Competitive Packet Routing with Priority Lists

Tobias Harks[1], Britta Peis[2], Daniel Schmand[3], and Laura Vargas Koch[4]

1 **Department of Mathematics, University of Augsburg, Germany**
  `tobias.harks@math.uni-augsburg.de`
2 **School of Business and Economics, RWTH Aachen University, Germany**
  `britta.peis@oms.rwth-aachen.de`
3 **School of Business and Economics, RWTH Aachen University, Germany**
  `daniel.schmand@oms.rwth-aachen.de`
4 **School of Business and Economics, RWTH Aachen University, Germany**
  `laura.vargas@oms.rwth-aachen.de`

──── **Abstract** ────

In competitive packet routing games, packets are routed selfishly through a network and scheduling policies at edges determine which packages are forwarded first if there is not enough capacity on an edge to forward all packages 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.

1 Introduction

A fundamental combinatorial optimization problem that has received considerable attention in the past (cf. [4, 14, 17, 18, 21, 24]) is *packet routing* in graphs. 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 vertices, the goal is to transfer all packets as fast as possible to their respective destination vertices. 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 vertex can store arbitrarily many packets, but only a limited number can enter an edge simultaneously at each discrete time step, see [17]. Applications can be found in routing models used in synchronized systems with a given clock-rate. Imagine, for example, a chip with components and wires corresponding to the nodes and edges, respectively, of the associated graph, and with a centralized clock rate for the chip given by a crystal oscillator.

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 vertex and represent the players

that route rational and 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. All players are ready to start right from the beginning (i.e., there are no release dates) and aim to minimize their respective arrival time 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, 12].

1.1   Our Contribution

In this paper, we further explore properties of selfish discrete store-and-forward packet routing with *priority based* scheduling policies. We consider local priority lists and global priority lists (that may or may not depend on the edge). We obtain the following results.

Price of Anarchy/Stability

For global priority lists and multiple source-sink instances we show that the price of stability (PoS for short) behaves as $PoS \in \Omega(\sqrt{n})$ and the price of anarchy (PoA) is $PoA \in \mathcal{O}(n^3)$, where n denotes the number of players. For global priority lists and symmetric games (that is, all packets travel from a common source to a common sink) the PoS is one, while the PoA for these games is exactly $(n+1)/2$ and this bound holds even for multiple sources and a single sink.

For general local priority lists (that is, the predefined order may be different among edges) and asymmetric multi-commodity games we derive that the PoA is in between $T/4$ and $4T^2$, where T is a kind of *dilation* of the graph, i.e., the maximal length of a path, where edges with travel time 0 contribute 1 to the path. This result is obtained via adapting the primal-dual technique introduced by Kulkarni and Mirrokni [16]. Note that the network model in [16] is different to ours in the sense that it allows for different weights and sizes of the packets. See Section 1.2 for comparison. While the result in [16] 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 correlated equilibria which are guaranteed to exist.

Computational Complexity

We turn to the question of computing *optimal* priority lists, that is, priority lists that induce best possible social optima or Nash equilibria. We show that even for the special case of tree graphs, the resulting problem 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. [21], where it is shown that the problem to compute a schedule minimizing the makespan (the latest arrival of any packet) is APX-hard. Our result implies that 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.

We finally derive several further hardness results for our model: In multi-commodity games with local priority lists it is NP-hard to compute a pure Nash equilibrium. Moreover,

it is NP-hard to compute a best response in symmetric games with local priority lists. These results are obtained by adapting the reduction described in Hoefer et al. [11] used to prove hardness in a more general setting. For global priority lists, we get an efficient Dijkstra-type algorithm for computing a best response and thus a pure Nash equilibrium.

1.2 Related work

Competitive Routing over Time

Hoefer et al. [11] 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 [11] might be rational, but zero travel times are not allowed. Furthermore, the type of capacity constraints is different: While [11] 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. Hoefer et al. analyzed four different scheduling policies: *FIFO*, *non-preemptive 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 non-existence of equilibria for the FIFO scheduling policy and the complexity of computing equilibria and best responses.

The model of Kulkarni and Mirrokni [16] is a generalization of the model of Hoefer et al. [11] 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 zero 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 [11] 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 [16]. A general framework to bound the robust price of anarchy via LP-Duality or Fenchel Duality was introduced by [16]. For the *Highest Density First* scheduling policy, they derive an upper bound of $4D^2$ for the robust price of anarchy, where D is the dilation of the graph. They also show a lower bound of $D/16$.

Flows over Time

Non-competitive 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, (non-competitive) packet routing is exactly the problem to find an integral multi-commodity flow satisfying unit demands of minimum time horizon. For an introduction to flows over time we refer to Skutella [23].

Koch and Skutella introduced in [15] a game-theoretic variant of flows over time. In their model, a continuum of players routes 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. [13] introduced discrete time

steps for a single-commodity model. One could see our work as an extension to unsplittable flow particles and multi-commodity 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 $\tau_e \in \mathbb{Z}_0^+$ denoting the time needed to traverse an edge $e \in E$. Additionally, each edge $e \in E$ is endowed with a capacity $u_e \in \mathbb{Z}^+$ 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 $\tau_e = 0$. The set of players 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 *symmetric* 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 $i, j \in N$.

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 $\theta \in \mathbb{Z}_0^+$. In our model, we are given priority lists $\pi_e : N \to \{1, \ldots, n\}$ on each $e \in E$ to resolve such conflicts: among those players seeking to traverse edge e at time $\theta \in \mathbb{Z}_0^+$, 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.

Given an instance (N, G, u, τ, π), each player $i \in N$ selects one path P_i from $\mathcal{P}_i$ with the goal to minimize the time when its packet entirely reaches its 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

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

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. (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 2.1.** Consider the graph depicted in Figure 1. We are given 3 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 zero, where 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 as well as the priority lists. Therefore, it is not possible to map each player $i \in N$ to a real-valued arrival time $C_i(P)$.
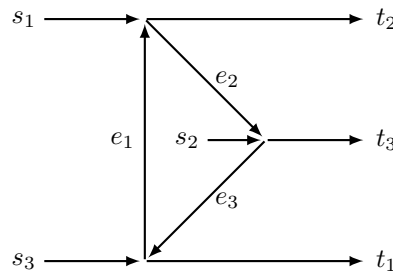
Figure 1 A graph showing that the mapping of paths to arrival times is not necessarily well-defined.

This observation motivates the exclusion of directed 0-cycles, that is, cycles C of length $\tau(C) = \sum_{e \in C} \tau_e = 0$. We assume in the following that there are no paths $P \in \mathcal{P}_i$ with total travel time equal to 0. Under this assumption, the following proposition shows that, 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.

▶ **Proposition 2.2.** *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 vertex 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 vertices sorted by the earliest arrival time of any player at that vertex. In each step, extract all vertices of minimal arrival time from the queue. Consider the graph H induced by these vertices and the corresponding edges of length $\tau_e = 0$. Repeatedly choose a vertex without incoming edges. Note that such a vertex needs to exist, since graph G is assumed to be free of 0-cycles . For each chosen vertex, route all players 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 vertex on their paths and reintroduce the vertex into the priority queue, if necessary. If the vertex is already in the priority queue, we possibly change its order in the queue. Now, delete the current vertex from H and go on with the next vertex without incoming edge. If H is empty continue with the next vertices from the queue. For a formal description of the algorithm, we refer to the full version. ◀

Using the relationship between packet routing and integral flows over time [23], we show that at least in the symmetric setting, a social optimum can be computed via Earliest Arrival Flows.

▶ **Definition 2.3.** Let $s, t \in V$. An *integral s-t-flow over time* is a set of functions $f_e : \mathbb{N} \to \mathbb{N}$ for all $e \in E$ satisfying the following two constraints:

$$f_e(\theta) \leq u_e \qquad\qquad \forall e \in E, \theta \in \mathbb{N} \qquad (2)$$

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

The first inequality constrains the capacity and the second one represents weak flow conservation. If (3) is fulfilled with equality, the flow is said to satisfy strong flow conservation. An
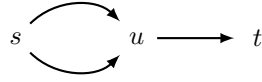
Figure 2 A network without pure Nash equilibrium.

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 2.4** (Wilkinson [25]). *An integral earliest arrival s-t-flow with strong flow conservation can be computed in pseudo-polynomial time by an adapted successive shortest path algorithm.*

Wilkinson's algorithm combines the ideas of the push-and-relabel algorithm and the successive shortest path algorithm. During the execution of the algorithm one can easily transfer an integral earliest arrival s-t flow x into a strategy profile P with the following properties. The paths P_i are cycle-free, there are no waiting times at intermediate nodes and the demands are fulfilled. We call such a path profile an *earliest arrival state.*

▶ **Lemma 2.5.** *An earliest arrival state in a symmetric network corresponds to 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 $\sum_{\theta \in \mathbb{Z}_+} \left(N_\theta^P - N_{\theta-1}^P\right) \theta \leq \sum_{\theta \in \mathbb{Z}_+} \left(N_\theta^{P'} - N_{\theta-1}^{P'}\right) \theta$. Adding $(N_{\theta-1}^P - N_\theta^{P'})\theta$ on both sides of the inequality yields

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

It follows that $\sum_{\theta \in \mathbb{Z}_+} \left(N_{\theta-1}^P - N_{\theta-1}^{P'}\right) \geq 0$. 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 an adapted successive shortest path algorithm.

▶ **Proposition 2.6.** *A social optimum of a symmetric competitive packet routing instance is computable via an adapted version of the successive shortest path algorithm even for local priority lists.*

While this proposition shows that a socially optimal profile can be computed via a successive shortest path algorithm in symmetric games, we prove in Section 4 that the computational complexity of computing the social optimum or the socially optimal Nash equilibrium becomes APX-hard in non-symmetric games, even when restricted to global priority lists.

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

■ **Table 1** The inefficiency of packet routing games with a global scheduling policy.

	Price of stability		Price of anarchy	
symmetric	1	Thm. 3.1	$\frac{n+1}{2}$	Thm. 3.2
multi-commodity	$\Omega(\sqrt{n})$	Prop. 3.3	$O(n^3)$	Prop. 3.4

▶ **Example 2.7.** Consider the network shown in Figure 2 with three vertices $V = \{s, u, t\}$, 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 an equilibrium. In this game player 1 tries to choose the same path as player 2 and player 2 always chooses the free path. It follows that already this very simple two-player symmetric (i.e., single-source-single-sink) game does not admit an equilibrium.

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 by adding players one by one according to the priority list, see Hoefer et al. [11].

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

## 3 Inefficiency of Nash equilibria

In this section, we examine the inefficiency of Nash equilibria. For the case of global priority lists, we find tight bounds for the PoS and PoA for symmetric games, a multi-commodity game with PoS in the order of $\sqrt{n}$ and an upper bound for the PoA in the multi-commodity case. 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 [16] to prove that the PoA is between $T/4$ and $4T^2$, where T is some kind of dilation of the graph.

3.1 Global Priority Lists

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

Proof. Consider a symmetric 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 $\pi = \{1, ..., n\}$. Observe that a social optimum P fulfills the earliest arrival property due to Lemma 2.5. Note that $C_i(P) \leq C_j(P)$ whenever $i \leq j$. For the sake of contradiction, suppose there exists at least one player with an improving move. Among all players with an improving move, we choose one of smallest index, say k. If k improves her strategy by switching from path P_k to P'_k, the arrival time of player k decreases, while the arrival time of players $\{1, ..., k-1\}$ stays the same. However, this is a contradiction to the fact that a socially optimal profile admits the earliest arrival property. ◀
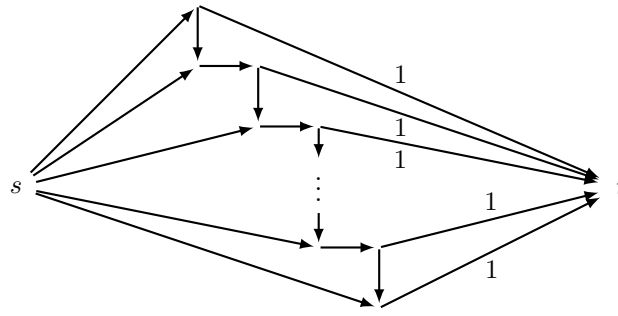
■ **Figure 3** A game with price of anarchy equal to $(n+1)/2$.

Theorem 3.1 shows that there always exists a socially optimal Nash equilibrium, as long as we restrict to global priority lists and symmetric games. In the following, we show that the price of anarchy for this case is exactly $(n+1)/2$.

▶ **Theorem 3.2.** *For all symmetric competitive packet routing games with global priority lists and n players, the price of anarchy is upper bounded by $1 + (n-1)/2k$, this bound is tight for $k = 1$.*

Proof. We first prove the upper bound. Let $S = (S_1, \ldots, S_n)$ be any Nash equilibrium. We start by showing that the cost of the Nash equilibrium S is at most

$$C(S) \leq n \cdot k + \frac{n^2}{2} - \frac{n}{2},$$

for k being the length of a shortest $s - t$-path with respect to the travel times τ_e. Since $\sum_{i=1}^{n} (k + i - 1) = n \cdot k + n^2/2 - n/2$, it suffices to show that the arrival time of the i-th player in the priority list does not exceed $k + i - 1$ time units. Suppose this is not the case, i.e. there are players arriving later. Clearly, the first player in the priority list needs k 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 has the following improvement move: she can start at source s and follow player 1. If she has to wait at the entry of any edge, she follows the player entering the edge directly ahead of her. Player j can only be delayed by players arriving on time. By using this strategy she arrives at sink t at the latest one time unit after the last player being able to delay her. Hence, she can guarantee to reach sink t by time $k + i - 1$.

Observe that no player can arrive before time unit k. So, the cost of the optimal solution can be lower bounded by nk. This yields the following bound.

$$\text{PoA} \leq \frac{C(S)}{C(\text{OPT})} \leq \frac{n \cdot k + \frac{n^2}{2} - \frac{n}{2}}{nk} = \frac{k - \frac{1}{2} + \frac{n}{2}}{k} = 1 + \frac{(n-1)}{2k},$$

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

For showing the tightness of the result, note that $1 + (n-1)/2k = (n+1)/2$ for $k = 1$. Now, consider the example depicted by the graph of Figure 3. This Braess-like graph topology has been used before to show lower bounds on the PoA in other settings, e.g. see [16]. The travel times τ_e are depicted next to the edges, where edges without 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 1. Thus, in an optimal solution, all players pick a direct path, resulting in an arrival time of 1 for each player, and an optimal social cost of n. However, the profile in which all players use the path containing all vertical edges is a Nash equilibrium as
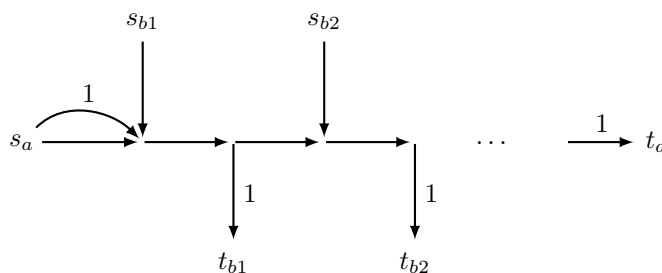
Figure 4 A game with price of stability in $\theta(\sqrt{n})$.

well. The i-th player in the order of the priority list arrives at time i and has no incentive to deviate, since there is no possible path she can use without waiting for all players of higher priority. Thus, there is a Nash equilibrium of social cost equal to $1 + 2 + \cdots + n = {n(n+1)}/{2}$. As a consequence, we obtain $\mathrm{PoA} = \frac{n(n+1)}{2}/n = {(n+1)}/{2}$, which completes the proof.   ◄

▶ **Remark.** Theorem 3.2 can be extended to networks with multiple sources and one sink.

For multi-commodity competitive packet routing games, earliest arrival flows do not necessarily exist [23]. 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 $\theta(\sqrt{n})$.

▶ **Proposition 3.3.** *There is a multi-commodity competitive packet routing game with a global priority list and price of stability in $\theta(\sqrt{n})$.*

Proof. Consider the game illustrated by the graph in Figure 4. 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 4. We consider a global priority list in which all horizontal players have priority over the vertical players.

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

$$\mathrm{PoS} = \frac{\frac{a^2}{2} + ba + b + \frac{a}{2}}{b + \frac{a^2}{2} + \frac{3a}{2}}.$$

If we replace b by $n - a$, we can differentiate the expression with respect to a. Taking the root we get $-2 + \sqrt{2n - 3}$. By substitution we get a price of stability which is in $\theta(\sqrt{n})$.   ◄

In the following proposition, we give a bound on the price of anarchy by estimating the waiting times of the players. The proof can be found in the full version.

▶ **Proposition 3.4.** *The price of anarchy in a multi-commodity competitive packet routing game with a global priority list is upper bounded by $1 + {n^3}/{2} \in O(n^3)$.*

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 3.5.** *In instances in which Nash equilibria exist, the price of anarchy is upper bounded by $4T^2$ for all priority lists, where $T = \max_{i \in N} \max_{P_j \in \mathcal{P}_i} \sum_{e \in P_j} \max\{\tau_e, 1\}$.*

In a related model of competitive routing games, Kulkarni and Mirrokni [16] prove a bound on the robust price of anarchy of $4D^2$, where D denotes the number of edges of the longest feasible path of any player. For the differences in the two models, see Section 1.2. Note 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. After this procedure, the values T and D coincide.

In contrast to [16], our result holds for *arbitrary* local priority lists, whereas the result of [16] applies only to the *Highest-Density-First*-rule. Besides, our analysis turns out to be much simpler.

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 one, each with the same priority lists $\pi(e)$. This results in an instance with travel times $\tau_e \in \{0, 1\}$. Similar, we replace each edge of capacity $u_e > 1$ by u_e-many parallel edges of unit capacity. Now, consider the following linear program.

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

We claim that the optimal LP-solution value is a T-approximation on the cost of a social optimum for the following reason. Given a socially optimal profile $P = (P_1, \ldots, P_n)$ with completion 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. Furthermore, assign $x_{eij\theta} = 1$ if player i enters edge e on her selected path P_j at time step θ, and $x_{eij\theta} = 0$ otherwise. It is easy to check that this is a feasible solution for the LP. For bounding the objective function we consider $x_{eij\theta} \cdot \tau_e$ and $x_{eij\theta} \cdot \theta$ separately. Clearly, $x_{eij\theta} \cdot \theta \leq C_i(P)$ since a player has not yet arrived at her sink if she uses edge e. By definition of T, there are at most T edges on every path P_j. Additionally we consider $\min_{e,j,\theta}\{\theta | x_{eij\theta} = 1\}$ as the starting time of player i and get

$$
\sum_{P_j \in \mathcal{P}_i} \sum_{e \in P_j} \sum_{\theta \in \mathbb{Z}_+} x_{eij\theta} \cdot \tau_e + \min_{e,j,\theta}\{\theta | x_{eij\theta} = 1\} \leq C_i(P) \quad \text{and}
$$

$$
\sum_{P_j \in \mathcal{P}_i} \sum_{e \in P_j} \sum_{\theta \in \mathbb{Z}_+} x_{eij\theta} \cdot \theta - \min_{e,j,\theta}\{\theta | x_{eij\theta} = 1\} \leq (T - 1)C_i(P).
$$

Thus, the optimal solution value of the LP is upper bounded by $\sum_{i \in N}(T-1)C_i(P)+C_i(P) = T \cdot \text{OPT}$, where $\text{OPT} = \sum_{i \in N} C_i(P)$ denotes the optimal social value of the competitive routing game.

Let us now modify the LP by dividing the right-hand side of the third constraint by $2T$. That is, we replace constraint $\sum_{i \in N} \sum_{P_j \in \mathcal{P}_i} x_{eij\theta} \leq 1$ by $\sum_{i \in N} \sum_{P_j \in \mathcal{P}_i} x_{eij\theta} \leq 1/2T$ for all $e \in E, \theta \in \mathbb{Z}_+$. Let LP^* denote the optimal objective value of the modified LP. By scaling

the $x_{eij\theta}$ variables of the LP-solution as described above by $1/2T$ and repeating the original solution $2T$ times, we achieve a feasible solution for the modified LP which loses a factor of at most $2T$ in the objective compared to the prior solution. Hence, the modified LP is a $2T^2$-approximation on the cost of a socially optimal profile. That is, $LP^* \leq 2T^2 \cdot \text{OPT}$.

In the remainder of the proof, we show that even the worst PNE has a social cost of at most $2 \cdot LP^*$ yielding the desired bound of $4T^2$ on the price of anarchy. Consider the dual of the modified LP:

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

By weak linear programming duality, we know that any feasible dual solution has objective value at most LP^*. It suffices to show that any pure Nash equilibrium $\bar{P}$ induces a feasible dual solution of objective value $1/2 \sum_{i \in N} C_i(\bar{P})$.

Let $\bar{P} = (\bar{P}_1, \ldots, \bar{P}_n)$ be any PNE. Define a dual solution $\bar{\alpha}, \bar{\beta}, \bar{\nu}$ as follows. For each $i \in N$ let $\bar{\alpha}_i = C_i(\bar{P})$. Furthermore, for each $\theta \in \mathbb{Z}_+$ and $e \in E$, let $\bar{\beta}_{e\theta} = |\{i \in N \mid e \in \bar{P}_i, \; \theta \leq C_i(\bar{P})\}|$ denote the number of players who have not yet arrived at their sink at time θ under profile $\bar{P}$. Finally, let $\bar{\nu}_{eij} = \tau_e + w_{ie}(P_j, \bar{P}_{-i})$ 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 $\bar{\alpha}_i - \sum_{e \in P_j} \bar{\nu}_{eij} \leq 0$ 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 $\bar{\nu}_{eij} - \bar{\beta}_{e\theta} \leq \theta + \tau_e$ can equivalently be written as $w_{ie}(P_j, \bar{P}_{-i}) \leq \theta + \bar{\beta}_{e\theta}$ 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 which have not yet arrived at their sink at time θ. This concludes the proof. ◀

▶ **Remark.** We get an example with price of anarchy $T/4$ from the proof of Theorem 3.2.

▶ **Remark.** Since our proof goes along the same lines as the primal-dual proof technique of Kulkarni and Mirrokni [16], 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 [22].

4   Computational Complexity

The problem to design either local or global priority lists to minimize the cost of a social optimum, or the cost of any Nash equilibrium, turns out to be APX-hard.

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

Proof (Sketch). We describe the main ideas of the proof. More details can be found in the full version. Assume we are given an instance of 3-OCCURRENCE-MAX-3-SAT, i.e. a maximum satisfiability problem with n variables, each of which occuring at most 3 times, together with m clauses where each clause contains exactly 3 variables. Inspired by [21], we define

an instance of a competitive packet routing game as follows. We design a tree graph with 8 players per variable and 3 players per clause such that the sum of the arrival times is equal to $34n + 6m + \#$(unsatisfied clauses). For each variable we introduce 4 variable-players and 4 dummy-players. The variable-players correspond to either the first or second occurrence of this variable, as a positive or negative literal, respectively. These players share an edge with one of the corresponding clause-players if the variable is the first or second occurrence of this variable as a positive or negative literal in the clause. The scheduling policies of the (fixed) starting edges of these players decode the variable assignment of the satisfiability problem. The 4 dummy-players per variable guarantee that the two players corresponding to the positive or negative occurrences leave at the same time, respectively. With this idea, we can decode the variable assignment in the satisfiability problem. We design the network and the travel times in such a way, that exactly one clause player and variable player meet, i.e. impose a waiting time of 1, if and only if the clause is not fulfilled by any variable. Thus maximizing the number of satisfied clauses in the satisfiability problem is equivalent to minimizing the social cost in the competitive routing instance. Since 3-OCCURRENCE-MAX-3-SAT is APX-hard (see [19, 20]), we derive APX-hardness for our problem. The variable-players and the topology of the graph are the same as in [21]. Due to the different objective function, we need to introduce the dummy players with their paths and conduct a different analysis. ◀

For computing best responses and 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 compute a best response for a player, even in the symmetric setting. Moreover, given a game with local priority lists, it is NP-hard to compute a Nash equilibrium if one exists.*

The proof of this proposition is based on a proof of [11]. The difference is that Hoefer et al. use the FIFO policy with a global tie breaking rule, where we use local priority policies. Their model allows 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 use local priority policies. In order to show the hardness for computing Nash equilibria we had to modify the graph and introduce an exclusive start-node for every player.

The following observation shows that we can compute best-responses and a Nash equilibrium in games with global priority policies. The main idea is to use a Dijkstra-like algorithm and the fact, that a player is never influenced by players with a lower priority. This idea has also been used in [11]. The proofs are obtained by extending the proofs of [11] step-by-step to edges with 0-travel times. We refer to the full version for further details.

▶ **Observation 4.3.** *For games with global priority policies there is a polynomial time algorithm to compute a Nash equilibrium and a best response of any given player.*

References

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

2 Sayan Bhattacharya, Janardhan Kulkarni, and Vahab S Mirrokni. Coordination mechanisms for selfish routing over time on a tree. In *Automata, Languages, and Programming – 41st International Colloquium, ICALP 2014, Copenhagen, Denmark, July 8-11, 2014, Proceedings, Part I*, pages 186–197, 2014.

3   Vittorio Bilò. A unifying tool for bounding the quality of non-cooperative solutions in weighted congestion games. In *Approximation and Online Algorithms – 10th International Workshop, WAOA 2012, Ljubljana, Slovenia, September 13-14, 2012, Revised Selected Papers*, pages 215–228, 2012.

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

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

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

7   Richard Cole, José R. Correa, Vasilis Gkatzelis, Vahab S. Mirrokni, and Neil Olver. Inner product spaces for minsum coordination mechanisms. In *Proceedings of the 43rd ACM Symposium on Theory of Computing, STOC 2011, San Jose, CA, USA, 6-8 June 2011*, pages 539–548, 2011.

8   Roberto Cominetti, José R. Correa, and Omar Larré. Existence and uniqueness of equilibria for flows over time. In *Automata, Languages and Programming – 38th International Colloquium, ICALP 2011, Zurich, Switzerland, July 4-8, 2011, Proceedings, Part II*, pages 552–563, 2011.

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

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

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

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

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

14   Ronald Koch, Britta Peis, Martin Skutella, and Andreas Wiese. Real-time message routing and scheduling. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, 12th International Workshop, APPROX 2009, and 13th International Workshop, RANDOM 2009, Berkeley, CA, USA, August 21-23, 2009. Proceedings*, pages 217–230, 2009.

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

16   Janardhan Kulkarni and Vahab S. Mirrokni. Robust price of anarchy bounds via LP and fenchel duality. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015, San Diego, CA, USA, January 4-6, 2015*, pages 1030–1049, 2015.

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

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

19   Christos H. Papadimitriou. *Computational complexity*. Addison-Wesley, 1994.

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

21   Britta Peis, Martin Skutella, and Andreas Wiese. Packet routing: Complexity and algorithms. In *Approximation and Online Algorithms, 7th International Workshop, WAOA 2009, Copenhagen, Denmark, September 10-11, 2009. Revised Papers*, pages 217–228, 2009.

22 Tim Roughgarden. Intrinsic robustness of the price of anarchy. *Journal of the ACM*, 62(5):32, 2015.

23 Martin Skutella. An introduction to network flows over time. In *Research Trends in Combinatorial Optimization, Bonn Workshop on Combinatorial Optimization, November 3-7, 2008, Bonn, Germany*, pages 451–482, 2008.

24 Aravind Srinivasan and Chung-Piaw Teo. A constant-factor approximation algorithm for packet routing and balancing local vs. global criteria. *SIAM Journal on Computing*, 30(6):2051–2068, 2001.

25 William L Wilkinson. An algorithm for universal maximal dynamic flows in a network. *Operations Research*, 19(7):1602–1612, 1971.