiko Röglin, and Shang-Hua Teng. 2011. Competitive routing over time. *Theoretical Computer Science* 412, 39 (2011), 5420–5432.

[13] Nicole Immorlica, Li Erran Li, Vahab S. Mirrokni, and Andreas S. Schulz. 2009. Coordination mechanisms for selfish scheduling. *Theoretical Computer Science* 410, 17 (2009), 1589–1598.

[14] Ronald Koch, Ebrahim Nasrabadi, and Martin Skutella. 2011. Continuous and discrete flows over time. *Mathematical Methods of Operations Research* 73, 3 (2011), 301–337.

[15] Ronald Koch, Britta Peis, Martin Skutella, and Andreas Wiese. 2009. Real-time message routing and scheduling. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, 12th International Workshop (APPROX'09), and 13th International Workshop (RANDOM'09).* 217–230.

[16] Ronald Koch and Martin Skutella. 2011. Nash equilibria and the price of anarchy for flows over time. *Theory of Computing Systems* 49, 1 (2011), 71–97.

[17] Janardhan Kulkarni and Vahab S. Mirrokni. 2015. Robust price of anarchy bounds via LP and fenchel duality. In *Proceedings of the 26th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'15).* 1030–1049.

[18] Frank T. Leighton, Bruce Maggs, and Satish Rao. 1994. Packet routing and job-shop scheduling in $\mathcal{O}$(congestion + dilation) steps. *Combinatorica* 14, 2 (1994), 167–186.

[19] Frank T. Leighton, Bruce Maggs, and Andrea W Richa. 1999. Fast algorithms for finding $\mathcal{O}$(congestion + dilation) packet routing schedules. *Combinatorica* 19, 3 (1999), 375–401.

[20] Christos H. Papadimitriou. 1994. *Computational Complexity.* Addison-Wesley.

[21] Christos H. Papadimitriou and Mihalis Yannakakis. 1991. Optimization, approximation, and complexity classes. *Journal of Computer and System Sciences* 43, 3 (1991), 425–440.

[22] Britta Peis, Martin Skutella, and Andreas Wiese. 2009. Packet routing: Complexity and algorithms. In *Approximation and Online Algorithms, 7th International Workshop (WAOA'09). Revised Papers.* 217–228.

[23] Tim Roughgarden. 2015. Intrinsic robustness of the price of anarchy. *Journal of the ACM* 62, 5 (2015), 32.
[24] Tim Roughgarden and Éva Tardos. 2002. How bad is selfish routing? *Journal of the ACM* 49, 2 (2002), 236–259.
[25] Martin Skutella. 2008. An introduction to network flows over time. In *Research Trends in Combinatorial Optimization, Bonn Workshop on Combinatorial Optimization*. 451–482.
[26] Aravind Srinivasan and Chung-Piaw Teo. 2001. A constant-factor approximation algorithm for packet routing and balancing local vs. global criteria. *SIAM Journal on Computing* 30, 6 (2001), 2051–2068.
[27] Torsten Tholey. 2005. Finding disjoint paths on directed acyclic graphs. In *Graph-Theoretic Concepts in Computer Science, 31st International Workshop (WG'05), Revised Selected Papers*. 319–330.
[28] William L. Wilkinson. 1971. An algorithm for universal maximal dynamic flows in a network. *Operations Research* 19, 7 (1971), 1602–1612.

Received February 2017; revised September 2017; accepted December 2017