Improving Selfish Routing for Risk-Averse Players

Dimitris Fotakis[1], Dimitris Kalimeris[2], and Thanasis Lianeas[3]

[1] School of Electrical and Computer Engineering, National Technical University of Athens, 15780 Athens, Greece
[2] Department of Informatics and Telecommunications, National and Kapodistrian University of Athens, 15784 Athens, Greece
[3] Department of Electrical and Computer Engineering, University of Texas at Austin, Austin, TX 78701, U.S.A.
fotakis@cs.ntua.gr, sdi1000049@di.uoa.gr, tlianeas@austin.utexas.edu

Abstract. We investigate how and to which extent one can exploit risk-aversion and modify the perceived cost of the players in selfish routing so that the Price of Anarchy (PoA) wrt. the total latency is improved. The starting point is to introduce some small random perturbations to the edge latencies so that the expected latency does not change, but the perceived cost of the players increases, due to risk-aversion. We adopt the simple model of γ-modifiable routing games, a variant of selfish routing games with restricted tolls. We prove that computing the best γ-enforceable flow is $\mathbf{NP}$-hard for parallel-link networks with affine latencies and two classes of heterogeneous risk-averse players. On the positive side, we show that for parallel-link networks with heterogeneous players and for series-parallel networks with homogeneous players, there exists a nicely structured γ-enforceable flow whose PoA improves fast as γ increases. We show that the complexity of computing such a γ-enforceable flow is determined by the complexity of computing a Nash flow for the original game. Moreover, we prove that the PoA of this flow is best possible in the worst-case, in the sense that there are instances where (i) the best γ-enforceable flow has the same PoA, and (ii) considering more flexible modifications does not lead to any improvement on the PoA.

1 Introduction

Routing games provide an elegant and practically useful model of selfish resource allocation in transportation and communication networks. In the last decades, the algorithmic properties and the Price of Anarchy (PoA) of routing games, in many different settings, have been studied extensively and are well understood (see e.g., [17]). The majority of previous work assumes that the players select their routes based on precise knowledge of edge delays. In most practical applications however, the players cannot accurately predict the actual delays due to their limited knowledge about the traffic conditions and due to unpredictable events that affect the edge delays and introduce uncertainty (see e.g., [14, 12, 1, 13] for some concrete examples). Hence, the players select their routes based only on delay estimations, and most important, they are fully aware of the uncertainty and the potential inaccuracy of these estimations. Therefore, to secure themselves from increased delays, whenever this may have a considerable influence, the players select their routes taking uncertainty into account (e.g., people take a safe route or plan for a longer-than-usual delay when they head to an important meeting or to catch a long-distance flight).

Some recent work (see e.g., [12, 15, 1, 13] and the references therein) considers routing games with *stochastic delays* and *risk-averse players*, where instead of the route that minimizes her expected delay, each player selects a route that guarantees a reasonably low actual delay with a reasonably high confidence. There have been different models of stochastic routing games, each modeling the individual cost of risk-averse players in a slightly different way. In all cases, the actual delay is modeled (implicitly or explicitly) as a random variable and the perceived cost of the players is either a combination of the expectation and the standard deviation (or the variance) of their delay [12, 13] or a player-specific quantile of the delay distribution [14, 1] (see also [18, 4, 12] about the individual cost functions of risk-averse players).

No matter the precise modeling, we should expect that stochastic delays and risk-aversion can only deteriorate (or at least, cannot improve) the network performance at equilibrium (see [12, 1, 13] for upper bounds on the PoA of stochastic routing games). Interestingly, the work of [15, 13] indicates that in certain settings, stochastic delays and risk-aversion can actually improve the network performance at equilibrium. Motivated by the results of [15, 13], we consider routing games on parallel-link and series-parallel networks and investigate how one can exploit risk-aversion in order to modify the perceived cost of the (possibly heterogeneous) players so that the PoA is significantly improved.

Routing Games. To discuss our approach more precisely, we introduce the basic notation and terminology about routing games. A (non-atomic) *selfish routing game* (or instance) is a tuple $\mathcal{G} = (G(V,E), (\ell_e)_{e\in E}, r)$, where $G(V,E)$ is a directed network with a source s and a sink t, $\ell_e : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a non-decreasing delay (or latency) function associated with edge e and $r > 0$ is the traffic rate. We let $\mathcal{P}$ denote the set of simple $s-t$ paths in G. We say that G is a parallel-link network if each $s-t$ path is a single edge (or link).

A (feasible) *flow* f is a non-negative vector indexed by $\mathcal{P}$ such that $\sum_{p\in\mathcal{P}} f_p = r$. We let $f_e = \sum_{p:e\in p} f_p$ be the amount of flow that f routes on each edge e. Given a flow f, the latency of each edge e is $\ell_e(f) = \ell_e(f_e)$, the latency of each path p is $\ell_p(f) = \sum_{e\in p} \ell_e(f)$ and the latency of f is $L(f) = \max_{p:f_p>0} \ell_p(f)$.

The traffic rate r is divided among an infinite population of players, each trying to minimize her latency. A flow f is a *Nash flow*, if all traffic is routed on minimum latency paths, i.e., for any path $p \in \mathcal{P}$ with $f_p > 0$ and for all paths $p' \in \mathcal{P}$, $\ell_p(f) \leq \ell_{p'}(f)$. Therefore, in a Nash flow f, all players incur a minimum common latency equal to $\min_p \ell_p(f) = L(f)$. Under weak assumptions on the delay functions, a Nash flow exists and is essentially unique (see e.g., [17]).

The efficiency of a flow f is measured by the *total latency* $C(f)$ of the players, i.e., by $C(f) = \sum_{e\in E} f_e \ell_e(f)$. The *optimal flow*, denoted o, minimizes the total latency among all feasible flows. The *Price of Anarchy* (PoA) is the main tool for quantifying the performance degradation due to selfish behavior of the players, which leads the network to a Nash flow instead of an optimal flow. The $\mathrm{PoA}(\mathcal{G})$ of a routing game $\mathcal{G}$ is the ratio $C(f)/C(o)$ of the total latency of the Nash flow f to that of the optimal flow o of $\mathcal{G}$. The PoA of a class of routing games is defined as the maximum PoA over all games in the class. For routing games with latency functions in a class $\mathcal{D}$, the PoA is equal to $\mathrm{PoA}(\mathcal{D}) = \rho(\mathcal{D}) = (1-\beta(\mathcal{D}))^{-1}$, where $\beta(\mathcal{D}) = \sup_{\ell\in\mathcal{D}, x\geq y\geq 0} \frac{y(l(x)-l(y))}{xl(x)}$ and the parameters β and ρ only depend on the class of latency functions $\mathcal{D}$ [17, 3]. For example, ρ is $4/3$ for linear and $\frac{27+6\sqrt{3}}{23}$ for quadratic latencies.

Using Risk-Aversion to Modify Edge Latencies. The starting point of our work is that in some practical applications, we may intentionally introduce variance to edge delays so that the expected delay does not change, but the risk-averse cost of the players increases. E.g., in a transportation network, we can randomly increase or decrease the proportion of time allocated to the green traffic light for short periods, or we can open or close an auxiliary traffic lane. In a communication network, we might randomly increase or decrease the link capacity allocated to a particular type of traffic or change its priority. At the intuitive level, we expect that the effect of such random changes to risk-averse players is similar to that of refundable tolls (see e.g., [6, 11]), albeit restricted in magnitude due to the bounded variance in edge delays that we can afford.

More specifically, let us consider an edge e with latency function $\ell_e(x)$ where we can increase the latency temporarily up to $(1+\alpha_1)\ell_e(x)$ and decrease it temporarily (and for relatively short time intervals) up to $(1-\alpha_2)\ell_e(x)$. If we implement the former change with probability p_1 and the latter with probability $p_2 < 1-p_1$ (the probabilities here essentially correspond to proportions of time in which e operates in each state), the latency function of e in a given time step is a random variable $d_e(x)$ with expectation:

$$\mathbb{E}[d_e(x)] = [p_1(1+\alpha_1) + p_2(1-\alpha_2) + (1-p_1-p_2)]\ell_e(x)$$

Adjusting p_1 and p_2 (and possibly α_1 and α_2) so that $p_1\alpha_1 = p_2\alpha_2$, we have $\mathbb{E}[d_e(x)] = \ell_e(x)$, i.e., for any given flow, the expected delay through e does not change. On the other hand, if the players are (homogeneously) risk-averse and their perceived delay is given by an $(1-p_1+\varepsilon)$-quantile of the delay distribution (e.g., as in [14, 1]), for some $\varepsilon > 0$, the latency perceived by the players on e is $(1+\alpha_1)\ell_e(x)$. Similarly, if the individual cost of the risk-averse players are given by the expectation plus the standard deviation of the delay distribution (e.g., as in [12]), the latency perceived by the players on e is $(1+\sqrt{p_1\alpha_1^2+p_2\alpha_2^2})\ell_e(x)$. In both cases, we can have a significant increase in the delay perceived by risk-averse players on e, while the expected delay remains unchanged. A similar result could be achieved with any delay distribution on e (possibly more sophisticated and with larger support), as long as its expectation is $\ell_e(x)$.

In most practical situations, the feasible changes in the latency functions are bounded (and relatively small). The same is particularly true for the proportion of time in which an edge can operate in an "abnormal" state of increased or decreased delay. Combined with the particular form of risk-averse individual cost, these factors determine an upper bound γ_e on the multiplicative increase[4] of the delay on each edge e. Moreover, the players may evaluate risk differently and be *heterogeneous* wrt. their risk-aversion factors. So, in general, the traffic rate r is partitioned into k risk-averse classes, where each class i consists of the players with risk-averse factor a^i and includes a traffic rate r^i. If we implement a multiplicative increase γ_e on the perceived latency of each edge e, the players in class i have perceived cost $(1 + a^i\gamma_e)\ell_e(f)$ on each e and $\sum_{e\in p}(1 + a^i\gamma_e)\ell_e(f)$ on each path[5] p. In the special case where the players are *homogeneous* wrt. their risk-aversion, there is a single class of players with traffic rate r and risk-averse factor $a = 1$.

Contribution. In this work, we assume a given upper bound γ on the maximum increase in the latency functions and refer to the corresponding routing game as a *γ-modifiable game*. We consider both homogeneous and heterogeneous risk-averse players. We adopt this model as a simple and general abstraction of how one can exploit risk-aversion to improve the PoA of routing games on parallel-link and series-parallel networks. Technically, our model is a variant of restricted refundable tolls considered in [9, 2] for homogeneous players and in [10] for heterogeneous players. However, on the conceptual side and to the best of our knowledge, this is the first time that risk-aversion is proposed as a means of implementing restricted tolls, and through this, as a potential remedy to the inefficiency of selfish routing.

We say that a flow f is *γ-enforceable* if there is γ_e-modification on each edge e, with $0 \leq \gamma_e \leq \gamma$, so that f is a Nash flow of the modified game, i.e., for each player class i, for every path p used by class i, and for all paths p', $\sum_{e\in p}(1 + a^i\gamma_e)\ell_e(f) \leq \sum_{e\in p'}(1 + a^i\gamma_e)\ell_e(f)$. In this work, we are interested in computing either the best γ-enforceable flow, which minimizes total latency among all γ-enforceable flows, or a γ-enforceable flow with low PoA.

In Section 3, we consider routing games on parallel links with homogeneous risk-averse players and show that for every $\gamma > 0$, there exists a nicely structured γ-enforceable flow whose PoA improves significantly as γ increases and is essentially best possible in the worst-case. More specifically, based on a careful rerouting procedure, we show that given an optimal flow o, we can find a γ-enforceable flow f (along with the corresponding γ-modification) that "mimics" o in the sense that if $f_e < o_e$, e gets a 0-modification, while if $f_e > o_e$, e gets a γ-modification (Lemma 1). It is interesting that the best γ-enforceable flow, which for parallel-link games with homogeneous players reduces to the solution of a convex program [2, Algorithm 1], may not have these properties. Moreover, to the best of our knowledge, this is the first time that the existence of such close-to-optimal γ-enforceable flows has been studied in the restricted tolls literature. The proof of Lemma 1 implies that given an optimal flow o, we can compute such a γ-enforceable flow f and the corresponding γ-modification in time $O(|E|\,T_{\mathrm{NE}})$, where T_{NE} is the time for computing a Nash flow of the original parallel-link instance. Generalizing the variational inequality approach of [3], similarly to [2, Section 4], we prove (Theorem 1) that the PoA of the γ-enforceable flow f constructed in the proof of Lemma 1 is at most $\max\{1, (1 - \beta_\gamma(\mathcal{D}))^{-1}\}$, where $\mathcal{D}$ is the class of latency functions in the original game and

$$\beta_\gamma(\mathcal{D}) = \sup_{\ell\in\mathcal{D}, x\geq y\geq 0} \frac{y(\ell(x) - \ell(y)) - \gamma(x - y)\ell(x)}{x\ell(x)}$$

[4] Despite our brief discussion about how such an upper bound γ_e can be determined, we deliberately avoid getting into the details of how γ_e's are calculated. This depends crucially (and not always in a simple way) on the particular practical application and cannot be incorporated into a theoretical model.

[5] To simplify the model and make it easily applicable to general networks, we assume that the latency modifications (and the resulting individual costs of the players) are separable. This is a relatively standard simplifying assumption (see e.g., [15, 13]) on the structure of risk-averse individual costs in networks and only affects the extension of our results to series-parallel networks.

is a natural generalization of the quantity $\beta(\mathcal{D})$ introduced in [3]. For example, our analysis implies that for affine latencies, the PoA of the γ-enforceable f is at most $\max\{1, (1-(1-\gamma)^2/4)^{-1}\}$ (Corollary 2), which is significantly less that $4/3$ even for small values of γ (e.g., it is less than $6/5$ for $\gamma = 0.1$). We also show that the PoA of such γ-enforceable flows is best possible in the worst-case for γ-modifiable games with latency functions in class $\mathcal{D}$. Specifically, we present a class of parallel-link instances with homogeneous risk-averse players where the PoA of the best γ-enforceable flow is $\max\{1, (1-\beta_\gamma(\mathcal{D}))^{-1}\}$ (Theorem 2).

In Section 4, we switch to parallel-link games with heterogenous risk-averse players. Interestingly, we show that computing the best γ-enforceable flow is $\mathbf{NP}$-hard for parallel-link games with affine latencies and two classes of heterogeneous players (Theorem 3). The proof modifies the construction in [16, Section 6], which shows that the best Stackelberg modification of parallel-link instances is $\mathbf{NP}$-hard. Our result significantly strengthens [10, Theorem 1], which establishes $\mathbf{NP}$-hardness of best restricted tolls in general $s-t$ networks with affine latencies. On the positive side, we show (Theorem 5) that the γ-enforceable flow f of Lemma 1 can be easily turned into a γ-enforceable flow for parallel-link instances with heterogeneous risk-averse players by applying [10, Algorithm 1]. Since only the γ-modifications are adjusted for heterogeneous players, but the flow itself does not change, the PoA of f is bounded as above (assuming that the minimum risk-averse factor is 1) and remains best possible in the worst case.

In Section 5, we extend our approach of finding an efficiently computable γ-enforceable flow that "mimics" the optimal flow to series-parallel networks. Series-parallel networks have received considerable attention in the literature of refundable tolls, see e.g., [5, 7], but to the best of our knowledge, they have not been explicitly considered in the setting of restricted tolls. We extend the rerouting procedure of Lemma 1 and combine it with a continuity argument for γ-enforceable Nash flows in series-parallel networks. Hence, we show that for routing games in series-parallel networks with homogeneous players, there is a γ-enforceable flow with PoA at most $\max\{1, (1-\beta_\gamma(\mathcal{D}))^{-1}\}$ (Lemma 2 and Theorem 6). Moreover, we prove that that such a γ-enforceable flow and the corresponding γ-modifications can be computed in time polynomially related to the time needed for computing Nash flows in series-parallel networks (Lemma 3). An interesting open question is whether this result can be extended to heterogeneous risk-averse players.

In Section 6, we consider a generalization of γ-modifiable games where the p-norm of the vector $(\gamma_e)_{e\in E}$ of edge modifications is at most γ. We refer to this class as (p,γ)-modifiable games. This generalization captures applications where the total variance introduced in the network (instead of the variance per edge) should be bounded by γ. (p,γ)-modifications on m parallel links include $\gamma/\sqrt[p]{m}$-modifications as a special case and could potentially lead to an improved PoA. We prove that for routing games with latency functions in class $\mathcal{D}$, the worst-case PoA under (p,γ)-modifications is essentially identical to the worst-case PoA under $\gamma/\sqrt[p]{m}$-modifications (Theorem 8). Therefore, even for (p,γ)-modifiable games, the PoA of the $\gamma/\sqrt[p]{m}$-enforceable flow of Lemma 1 is essentially best possible.

Previous Work. On the conceptual side, our work is closest to those considering the PoA of stochastic routing games with risk-averse players, such as [12, 1, 15]. In this direction, Nikolova and Stier-Moses [13] recently introduced the *price of risk-aversion* (PRA), which is the worst-case ratio of the total latency of the Nash flow for risk-averse players to the total latency of the Nash flow for risk-neutral players. They proved that for the mean-variance (separable) cost and general networks with homogeneous players, the PRA is at most $1 + a\kappa\eta$, where a is the risk-aversion, κ is the maximum variance-to-mean ratio on some edge, and η is a parameter of the network ($\eta = 1$ for parallel-link and series-parallel networks). Interestingly, PRA can be smaller than 1 and as low as $1/\rho(\mathcal{D})$ for stochastic routing games on parallel-links (i.e., risk-aversion can improve the PoA to 1 for certain instances). This observation served as part of the motivation for this work.

On the technical side, our work is closest to those investigating the properties of restricted refundable tolls for selfish routing games [9, 2, 10]. In this direction, Bonifaci et al. [2] proved that for parallel-link networks with homogeneous players, computing the best γ-enforceable flow reduces to the solution of a convex program (and can be computed efficiently e.g., for linear delays). Moreover, they presented a tight bound of $\max\{1, (1 - \beta_\gamma(\mathcal{D}))^{-1}\}$ on the PoA of a γ-enforceable flow for routing games with latency functions in class $\mathcal{D}$. In this work, we introduce an efficiently computable and nicely structured class of γ-enforceable flows, generalize their analysis and extend the PoA bound to parallel-link games with heterogeneous players. Recently, Jelinek et al. [10] considered restricted tolls for heterogeneous players and proved that computing the best γ-enforceable flow for $s - t$ networks with affine latencies is $\mathbf{NP}$-hard. On the positive side, they proved that for parallel-link games with heterogeneous players, deciding whether a given flow is γ-enforceable (and finding the corresponding γ-modification) can be performed in polynomial time. Moreover, they showed how to compute the best γ-enforceable flow for parallel-link games with heterogeneous players if the maximum allowable modification on each edge is either 0 or infinite. In this work, we prove that computing the best γ-enforceable flow is $\mathbf{NP}$-hard for parallel links with affine latencies and show how to compute a γ-enforceable flow with best possible worst-case PoA for heterogeneous players.

2 The Model and Preliminaries

The basic model of routing games and most of the notation are introduced in Section 1. Next, we introduce some general notation and terminology and the classes of γ-modifiable and (p, γ)-modifiable games. In the following, for a positive integer k, we let $[k] = \{1, \ldots, k\}$.

γ-Modifiable Routing Games. A selfish routing game with heterogeneous players in k classes is a tuple $\mathcal{G} = (G(V, E), (\ell_e)_{e \in E}, (a^i)_{i \in [k]}, (r^i)_{i \in [k]})$, where G is a directed $s - t$ network with $m = |E|$ edges (or links), a^i is the aversion factor of the players in class i and r_i is the amount of traffic with aversion a^i. Wlog., we assume that $a^1 = 1$ and $a^1 < a^2 < \ldots < a^k$. In the special case that the players are homogeneous, we have a single class of players with risk aversion $a^1 = 1$ and traffic rate r. If the players are homogeneous, we usually denote an instance simply as $\mathcal{G} = (G, \ell, r)$.

Given a flow f, we let $f_p^{a^i}$ be the flow with aversion a^i on path p and $f_p = \sum_i f_p^{a^i}$ be the total flow on path p. Similarly, $f_e^{a^i} = \sum_{p: e \in p} f_p^{a^i}$ denotes the flow with aversion a^i on edge e and $f_e = \sum_i f_p^{a^i}$ is the total flow on edge e. We let $a_e(f)$ denote the aversion factor of some arbitrary player on edge e under f. If e is not used by f, we let $a_e(f) = a^k$. We let $a_e^{\min}(f)$ (resp. $a_e^{\max}(f)$) be the smallest (resp. largest) aversion factor in e under f. We say that an edge e (resp. path p) is used by players of type a^i if $f_e^{a^i} > 0$ (resp. for all $e \in p$). To simplify notation, we may use j in the subscript, instead of e_j, and i in the superscript instead of a^i. We also write ℓ_e, instead of $\ell_e(f)$ or $\ell_e(f_e)$, and a_j, instead of $a_j(f)$, when f is clear from the context.

We say that a routing game $\mathcal{G}$ is *γ-modifiable* if we can select a $\gamma_e \in [0, \gamma]$ for each edge e and change the edge latencies perceived by the players of type a^i from $\ell_e(x)$ to $(1 + a^i \gamma_e)\ell_e(x)$ using small random perturbations, as discussed in Section 1. Any vector $\boldsymbol{\Gamma} = (\gamma_e)_{e \in E}$, where $\gamma_e \in [0, \gamma]$ for each edge e, is a *γ-modification* of $\mathcal{G}$. Given a γ-modification $\boldsymbol{\Gamma}$, we let $\mathcal{G}^{\boldsymbol{\Gamma}}$ denote the γ-modified routing game where the perceived cost of the players is changed according to the modification $\boldsymbol{\Gamma}$.

A flow f is a *Nash flow* for the modified game $\mathcal{G}^{\boldsymbol{\Gamma}}$, if it routes all traffic on paths of minimum perceived cost, i.e., if for every path p and any aversion type a^i with $f_p^i > 0$ and every path p', $\sum_{e \in p}(1 + a^i \gamma_e)\ell_e(f) \le \sum_{e \in p'}(1 + a^i \gamma_{e'})\ell_{e'}(f)$. Given a routing game $\mathcal{G}$, we say that a flow f is *γ-enforceable*, or simply *enforceable*, if there exists a γ-modification $\boldsymbol{\Gamma}$ of $\mathcal{G}$ such that f is a Nash flow of $\mathcal{G}^{\boldsymbol{\Gamma}}$. We sometimes let $L(\mathcal{G})$ denote the common perceived cost of the players in the (unique) Nash flow of a routing game $\mathcal{G}$.

Our main assumption is that γ-modifications do not affect the expected latency. Therefore, the (expected) total latency of f in both $\mathcal{G}^{\boldsymbol{\Gamma}}$ and $\mathcal{G}$ is $C(f) = \sum_{e \in E} f_e \ell_e(f)$. Hence, the optimal flow o of $\mathcal{G}$ is also an optimal flow of $\mathcal{G}^{\boldsymbol{\Gamma}}$. A flow f is the *best γ-enforceable* flow of $\mathcal{G}$ if for any other γ-enforceable flow f' of $\mathcal{G}$, $C(f) \le C(f')$. The Price of Anarchy $\mathrm{PoA}(\mathcal{G}^{\boldsymbol{\Gamma}})$ of the modified game $\mathcal{G}^{\boldsymbol{\Gamma}}$ is equal to $C(f)/C(o)$, where f is the Nash flow of $\mathcal{G}^{\boldsymbol{\Gamma}}$. For a γ-modifiable game $\mathcal{G}$, the PoA of $\mathcal{G}$ under γ-modifications, denoted $\mathrm{PoA}_\gamma(\mathcal{G})$, is $C(f)/C(o)$, where f is the best γ-enforceable flow of $\mathcal{G}$. For routing games with latency functions in class $\mathcal{D}$, $\mathrm{PoA}_\gamma(\mathcal{D})$ denotes the maximum $\mathrm{PoA}_\gamma(\mathcal{G})$ over all γ-modifiable games $\mathcal{G}$ with latencies in $\mathcal{D}$.

(p, γ)-Modifiable Routing Games. For a generalization of γ-modifiable games, we consider an integer $p \geq 1$, select a modification $\gamma_e \geq 0$ for each edge e so that $\|(\gamma_e)_{e \in E}\|_p = \sqrt[p]{\sum_{e \in E} \gamma_e^p} \leq \gamma$ and change the perceived edge latencies as above. We refer to such games as (p, γ)-modifiable. Clearly, γ-modifiable games are (∞, γ)-modifiable. The notation and the notions introduced for γ-modifiable games can be naturally generalized to (p, γ)-modifiable games. In particular, the PoA of a game $\mathcal{G}$ under (p, γ)-modifications, denoted $\mathrm{PoA}_\gamma^p(\mathcal{G})$, is $C(f)/C(o)$, where f is the best (p, γ)-enforceable flow of $\mathcal{G}$. Similarly, $\mathrm{PoA}_\gamma^p(\mathcal{D})$ is the maximum PoA of all (p, γ)-modifiable games with latency functions in class $\mathcal{D}$.

Series-Parallel Networks. A directed $s - t$ network $G(V, E)$ is *series-parallel* if it either consists of a single edge (s, t) or can be obtained from two series-parallel graphs with terminals (s_1, t_1) and (s_2, t_2) composed either in series or in parallel. In a *series composition*, t_1 is identified with s_2, s_1 becomes s, and t_2 becomes t. In a *parallel composition*, s_1 is identified with s_2 and becomes s, and t_1 is identified with t_2 and becomes t. A series-parallel network can be completely specified by its *decomposition tree*, which is a rooted tree with a leaf for every edge. Each internal node of the decomposition tree represents either a series or a parallel component obtained from series (resp. parallel) compositions of the networks represented by its subtrees. The root of the tree represents the entire network. The decomposition tree of a series-parallel network $G(V, E)$ can be computed in $\mathrm{O}(|V| + |E|)$ time (see e.g. [19] for more details).

3 Modifying Routing Games in Parallel-Link Networks

We proceed to study γ-modifiable games on parallel-link networks with homogeneous risk-averse players. We first discuss a characterization of instances where the optimal flow is γ-enforceable. If the optimal flow is not enforceable, we show how to find a γ-enforceable flow that is close to optimal (in a sense made precise in Lemma 1). Furthermore, we provide an upper bound on the PoA achieved by such γ-enforceable flows and show that this bound is essentially best possible, in the sense that there are instances where no γ-enforceable flow can achieve a better PoA.

The following is a corollary of [2, Theorem 1] (and also of the main result in e.g., [6, 11]), applied to the special case of parallel links. For completeness, we include a simple proof in the Appendix, Section A.1.

Proposition 1. *Let $\mathcal{G}$ be a γ-modifiable game on parallel links and let o be the optimal flow of $\mathcal{G}$. Then, o is γ-enforceable in $\mathcal{G}$ if and only if for any link e with $o_e > 0$ and all links $e' \in E$, $\ell_e(o) \leq (1 + \gamma)\ell_{e'}(o)$.*

We next show that for any instance $\mathcal{G}$ with optimal flow o, there exist a flow f mimicking o and a γ-modification enforcing f as a Nash flow of the modified instance. Given the optimal flow o, the proof indicates an approach for computing f and the appropriate γ-modification. The running time is polynomial if we can compute the optimal flow o and a Nash flow of the original instance in polynomial time. Moreover, we later show that such a flow f achieves a best possible PoA in the worst case.

Lemma 1. *Let $\mathcal{G} = (G, \ell, r)$ be a γ-modifiable instance on parallel-links with homogeneous risk-averse players and let o be the optimal flow of $\mathcal{G}$. There is a feasible flow f and a γ-modification $\boldsymbol{\Gamma}$ of $\mathcal{G}$ such that*

(i) f is a Nash flow of the modified instance $G^{\boldsymbol{\Gamma}}$.

(ii) for any link e, if $f_e < o_e$, then $\gamma_e = 0$, and if $f_e > o_e$, then $\gamma_e = \gamma$.

Proof. Let o be the optimal flow of $\mathcal{G}$. If o is γ-enforceable, there is a γ-modification that turns o into a Nash flow of $\mathcal{G}^{\boldsymbol{\Gamma}}$. Clearly, the lemma holds in this case, if we set $f_e = o_e$, for all links e. If o is not γ-enforceable, we use induction on the number of links in $\mathcal{G}$.

In the base case of a single link, for any flow rate r, f and o coincide and the lemma holds under any modification. For the inductive step, let m be a used link with maximum latency in o. Removing link m and decreasing the total traffic rate by $o_m > 0$, we obtain an instance $\mathcal{G}_{-m} = (G_{-m}, \ell, r - o_m)$ with one link less than $\mathcal{G}$. By induction hypothesis, the lemma holds for $\mathcal{G}_{-m}$. So, there is a flow f' and a γ-modification $\boldsymbol{\Gamma}' = (\gamma'_e)_{e \in E_{-m}}$ so that (i) f' is the Nash flow of $\mathcal{G}^{\boldsymbol{\Gamma}'}_{-m}$ and (ii) for any link $e \in E_{-m}$, if $f'_e < o_e$, then $\gamma'_e = 0$, and if $f'_e > o_e$ then $\gamma'_e = \gamma$ (note that the restriction of o to E_{-m} is an optimal flow for $\mathcal{G}_{-m}$).

Now we restore link m and the traffic rate to r. If there is a modification γ_m so that $(1+\gamma_m)\ell_m(o) = L(f')$, we add γ_m to $\boldsymbol{\Gamma}'$ and obtain modification $\boldsymbol{\Gamma}$. Setting $f_m = o_m$ and $f_e = f'_e$, for the remaining links $e \neq m$, and using the induction hypothesis for f' and $\boldsymbol{\Gamma}'$, we obtain the desired flow f and the corresponding γ-modification $\boldsymbol{\Gamma}$. Otherwise, we have that $\ell_m(o) > L(f')$, since for any used link e under f', with $f'_e < o_e$, $\ell_e(f') \leq \ell_e(o) \leq \ell_m(o)$, and for any used link e' with $f'_{e'} \geq o_{e'}$, $\ell_e(f') = (1+\gamma_{e'})\ell_{e'}(f') \geq \ell_{e'}(f')$, by properties (i) and (ii) of the induction hypothesis.

To deal with the case where $\ell_{e_m}(o) > L(f')$, we carefully reroute flow from link m to the remaining links while maintaining properties (i) and (ii) in $\mathcal{G}_{-m}$. We do so until the latency of m becomes equal to the cost of the equilibrium flow that we maintain (under rerouting) in $\mathcal{G}_{-m}$. In order to maintain property (ii), we should pay attention to links e where the flow f'_e reaches o_e for the first time and to links e' where $\gamma'_{e'}$ reaches γ for the first time. For the former, we stop increasing flow and start increasing γ'_e, so that the equilibrium property is maintained. For the latter, we stop increasing $\gamma'_{e'}$ and start increasing the flow again.

More formally, we partition the links in E_{-m} in three classes, according to property (ii) and to the current equilibrium flow f' and modification $\boldsymbol{\Gamma}'$. Specifically, we let $E_1 = \{e \in E_{-m} : f'_e < o_e \text{ and } \gamma'_e = 0\}$, $E_2 = \{e \in E_{-m} : f'_e = o_e \text{ and } \gamma'_e < \gamma\}$ and $E_3 = \{e \in E_{-m} : f'_e \geq o_e \text{ and } \gamma'_e = \gamma\}$. By property (ii) and the induction hypothesis, E_1, E_2 and E_3 form a partitioning of E_{-m}. We let $L = (1+\gamma'_e)\ell_e(f')$, where e is any link with $f'_e > 0$, be the cost of the current equilibrium flow f' in $\mathcal{G}_{-m}$. Moreover, we let $L_1 = \min_{e \in E_1} \ell_e(o)$ be the minimum cost of an equilibrium flow in $\mathcal{G}_{-m}$ that causes some links of E_1 to move to E_2, let $L_2 = \min_{e \in E_2}(1+\gamma)\ell_e(o)$ be the minimum cost of an equilibrium flow in $\mathcal{G}_{-m}$ that causes some links of E_2 to move to E_3, and let $L' = \min\{L_1, L_2\} \geq L$, where the inequality follows from the definition of E_1 and E_2.

We reroute flow from link m to the links in $E_1 \cup E_3$ and increase γ'_e's for the links in E_2 so that we obtain an equilibrium flow in E_{-m} with cost L'. To this end, we let x_e be such that $L' = (1+\gamma'_e)\ell_e(f'_e + x_e)$, for all $e \in E_1 \cup E_3$. Namely, x_e is the amount of flow[6] we need to reroute to a link $e \in E_1 \cup E_3$ so that its cost becomes L'. For each link $e \in E_2$, we let $x_e = 0$ and increase its modification factor so that $L' = (1+\gamma'_e)\ell_e(o)$. So the total amount of flow that we need to reroute from E_{-m} is $x = \sum_{e \in E_{-m}} x_e$. Next, we distinguish between different cases depending on the flow and the latency in link m after rerouting.

If $x < o_m$ and $\ell_m(o_m - x) \geq L'$, we update the flow on link m to $o_m - x$, the flow on each link $e \in E_{-m}$ to $f'_e + x_e$, and the modification factors of all links in E_2 and apply the rerouting procedure again (in fact, if $\ell_m(o_m - x) = L'$, the procedure terminates with the desired γ-enforceable flow and the corresponding γ-modification). We note that by the definition of L', every time we apply the rerouting procedure, either some links e move from E_1 to E_2 (because after the update $f'_e = o_e$) or some links e' move from E_2 to E_3 (because after the update $\gamma'_e = \gamma$). Since links in E_3 cannot move to a different class, this rerouting procedure can be applied at most $2m$ times (in total, for all induction steps).

If $x < o_m$ and $\ell_m(o_m - x) < L'$, by continuity (see also [8, Section 3]), there is some $L'' \in (L, L')$ such that updating the flow and the modification factors with target equilibrium cost L'' (instead of L') reroutes flow $x' \leq x < o_m$ from link m to the links in E_{-m} so that $\ell_m(o_m - x') = L''$ and L'' is the cost of any used link in E_{-m}. Hence, we obtain the desired γ-enforceable flow f and the corresponding modification Γ. Such a value L'' can be found (either by binary search or) by computing the (unique) equilibrium flow[7] f for the links in $E_1 \cup E_3 \cup \{m\}$ with total traffic rate $o_m + \sum_{e \in E_1 \cup E_3} f'_e$ and modifications $\gamma_e = 0$ for

[6] Note that if $\ell_e(x)$ is not strictly increasing, x_e may not be uniquely defined (and it may be $L' = L$). Then, for each $e \in E_1$, we let x_e be the largest value such that $f'_e + x_e \leq o_e$ and $L' = \ell_e(f'_e + x_e)$ (i.e., if $L' = \ell_e(o_e)$, x_e becomes $o_e - f'_e$ so that e moves from E_1 to E_2). For each $e \in E_3$, we let x_e be the smallest value such that $L' = \ell_e(f'_e + x_e)$.

[7] Before rerouting, the equilibrium cost for the links in $E_1 \cup E_3$ (with traffic rate $\sum_{e \in E_1 \cup E_3} f'_e$) is $L < L'$ and $\ell_e(o_m) > L$. After we reroute x units of flow from link m to $E_1 \cup E_3$, the equilibrium cost for the links in $E_1 \cup E_3$ (with traffic rate

all links $e \in E_1 \cup \{m\}$ and $\gamma_e = \gamma$ for all links $e \in E_3$ (note also that by the definition of L', the link partitioning E_1, E_2 and E_3 does not change if we reroute flow with target equilibrium cost $L'' \in (L, L')$). Moreover, for all links $e \in E_2$, we let $f_e = o_e$ and set γ_e so that $L'' = (1 + \gamma_e)\ell_e(o_e)$, where $\gamma_e \leq \gamma$, because $L'' \leq L'$ (and by the definition of L').

If $x = o_m$ and $\ell_m(0) < L'$, the target equilibrium cost L'' lies between L and L' and we apply the same procedure as above. If $x = o_m$ and $\ell_m(0) \geq L'$, we let $\gamma_m = 0$ and $f_m = 0$. Then, we apply rerouting as above and set $f_e = f'_e$ and $\gamma_e = \gamma'_e$ for the remaining links $e \in E_{-m}$ (where f'_e and γ'_e are the values after the update of the flow and the modification factors). Thus, we obtain the desired enforceable flow f and the corresponding γ-modification.

If $x > o_m$ and $\ell_m(0) \geq L'$, the target equilibrium cost L'' lies between L and L' and link m is not used at equilibrium. So, we let $\gamma_m = 0$ and $f_m = 0$, compute the equilibrium flow f for the links in $E_1 \cup E_3$ with traffic rate $r - \sum_{e\in E_2} o_e$ and modifications $\gamma_e = 0$ for all $e \in E_1$ and $\gamma_e = \gamma$ for all $e \in E_3$. If $L'' \in (L, L')$ is the cost of this equilibrium flow, for all links $e \in E_2$, we let $f_e = o_e$ and set γ_e so that $L'' = (1+\gamma_e)\ell_e(o_e)$. If $x > o_m$ and $\ell_m(0) < L'$, the target equilibrium cost L'' again lies between L and L', but now link m may be used at equilibrium. Hence, we apply the same procedure but with link m now included in E_1. With this case, we have covered all the cases and have concluded the proof of the lemma. □

The proof of Lemma 1 implies that if the optimal flow and the Nash equilibrium flow can be computed efficiently, we can efficiently compute such a γ-enforceable flow f and the corresponding modification $\boldsymbol{\Gamma}$.

Corollary 1. *Let $\mathcal{G} = (G, \ell, r)$ be a γ-modifiable instance on parallel-links with homogeneous risk-averse players. Given the optimal flow of $\mathcal{G}$, we can compute a feasible flow f and a γ-modification $\boldsymbol{\Gamma}$ of $\mathcal{G}$ with the properties (i) and (ii) of Lemma 1 in time $O(mT_{NE})$, where T_{NE} is the complexity of computing the Nash flow of any given γ-modification of $\mathcal{G}$.*

Proof. In the proof of Lemma 1, we need to compute the equilibrium flow of a subinstance of $\mathcal{G}$, with a given γ-modification, at most once in each induction step. To see this, notice that the equilibrium computation step for the links in $E_1 \cup E_2 \cup \{m\}$ (or in $E_1 \cup E_2$) always concludes the proof of the induction step. Moreover, the computation of the values x_e for links in $E_1 \cup E_3$ can be reduced to a Nash flow computation on parallel links, a new link with constant latency function L' and the links in $E_1 \cup E_3$ with latency functions $(1 + \gamma'_e)\ell_e(f'_e + x)$ and with traffic rate sufficiently large. We need to compute such values x_e at most $3m$ times in total, once at the beginning of each induction step and at most $2m$ times after some link moves either from E_1 to E_2 or from E_2 to E_3. Therefore, we need $O(m)$ Nash flow computations in total. □

Price of Anarchy Analysis. The γ-enforceable flow f of Lemma 1 has a simple and nice structure and is fast and simple to compute. We highlight that it is different, in general, from the best γ-enforceable flow computed by [2, Algorithm 1]. We next show an upper bound on the PoA of f, which also serves as an upper bound on the PoA_γ of the best γ-enforceable flow. Moreover, we show that the PoA of f is best possible in the worst-case (see Theorem 2). The approach is conceptually similar to that of [3] and exploits the properties (i) and (ii) of Lemma 1. The improvement on the PoA due to γ-modifications is quantified by the term $-\gamma(x-y)\ell(x)$ which appears in $\sup_{\ell\in\mathcal{D},x\geq y\geq 0} \frac{y(\ell(x)-\ell(y))-\gamma(x-y)\ell(x)}{x\ell(x)}$. The results are similar to the results in [2, Section 4], although our approach and the γ-modification that we consider here are different.

Theorem 1. *For γ-modifiable instances on parallel-links with latency functions in class $\mathcal{D}$,*

$$PoA_\gamma(\mathcal{D}) \leq \rho_\gamma(\mathcal{D}) = \max\left\{1, \frac{1}{1-\beta_\gamma(\mathcal{D})}\right\}, \text{ where } \beta_\gamma(\mathcal{D}) = \sup_{\ell\in\mathcal{D},x\geq y\geq 0} \frac{y(\ell(x)-\ell(y))-\gamma(x-y)\ell(x)}{x\ell(x)}$$

$\sum_{e\in E_1\cup E_3}(f'_e + x_e))$ is $L' > L$ and $\ell_e(o_m - x) < L' < L$. Hence, by continuity and due to the parallel link structure of the network, the unique equilibrium flow has equilibrium cost $L'' \in (L, L')$.

Proof sketch. Let $\mathcal{G} = (G, \ell, r)$ be an instance on parallel-links with latency functions in class $\mathcal{D}$ and let o be the optimal solution of $\mathcal{G}$. We consider the γ-enforceable flow f and the corresponding modification $\boldsymbol{\Gamma} = (\gamma_e)_{e \in E}$ that exist for $\mathcal{G}$, due to Lemma 1. By definition, $\mathrm{PoA}_\gamma(\mathcal{G}) \leq PoA(\mathcal{G}^{\boldsymbol{\Gamma}})$. We next establish an upper bound on $\mathrm{PoA}(\mathcal{G}^{\boldsymbol{\Gamma}})$.

Similarly to the proof of Lemma 1, we partition the links used by f into sets E_1, E_2 and E_3 defined as $E_1 = \{e \in E : 0 < f_e < o_e\}$, $E_2 = \{e \in E : f_e = o_e > 0\}$ and $E_3 = \{e \in E : f_e > o_e\}$. Using the fact that f is a Nash flow of $\mathcal{G}^{\boldsymbol{\Gamma}}$, we can show that (see Section A.2, in the Appendix for the details):

$$\sum_{e \in E} f_e \ell_e(f) \leq \sum_{e \in E} o_e \ell_e(o) + \sum_{e \in E_3} \Big(o_e(\ell_e(f) - \ell_e(o)) - \gamma(f_e - o_e)\ell_e(f) \Big) \qquad (1)$$

Using the definition of $\beta_\gamma(\mathcal{D})$ as $\beta_\gamma(\mathcal{D}) = \sup_{\ell \in \mathcal{D}, x \geq y \geq 0} \frac{y(\ell(x) - \ell(y)) - \gamma(x-y)\ell(x)}{x\ell(x)}$, we obtain that:

$$\sum_{e \in E} f_e \ell_e(f) \leq \sum_{e \in E} o_e \ell_e(o) + \beta_\gamma(\mathcal{D}) \sum_{e \in E_3} f_e \ell_e(f)$$

Thus, $\mathrm{PoA}_\gamma(\mathcal{G}) \leq \mathrm{PoA}(\mathcal{G}^{\boldsymbol{\Gamma}}) = \frac{\sum_{e \in E} f_e \ell_e(f)}{\sum_{e \in E} o_e \ell_e(o)} \leq \rho_\gamma(\mathcal{D})$. Therefore, for the class of instances with latency functions in class $\mathcal{D}$, $\mathrm{PoA}_\gamma(\mathcal{D}) \leq \rho_\gamma(\mathcal{D})$. ◻

Next we give upper bounds on the $\mathrm{PoA}_\gamma(\mathcal{D})$ for γ-modifiable instances with polynomial latency functions. These bounds apply to the γ-enforceable flow f of Lemma 1 and to the best γ-enforceable flow (see also the similar PoA bounds in [2, Section 4]). The proofs are deferred to the Appendix, Section A.3.

Corollary 2. *For γ-modifiable instances on parallel links with polynomial latency functions of degree d, we have that $\mathrm{PoA}_\gamma(d) = 1$, for all $\gamma \geq d$, and*

$$\mathrm{PoA}_\gamma(d) \leq \frac{1}{1 - d\left(\frac{\gamma+1}{d+1}\right)^{\frac{d+1}{d}} + \gamma}, \text{ for all } \gamma \in [0, d)\,.$$

For affine latency functions, in particular, $\mathrm{PoA}_\gamma(1) = 1$, for all $\gamma \geq 1$, and

$$\mathrm{PoA}_\gamma(1) \leq \frac{1}{1 - (1-\gamma)^2/4}, \text{ for all } \gamma \in [0, 1)\,.$$

To quantify the improvement due to γ-modifications, we observe that for affine latency functions, the worst-case $\mathrm{PoA}_\gamma(1)$ with γ-modifications decreases fast as γ grows from 0 to 1.

In the Appendix, Section A.4, we show that our bounds on the PoA_γ are best possible in the worst-case.

Theorem 2. *For any class of latency functions $\mathcal{D}$ and for any $\epsilon > 0$, there is a γ-modifiable instance $\mathcal{G}$ on parallel links with homogeneous risk-averse players and latencies in class $\mathcal{D}$ so that $\mathrm{PoA}_\gamma(\mathcal{G}) \geq \rho_\gamma(\mathcal{D}) - \epsilon$.*

4 Modifying Parallel-Link Games with Heterogeneous Players

In contrast to the case of homogeneous players, where we can compute the best γ-enforceable flow in polynomial time (at least for affine latencies, see [2, Algorithm 1]), we next show that computing the best γ-enforceable flow for heterogeneous risk-averse players is $\mathbf{NP}$-hard. This holds even for affine latencies and only two classes of risk-averse players.

Theorem 3. *Given an instance $\mathcal{G}$ on parallel links with affine latencies and two classes of risk-averse players, a $\gamma > 0$, and a target cost $C > 0$, it is $\mathbf{NP}$-hard to determine whether there exists a γ-enforceable flow of $\mathcal{G}$ of total latency at most C.*

Proof. The proof is a modification of the construction in [16, Section 6], which shows that the best Stackelberg modification on parallel link networks with affine latencies and two classes of players (selfish and coordinated) is $\mathbf{NP}$-hard. Intuitively, the players with low aversion factor a^1 (resp. high aversion factor a^2) in our construction correspond to selfish (resp. coordinated) players in the construction of [16, Section 6].

Formally, we reduce $(1/3, 2/3)$-PARTITION to the best γ-enforceable flow. An instance of $(1/3, 2/3)$-PARTITION consists of n positive integers $s_1, s_2, \ldots, s_n$, so that $S = \sum_{i=1}^{n} s_i$ is a multiple of 3. We seek to determine whether there exists a subset X so that $\sum_{i \in X} s_i = 2S/3$.

For every instance $\mathcal{I}$ of $(1/3, 2/3)$-PARTITION, we create a routing game $\mathcal{G}$ consisting of $n+1$ parallel links, with affine latencies $\ell_i(x) = (x/s_i) + 4$, for all $i \in \{1, 2, \ldots, n\}$, and $\ell_{n+1}(x) = 3x/S$. The traffic rate is $r = 2S$, partitioned into two classes with traffic $r^1 = 3S/2$ and $r^2 = S/2$. We set $\gamma = 2/17$. For clarity, we first show $\mathbf{NP}$-hardness for the case where $a^1 = 0$ and $a^2 = 1$, i.e., the first class consists of risk-neutral players and the second class consists of risk-averse players. At the end of the proof, we discuss how to extend the proof to the case where $1 = a^1 < a^2$. We next show that $\mathcal{I}$ admits a $(1/3, 2/3)$-partition if and only if the routing game $\mathcal{G}$ admits a γ-enforceable flow f of total latency at most $35S/4$.

Let $\mathcal{I}$ be a YES-instance of $(1/3, 2/3)$-PARTITION and let X be a subset such that $\sum_{i \in X} s_i = 2S/3$. Given X, we construct a γ-enforceable flow f as follows: For risk-neutral players, we set $f_i^1 = 0$, for $i \in X$, $f_i^1 = s_i/4$, for $i \in \{1, \ldots, n\} \setminus X$ and $f_{n+1}^1 = 17S/12$. For risk-averse players, we set $f_i^2 = 3s_i/4$, for $i \in X$, and 0, otherwise. It is easy to see that f is the Nash flow of the γ-modification with $\gamma_i = 0$, for $i \in X$, $\gamma_i = \gamma = 2/17$, for $i \notin X$. Then, the delay (and the perceived cost of the risk-neutral players) is $17/4$ for any edge not in X (including edge $n+1$) and $19/4$ for any edge in X. The perceived cost of risk-averse players is $19/4$ on all edges. Therefore, $C(f) = 35S/4$.

Now, let $\mathcal{I}$ be a NO-instance of $(1/3, 2/3)$-PARTITION. We will show that any γ-enforceable flow f has total latency greater than $35S/4$. To this end, we fix an arbitrary γ-enforceable flow f (along with the corresponding γ-modification that turns f into a Nash flow) and partition the edges into two sets: X_1 consists of all edges i with positive risk-neutral traffic in f, i.e., with $f_i^1 > 0$, and X_2 consists of the remaining edges i with $f_i^1 = 0$. The total traffic on the edges of X_1 is $r(X_1) = \sum_{i \in X_1} f_i = (3/2 + \tau)S$, where $\tau \geq 0$ corresponds to the risk-averse traffic routed through X_1. We also let $r(X_2) = \sum_{i \in X_2} = (1/2 - \tau)S$ be the total traffic through X_2. We observe that edge $n+1$ must be in X_1. Otherwise, $\ell_{n+1}(f) \leq 3/2 < 4$ and risk-neutral players could profitably deviate to edge $n+1$. For convenience, we let $f_{n+1} = (3/2 - \mu)S$ be the total traffic on edge $n+1$ and let $(\mu + \tau)S$ be the total traffic on the remaining edges of X_1. We can assume that $\mu \geq 0$, since otherwise $\ell_{n+1}(f) > 9/2$ and $C(f) > (9/2)(3S/2) + 4(S/2) = 35S/4$. We also let $tS = \sum_{i \in X_1 \setminus \{n+1\}} s_i$ and $(1-t)S = \sum_{i \in X_2} s_i$.

Since all edges in X_1 include some risk-neutral traffic and since all risk-neutral players must have the same latency in f (because γ-modifications do not affect risk-neutral players), all edges in X_1 must have latency equal to $\ell_{n+1}(f) = 9/2 - 3\mu$. Therefore, the total cost of the (risk-neutral and risk-averse) players routed through X_1 is $(9/2 - 3\mu)(3/2 + \tau)S$. Moreover, for any edge $i \in X_1$, $i \neq n+1$, it must be $4 + f_i/s_i = 9/2 - 3\mu$, which implies that $f_i = (1/2 - 3\mu)s_i$. Hence, $\tau + \mu = (1/2 - 3\mu)t$, which implies that $1 - t = 1 - \frac{\tau+\mu}{1/2 - 3\mu}$ and that $0 \leq \frac{\tau+\mu}{1/2-3\mu} \leq 1$. Using the latter, we obtain that $\mu \leq 1/8$.

To minimize the total latency of risk-averse traffic through X_2, the remaining traffic of $(1/2 - \tau)S$ is routed on the edges of X_2 so that all of them have the same latency, which must be equal to $4 + \frac{1/2-\tau}{1-t}$. Therefore, the total latency of f is:

$$C(f) = \left(\tfrac{9}{2} - 3\mu\right)\left(\tfrac{3}{2} + \tau\right)S + \left(4 + \tfrac{1/2-\tau}{1-t}\right)\left(\tfrac{1}{2} - \tau\right)S \qquad (2)$$

where $t = \frac{\tau+\mu}{1/2-3\mu} \in [0, 1]$, $\tau \geq 0$, $\mu \in [0, 1/8]$. Moreover, since f is γ-enforceable, we have that

$$4 + \tfrac{1/2-\tau}{1-t} \leq \left(\tfrac{9}{2} - 3\mu\right)\left(1 + \tfrac{2}{17}\right) \Rightarrow \tfrac{1/2-\tau}{1-t} \leq \tfrac{35-114\mu}{34}\,,$$

so that the risk-averse players do not profitably deviate from the edges of X_2 to the edges of X_1.

Under the constraints above, $C(f)$ is minimized for $\tau = 0$ and $\mu = 1/12$, for which give $t = 1/3$ and $C(f) = 35S/4$. However, since $\mathcal{I}$ is NO-instance of $(1/3, 2/3)$-PARTITION, $t \neq 1/3$. Therefore, we have either that $\tau = 0$ and $\mu \neq 1/12$ or $\tau > 0$. In both cases, $C(f) > 35S/4$. This concludes the proof for the case of two player classes, one with risk-neutral players and the other with risk-averse players.

A technical subtlety is that the minimum total latency $C(f)$ can decrease if we have $a^1 > 0$ (i.e., if we have $3S/2$ players of low risk-aversion $a^1 > 0$ and $S/2$ players of high risk-aversion $a^2 > a^1$). This is because we can use γ-modifications for players with aversion a^1 so that we improve the total latency of their routing through X_1. Applying the same analysis as above but with $a^1 > 0$, we obtain that for any edge $i \in X_1$, $i \neq n+1$, $(1/2 - 3\mu)s_i - O(a^1 s_i) \leq f_i \leq (1/2 - 3\mu)s_i + O(a^1 s_i)$, where a factor $\gamma_i \leq 2/17$ is also hidden in the O-notation. Therefore, $(1/2 - 3\mu)t - O(a^1 S) \leq \tau + \mu \leq (1/2 - 3\mu)t + O(a^1 S)$, which implies that $\frac{\tau+\mu}{1/2-3\mu} - O(a^1 S) \leq t \leq \frac{\tau+\mu}{1/2-3\mu} + O(a^1 S)$. Using the latter, we obtain that $\mu \leq \frac{1-O(a^1 S)}{6(4/3-O(a^1 S))}$. The possible decrease in t and the possible increase in μ, both by $O(a^1 S)$, at $\tau = 0$, imply a decrease in $C(f)$ by $O(a^1 S)$, due to $a^1 > 0$, which allows for a decrease in the flow and the latency of edge $n+1$. On the other hand, if $t \neq 1/3$, $|t - 1/3| \geq 1/S$, and the increase in the cost of $C(f)$ due to $t \neq 1/3$ is $\Omega(1/S)$. So, if we select a^1 appropriately small, e.g., setting $a^1 = O(1/S^3)$ suffices, the decrease in $C(f)$ due to $a^1 > 0$ is strictly less than the increase in $C(f)$ due to $t \neq 1/3$. So, the $\mathbf{NP}$-hardness proof extends to the case where $0 < a^1 < a^2 = 1$. Then, multiplying s_i's, a^1 and a^2 by $1/a^1$, we can show that computing the best γ-enforceable flow remains $\mathbf{NP}$-hard if $1 = a^1 < a^2$. □

4.1 Finding a γ-Enforceable Flow with Good Price of Anarchy

Since the best enforceable flow is $\mathbf{NP}$-hard, we proceed to establish the existence of an enforceable flow that "mimics" the optimal flow o, in the sense described by the properties (i) and (ii) in Lemma 1 and achieves a PoA as low as in Theorem 1. In the following, we assume that the links are indexed in increasing order of $\ell_i(f)$ order, i.e. $i < j \Rightarrow \ell_i(f) \leq \ell_j(f)$, with ties broken in favor of links with $f_e > 0$. We also recall that for the risk-aversion factors of the players, we assume wlog. that $1 = a^1 < a^2 < \ldots < a^k$. We begin with a necessary and sufficient condition for a flow f to be γ-enforceable. [10, Algorithm 1] shows how to efficiently compute a γ-modification for any flow f that satisfies the following.

Theorem 4. *([10, Theorem 5]) Let $\mathcal{G}$ be a γ-modifiable instance on parallel links with heterogeneous players, let f be a feasible flow of $\mathcal{G}$, and let μ be the maximum index of a link used by f. Then, f is γ-enforceable if and only if (i) for any link $i \in [\mu]$, $\gamma \ell_i(f) \geq \sum_{l=i}^{\mu-1} \frac{\ell_{l+1}(f) - \ell_l(f)}{a_{l+1}^{\min}}$ and (ii) for all links i and j, if $\ell_i(f) < \ell_j(f)$, then $a_i(f) \leq a_j(f)$ (i.e., more risk-averse players are routed on links with higher latency).*

To obtain a γ-enforceable flow f for an instance with heterogeneous players, we combine Lemma 1 with Theorem 4 and [10, Algorithm 1]. Specifically, we first ignore player heterogeneity and compute, using Lemma 1 and Corollary 1, a γ-enforceable flow f and the corresponding modification $\boldsymbol{\Gamma}$ so that f is a Nash flow of the modified instance $\mathcal{G}^{\boldsymbol{\Gamma}}$ when all players have the minimum risk-averse factor $a^1 = 1$. Assuming that the links are indexed in increasing order of their latencies in f, since f is γ-enforceable with risk-averse factor $a^1 = 1$ for all players, Theorem 4 implies that for any link $i \in [\mu]$, $(1+\gamma)\ell_i(f) \geq \ell_\mu(f)$.

Next, we greedily allocate the heterogeneous risk-averse players to f, taking their risk-averse factors into account, so that each link i receives flow f_i and property (ii) in Theorem 4 is satisfied (this is always possible). Finally, we use [10, Algorithm 1] and compute a γ-modification that turns f into an equilibrium flow for the modified instance with heterogeneous players. This is possible because, by construction, f satisfies condition (i) of Theorem 4. Specifically, for link $i \in [\mu]$,

$$\gamma \ell_i(f) \geq \ell_\mu(f) - \ell_i(f) = \sum_{l=i}^{\mu-1} (\ell_{l+1}(f) - \ell_l(f)) \geq \sum_{l=i}^{\mu-1} \frac{\ell_{l+1}(f) - \ell_l(f)}{a_{l+1}^{\min}}$$

The first inequality holds because f is γ-enforceable when all players are homogeneous and have risk-averse factor $a^1 = 1$ and the last inequality holds because $a_{l+1}^{\min} \geq 1$, for any link l.

Moreover, since f satisfies the properties of (i) and (ii) in Lemma 1, the PoA of f can be bounded as in Theorem 1 and (in Corollary 2, for polynomial and affine latencies). Hence, we obtain the following.

Theorem 5. *Let $\mathcal{G} = (G, \ell, (a^i)_{i\in[k]}, (r^i)_{i\in[k]})$ be a γ-modifiable instance on parallel-links with heterogeneous risk-averse players. Given the optimal flow of $\mathcal{G}$, we can compute a feasible flow f and a γ-modification $\boldsymbol{\Gamma}$ of $\mathcal{G}$ in time $O(mT_{NE})$, where T_{NE} is the complexity of computing the Nash flow of any given γ-modification of $\mathcal{G}$ with homogeneous risk-averse players. Moreover, the PoA_γ, under γ-modifications, achieved by f is upper bounded as in Theorem 1 and Corollary 2.*

5 Modifying Routing Games in Series-Parallel Networks

In this section, we consider γ-modifiable instances on series-parallel networks with homogeneous players and generalize the results of Section 3. We start with a sufficient and necessary condition for the optimal flow o to be γ-enforceable. The following generalizes Proposition 1 and is a corollary of [2, Theorem 1].

Proposition 2. *Let $\mathcal{G}$ be a γ-modifiable instance on a series-parallel network and let o be the optimal flow of $\mathcal{G}$. Then, o is γ-enforceable if and only if for any pair of internally vertex-disjoint paths p and q with common endpoints (possibly different from s and t) and with $o_e > 0$ for all edges $e \in p$, $\ell_p(o) \leq (1+\gamma)\ell_q(o)$.*

We proceed to generalize Lemma 1 to series-parallel networks. The proof of the following is based on an extension of the rerouting procedure used in the proof of Lemma 1 combined with a continuity property of γ-enforceable flows in series-parallel networks.

Lemma 2. *Let $\mathcal{G} = (G, \ell, r)$ be a γ-modifiable instance with homogeneous risk-averse players on a series-parallel network G and let o be the optimal flow of $\mathcal{G}$. There is a feasible flow f and a γ-modification $\boldsymbol{\Gamma}$ of $\mathcal{G}$ such that*

(i) f is a Nash flow of the modified instance $G^{\boldsymbol{\Gamma}}$.

(ii) for any edge e, if $f_e < o_e$, then $\gamma_e = 0$, and if $f_e > o_e$, then $\gamma_e = \gamma$.

Proof sketch. The proof is by induction on the structure of the series parallel network G and generalizes the proof of Lemma 1. For the base case of a single edge e, the lemma holds without any modifications.

For the inductive step, assume that G is the result of a series composition of series-parallel networks G_1 and G_2. Using the induction hypothesis, for $i \in \{1,2\}$, we let f_i be a γ-enforceable flow and $\boldsymbol{\Gamma}_i$ be a γ-modification of the corresponding instance $\mathcal{G}_i$ such that f_i is the Nash flow of $\mathcal{G}_i^{\boldsymbol{\Gamma}_i}$. Any $s-t$ path p of G is a concatenation of source-sink paths p_1 of G_1 and p_2 of G_2. Thus, by letting $\boldsymbol{\Gamma}$ be the combination of $\boldsymbol{\Gamma}_1$ and $\boldsymbol{\Gamma}_2$ and f be the combination of f_1 and f_2, we obtain a γ-enforceable flow f and the corresponding modification $\boldsymbol{\Gamma}$ that satisfy the lemma, due to the induction hypothesis.

The interesting case is where G is the result of a parallel composition of series-parallel networks G_1 and G_2. By induction hypothesis, for $i \in \{1,2\}$, we let f_i be a γ-enforceable flow of rate r_i, with $r_1 + r_2 = r$, and $\boldsymbol{\Gamma}_i$ be a γ-modification of $\mathcal{G}_i$ such that f_i is the Nash flow of $\mathcal{G}_i^{\boldsymbol{\Gamma}_i}$. In the following, we let $L_i = L(f_i)$ be the equilibrium cost of flow f_i through network G_i with latency functions modified according to $\boldsymbol{\Gamma}_i$.

If $L_1 = L_2$, i.e., $\mathcal{G}_1^{\boldsymbol{\Gamma}_1}$ with traffic rate r_1 and $\mathcal{G}_2^{\boldsymbol{\Gamma}_2}$ with traffic rate r_2 have the same equilibrium cost, then combining f_1 and f_2 to a feasible flow f of $\mathcal{G}$ and combining $\boldsymbol{\Gamma}_1$ and $\boldsymbol{\Gamma}_2$ to γ-modifications Γ, we obtain a γ-enforceable flow f and a modification $\boldsymbol{\Gamma}$ that satisfy the lemma, due to the induction hypothesis.

Otherwise, we assume wlog. that $L_1 > L_2$. To deal with this case, we generalize the rerouting procedure of Lemma 1. Starting with f_1 and f_2, we reroute flow from some used paths of $G_1^{\boldsymbol{\Gamma}_1}$ to $G_2^{\boldsymbol{\Gamma}_2}$, maintaining the equilibrium property on both $G_1^{\boldsymbol{\Gamma}_1}$ and $G_2^{\boldsymbol{\Gamma}_2}$ and trying to equalize their equilibrium cost. As in Lemma 1, we have also to maintain property (ii), by paying attention to edges e where the flow f_e reaches o_e for the first time and to edges e' where $\gamma_{e'}$ reaches γ for the first time. For the former, we stop increasing the flow through any paths including e and start increasing γ_e, so that the equilibrium property is maintained. For the latter, we stop increasing $\gamma_{e'}$ and start increasing again the flow through paths that include e'.

The idea of the proof is similar to the induction step in Lemma 1. However, since G_1 and G_2 are (possibly large) series-parallel networks connected in parallel, and not just two parallel links, we need a continuity property about the changes in the equilibrium flow of a network when the traffic rate slightly increases or decreases. This property essentially follows from [8, Section 3], but we give its formal proof in the Appendix, for completeness. Missing technical details can be found in the Appendix, Section A.5. □

Next, we observe that the proof of Theorem 1 does not use the parallel-link structure of the network, it just uses the properties (i) and (ii) of Lemma 1 and Lemma 2. Hence, we obtain the same upper bound on the PoA for the γ-enforceable flow f of Lemma 2 and also on the PoA_γ of the best γ-enforceable flow for series-parallel networks with homogeneous players.

Theorem 6. *For γ-modifiable instances on series-parallel networks with homogeneous players and latency functions in class $\mathcal{D}$,*

$$PoA_\gamma(\mathcal{D}) \le \rho_\gamma(\mathcal{D}) = \max\left\{1, \frac{1}{1-\beta_\gamma(\mathcal{D})}\right\}, \ \textit{where } \beta_\gamma(\mathcal{D}) = \sup_{\ell\in\mathcal{D}, x\ge y\ge 0} \frac{y(\ell(x)-\ell(y))-\gamma(x-y)\ell(x)}{x\ell(x)}$$

Moreover, given the optimal flow of an instance $\mathcal{G}$ on a series-parallel network, we show how to compute a γ-enforceable flow f and the corresponding modification so that we achieve the upper bound on the PoA established by Theorem 6. Given o, the running time is determined by the time required to compute a Nash flow of the original instance.

We first use Proposition 2 and determine whether the optimal flow o is γ-enforceable. To this end, we remove from G all edges unused by o and check the feasibility of the following linear system.

$$\begin{array}{rl} 0 \le \gamma_e \le \gamma & \forall \text{ used edges } e \\ \sum_{e\in p}(1+\gamma_e)\ell_e(o) = \max_{p:o_p>0} \ell_p(o) & \forall \text{ used paths } p \end{array} \qquad (O_\gamma)$$

If the linear system (O_γ) is not feasible, then o is not γ-enforceable, by Proposition 2. Otherwise, using the solution of (O_γ) as γ_e's for the edges of G used by o and setting $\gamma_e = 0$ for the unused edges[8] e, we enforce o as a Nash flow of the modified game $\mathcal{G}^{\boldsymbol{\Gamma}}$.

If (O_γ) is not feasible and o is not γ-enforceable, we exploit the constructive nature of the proof of Lemma 2 and find a γ-enforceable flow in time dominated by the time required to compute a Nash flow in series-parallel networks. The details can be found in the Appendix, Section A.6.

Lemma 3. *Let $\mathcal{G}$ be a γ-modifiable instance on a series-parallel network with homogeneous players. Given the optimal flow of $\mathcal{G}$ and for any $\epsilon > 0$, we can compute a feasible flow f and a γ-modification $\boldsymbol{\Gamma}$ of $\mathcal{G}$ with the properties (i) and (ii) of Lemma 2 in time $O(m^2 T_{NE}\log(r/\epsilon))$, where T_{NE} is the complexity of computing the Nash flow of any given γ-modification of $\mathcal{G}$ and ϵ is an accuracy parameter.*

6 Parallel-Link Games with Relaxed Modification Restrictions

In this section, we consider (p,γ)-modifiable games on parallel links with heterogeneous risk-averse players. Observing that any $\gamma/\sqrt[p]{m}$-modification is a (p,γ)-modification for a (p,γ)-modifiable game, we next show an upper bound on the PoA under such modifications. The details can be found in the Appendix, Section A.7.

[8] Any (unused) path p with an unused edge has $\ell_p(o) \ge \max_{p:o_p>0}\ell_p(o)$. Moreover, the perceived cost of p can only increase due to edge modifications. Since the modifications corresponding to the solution of (O_γ) make the perceived cost of all used paths equal to $\max_{p:o_p>0}\ell_p(o)$, o becomes a Nash flow of $\mathcal{G}^{\boldsymbol{\Gamma}}$.

Theorem 7. *For any (p, γ)-modifiable instance $\mathcal{G}$ on m parallel links with heterogeneous risk-averse players and latency functions in class $\mathcal{D}$, $PoA^p_\gamma(\mathcal{G}) \leq PoA_{\gamma_0}(\mathcal{G}) \leq \rho_{\gamma_0}(\mathcal{D})$, where $\gamma_0 = \gamma / \sqrt[p]{m}$.*

The above bound is tight under weak assumptions on the class $\mathcal{D}$ of latency functions. More specifically, we say that a class of latency functions $\mathcal{D}$ is of the form $\mathcal{D}_0$ if (a) ℓ is continuous and twice differentiable in $(0, +\infty)$, (b) $\ell'(x) > 0$, $\forall x \in (0, +\infty)$ or ℓ is constant, (c) ℓ is semi-convex, i.e. $x\ell(x)$ is convex in $[0, +\infty)$ and (d) if $\ell \in \mathcal{D}$, then $(\ell + c) \in \mathcal{D}$, for all constants $c \in \mathbb{R}$ such that for all $x \in \mathbb{R}_{\geq 0}$, $\ell(x) + c \geq 0$ [9]. Then we obtain the following.

Theorem 8. *For any class $\mathcal{D}$ of the form $\mathcal{D}_0$ and any $\epsilon > 0$, there is an instance $\mathcal{G}$ on m parallel links with homogeneous players and latency functions in class $\mathcal{D}$, so that $\mathrm{PoA}^p_\gamma(\mathcal{G}) \geq \rho_{\gamma_0}(\mathcal{D}) - \epsilon$, where $\gamma_0 = \gamma / \sqrt[p]{m}$.*

Proof. For convenience and clarity, we divide the proof into three steps. We consider an instance $\mathcal{I}_m$, with m parallel links, where the first $m-1$ links have the same latency function $\ell \in \mathcal{D}$ (to be fixed later) and link m has the constant latency function $(1+\gamma_1)\ell(\frac{r}{m-1})$, where $\gamma_1 = \gamma / \sqrt[p]{m-1}$. The instance has homogeneous risk-averse players with risk-aversion $a^1 = 1$. Also we let $\gamma_0 = \gamma / \sqrt[p]{m}$.
Claim 1. For every $m \geq 2$ and any $\ell \in \mathcal{D}$ with $\ell(0) = 0$, $\mathrm{PoA}^p_\gamma(\mathcal{I}_m) = \mathrm{PoA}_{\gamma_1}(\mathcal{I}_m)$.

In words, Claim 1 states that the best (p, γ)-modification for the instance $\mathcal{I}_m$ is the one that splits γ equally to the first $m-1$ edges. The proof applies KKT optimality conditions and is deferred to Section A.8.
Claim 2. For every $m \geq 2$ and any $\epsilon > 0$, there is a latency function $\ell_{\epsilon,m}$ with $\ell_{\epsilon,m}(0) = 0$ such that setting $\ell = \ell_{\epsilon,m}$ in the instance $\mathcal{I}_m$ results in $\mathrm{PoA}_{\gamma_1}(\mathcal{I}_m) \geq \rho_{\gamma_1}(\mathcal{D}) - \epsilon/2$.

The proof of Claim 2 is similar to the proof of Theorem 2, with the instance now consisting m parallel links, instead of 2. The proof can be found in Section A.9.

Since $\ell_{\epsilon,m}(0) = 0$, we can combine claims 1 and 2 and obtain that for any $m \geq 2$ and any $\epsilon > 0$, $\mathrm{PoA}^p_\gamma(\mathcal{I}_m) \geq \rho_{\gamma_1}(\mathcal{D}) - \epsilon/2$, is we use the latency function $\ell_{\epsilon,m}$.
Claim 3. For every class of latency functions $\mathcal{D}$, any $\epsilon > 0$ and any γ, there exists an $m_\epsilon \geq 2$ such that $\rho_{\gamma_1}(\mathcal{D}) \geq \rho_{\gamma_0}(\mathcal{D}) - \epsilon/2$.

The proof is based on the fact that γ_1 tends to γ_0 as the number of parallel links m grows. The details can be found in Section A.10. Therefore, for all $\epsilon > 0$, there are an m_ϵ and a latency function $\ell_{\epsilon,m_\epsilon}$ such that $\mathrm{PoA}^p_\gamma(\mathcal{I}_{m_\epsilon}) \geq \rho_{\gamma_0}(\mathcal{D}) - \epsilon$. □

References

1. H. Angelidakis, D. Fotakis, and T. Lianeas. Stochastic congestion games with risk-averse players. In *Proc. of the 6th Symposium on Algorithmic Game Theory (SAGT '13)*, LNCS 8146, pp. 86-97, 2013.
2. V. Bonifaci, M. Salek, and G. Schäfer. Efficiency of restricted tolls in non-atomic network routing games. In *Proc. of the 4th Symposium on Algorithmic Game Theory (SAGT '10)*, LNCS 6982, pp. 302-313, 2011.
3. J.R. Correa, A.S. Schulz, and N.E. Stier Moses. Selfish Routing in Capacitated Networks. *Mathematics of Operations Research*, 29(4):961-976, 2004.
4. A. Fiat and C.H. Papadimitriou. When the players are not expectation maximizers. In *Proc. of the 3th Symposium on Algorithmic Game Theory (SAGT '10)*, LNCS 6386, pp. 1-14, 2010.
5. L. Fleischer. Linear Tolls Suffice: New Bounds and Algorithms for Tolls in Single Source Networks. *Theoretical Computer Science*, 348:217-225, 2005.
6. L. Fleischer, K. Jain, and M. Mahdian. Tolls for Heterogeneous Selfish Users in Multicommodity Networks and Generalized Congestion Games. In *Proc. of the 45th IEEE Symp. on Foundations of Computer Science (FOCS '04)*, pp. 277-285, 2004.
7. D. Fotakis and P. Spirakis. Cost-Balancing Tolls for Atomic Network Congestion Games. In *Proc. of the 3rd Workshop on Internet and Network Economics (WINE '07)*, pp. 179-190, 2007.
8. M.A. Hall. Properties of the Equilibrium State in Transportation Networks. *Transportation Science*, 12(3):208-216, 1978.
9. M. Hoefer, L. Olbrich, and A. Skopalik. Taxing subnetworks. In *Proc. of the 4th Workshop on Internet and Network Economics (WINE '08)*, LNCS 5385, pp. 286-294, 2008.

[9] Property (d) requires that $\mathcal{D}$ should be closed under addition of constants, as long as the resulting function remains nonnegative.

10. T. Jelinek, M. Klaas, and G. Schäfer. Computing optimal tolls with arc restrictions and heterogeneous players. In *Proc. of the 31st Symposium on Theoretical Aspects of Computer Science (STACS '14)*, LIPIcs 25, pp. 433-444, 2014.
11. G. Karakostas and S. Kolliopoulos. Edge Pricing of Multicommodity Networks for Heterogeneous Users. In *Proc. of the 45th IEEE Symp. on Foundations of Computer Science (FOCS '04)*, pp. 268-276, 2004.
12. E. Nikolova and N. Stier Moses. Stochastic selfish routing. In *Proc. of the 4th Symposium on Algorithmic Game Theory (SAGT '11)*, LNCS 6982, pp. 314-325, 2011.
13. E. Nikolova and N. Stier-Moses. The burden of risk aversion in mean-risk selfish routing. In *Proc. of the 16th ACM Conference on Electronic Commerce (EC '15)*, pp. 489-506, 2015.
14. F. Ordóñez and N. Stier Moses. Wardrop equilibria with risk-averse users. *Transportation Science*, 44(1):63-86, 2010.
15. G. Piliouras, E. Nikolova, and J.S. Shamma. Risk Sensitivity of Price of Anarchy under Uncertainty. In *Proc. of the 14th ACM Conference on Electronic Commerce (EC '13)*, pp. 715-732, 2013.
16. T. Roughgarden. Stackelberg Scheduling Strategies. *SIAM Journal on Computing*, 33(2):332-350, 2004.
17. Tim Roughgarden. *Selfish routing and the Price of Anarchy*. MIT press, 2005.
18. A. Tversky and D. Kahneman. Prospect theory: An analysis of decision under risk. *Econometrica*, 47(2):263-291, 1979.
19. J. Valdez, R.E. Tarjan, and E.L. Lawler. The Recognition of Series-Parallel Digraphs. *SIAM Journal on Computing*, 11(2):298-313, 1982.

A Appendix: Omitted Proofs and Technical Details

A.1 The Proof of Proposition 1

Let o be γ-enforceable and let $\boldsymbol{\Gamma} = (\gamma_e)_{e\in E}$ be a γ-modification that turns o into a Nash flow of $\mathcal{G}^{\boldsymbol{\Gamma}}$. Wlog., we consider any pair of links e and e' with $o_e > 0$ and $\ell_{e'}(o) \leq \ell_e(o)$. Since o is a Nash flow of $\mathcal{G}^{\boldsymbol{\Gamma}}$, $(1+\gamma_e)\ell_e(o) \leq (1+\gamma_{e'})\ell_{e'}(o)$, which implies that

$$\ell_e(o) \leq (1+\gamma_e)\ell_e(o) \leq (1+\gamma_{e'})\ell_{e'}(o) \leq (1+\gamma)\ell_{e'}(o)$$

For the converse, let $\ell_{\max}(o) = \max_{e:o_e>0} \ell_e(o)$ be the maximum latency among all used edges in o. By hypothesis, for any link e, $\ell_{\max}(o) \leq (1+\gamma)\ell_e(o)$. For each edge e used by o, we let $\gamma_e = \frac{\ell_{\max}(o)-\ell_e(o)}{\ell_e(o)}$. We observe that $\gamma_e \leq \gamma$, because $\ell_{\max}(o) \leq (1+\gamma)\ell_e(o)$. For each edge e with $o_e = 0$, we let $\gamma_e = 0$. For any link e with $o_e > 0$, we have that $(1+\gamma_e)\ell_e(o) = \ell_{\max}(o)$. Moreover, for any link e' with $o_{e'} = 0$, the optimality conditions of o (see e.g., [17, Corollary 2.4.6]) imply that $\ell_{e'}(o) \geq \ell_{\max}(o)$. Specifically, since $o_{e'} = 0$, it must be $(o_{e'}\ell_{e'}(o))' \geq (o_e\ell_e(o))'$ for any link e with $o_e > 0$ and $\ell_e(o) = \ell_{\max}(o)$ (here $(x\ell_e(x))'$ denotes the derivative of $x\ell_e(x)$). Using that $(o_{e'}\ell_{e'}(o))' = \ell_{e'}(o)$, because $o_{e'} = 0$, and that $(o_e\ell_e(o))' = \ell_e(o) + o_e(\ell_e(o))' \geq \ell_{\max}(o)$, because we consider non-decreasing latency functions, we conclude that $\ell_{e'}(o) \geq \ell_{\max}(o)$. Therefore, o is enforceable by the particular γ-modification. □

A.2 Missing Technical Details of the Proof of Theorem 1

We next prove (1), which is the main inequality used in the proof of Theorem 1. Using the fact that f is a Nash flow of $\mathcal{G}^{\boldsymbol{\Gamma}}$, we have that

$$\sum_{e\in E} f_e(1+\gamma_e)\ell_e(f) \leq \sum_{e\in E} o_e(1+\gamma_e)\ell_e(f) \tag{3}$$

Since all links not in $E_1 \cup E_2 \cup E_3$ have $f_e = 0$ (i.e., they are not used by f) and since, by Lemma 1, all links $e \in E_1$ have $\gamma_e = 0$ and all links $e \in E_3$ have $\gamma_e = \gamma$, the lhs of (3) becomes:

$$\sum_{e\in E} f_e(1+\gamma_e)\ell_e(f) = \sum_{e\in E_1} f_e\ell_e(f) + \sum_{e\in E_2} f_e(1+\gamma_e)\ell_e(f) + \sum_{e\in E_3} f_e(1+\gamma)\ell_e(f) \tag{4}$$

Similarly, and since for any link e with $0 = f_e < o_e$, $\gamma_e = 0$, by Lemma 1, the rhs of (3) becomes:

$$\sum_{e\in E} o_e(1+\gamma_e)\ell_e(f) = \sum_{e\notin E_2\cup E_3} o_e\ell_e(f) + \sum_{e\in E_2} o_e(1+\gamma_e)\ell_e(f) + \sum_{e\in E_3} o_e(1+\gamma)\ell_e(f) \tag{5}$$

Combining (3) with (4) and (5), we obtain that:

$$\begin{aligned}\sum_{e\in E} f_e\ell_e(f) &\le \sum_{e\in E} o_e\ell_e(f) - \sum_{e\in E_2}\gamma_e(f_e-o_e)\ell_e(f) - \sum_{e\in E_3}\gamma(f_e-o_e)\ell_e(f)\\ &\le \sum_{e\in E} o_e(\ell_e(f)-\ell_e(o)) + \sum_{e\in E} o_e\ell_e(o) - \sum_{e\in E_3}\gamma(f_e-o_e)\ell_e(f)\\ &\le \sum_{e\in E} o_e\ell_e(o) + \sum_{e\in E_3}\Big(o_e(\ell_e(f)-\ell_e(o)) - \gamma(f_e-o_e)\ell_e(f)\Big)\end{aligned}$$

For the second inequality, we use that for all links $e \in E_2$, $f_e = o_e$. For the third inequality, we use that for all links $e \notin E_3$, $f_e \le o_e$ and thus, $\ell_e(f) - \ell_e(o) \le 0$. This concludes the proof of (1). □

A.3 The Proof of Corollary 2

Let $\mathcal{G} = (G, \ell, r)$ be a γ-modifiable instance on parallel links with homogeneous risk-averse players and polynomial latency functions of degree d and let o be the optimal flow of $\mathcal{G}$. It is well known (see e.g., [17]) that o is the equilibrium flow of a routing game on the same underlying network G and with the same traffic rate r, but with latency functions $(x\ell_e(x))'$, instead of $\ell_e(x)$. Therefore, for polynomials of degree d, o is the equilibrium flow of instance $\mathcal{G}' = (G, \ell + x\ell', r)$, where the latency function of a link e is $\sum_{i=0}^d (i+1)a_e(i)x^i$, instead of $\sum_{i=0}^d a_e(i)x^i$, where all $a_e(i)$ are nonnegative.

If $\gamma \ge d$, $\sum_{i=0}^d (i+1)a_e(i)x^i \le (1+\gamma)\sum_{i=0}^d a_e(i)x^i$. Thus, there exist γ_e's so that for any link e, $\sum_{i=0}^d (i+1)a_e(i)(o_e)^i = (1+\gamma_e)\sum_{i=0}^d a_e(i)(o_e)^i$, which implies that the optimal flow of $\mathcal{G}$ is γ-enforceable.

If $\gamma < d$, we bound $\beta_\gamma(d)$. We observe that $\frac{y(\ell(x)-\ell(y))-\gamma(x-y)\ell(x)}{x\ell(x)} = \frac{y}{x}\Big(1+\gamma-\frac{\ell(y)}{\ell(x)}\Big) - \gamma$. If $\ell(x)$ is a polynomial of degree d, $\ell(x) = \sum_{i=0}^d a_i x^i$. Hence, for all $y \le x$, $\frac{\ell(y)}{\ell(x)} \ge \frac{y^d}{x^d}$, because $y^{d-i} \le x^{d-i}$ and thus, $\sum a_i y^i x^d \ge \sum a_i x^i y^d$. Therefore, $\frac{y}{x}\Big(1+\gamma-\frac{\ell(y)}{\ell(x)}\Big) - \gamma \le \frac{y}{x}\Big(1+\gamma-\frac{y^d}{x^d}\Big) - \gamma$. The rhs is maximized for y satisfying $(y^{d+1})' = (1+\gamma)x^d$, which gives $y = \sqrt[d]{\frac{\gamma+1}{d+1}}x$. Thus, we obtain that $\beta_\gamma(d) = d\Big(\frac{\gamma+1}{d+1}\Big)^{\frac{d+1}{d}} - \gamma$ and that $\rho_\gamma(d) = \Big(1 - d(\frac{\gamma+1}{d+1})^{\frac{d+1}{d}} + \gamma\Big)^{-1}$. □

A.4 The Proof of Theorem 2

Let $\epsilon > 0$ and consider a γ-modifiable instance $\mathcal{G}$ on two parallel links, e_1 and e_2, with traffic rate r_ϵ, which will be determined later, and homogeneous risk-averse players. As for latency functions, the latency function of link e_1 is $\ell_1(x) = \ell_\epsilon(x)$, where $\ell_\epsilon(x)$ is a latency function in $\mathcal{D}$ that will be determined later, and the latency function of link e_2 is the constant function $\ell_2(x) = (1+\gamma)\ell_\epsilon(r_\epsilon)$.

We consider any γ-modification $\boldsymbol{\Gamma} = (\gamma_1, \gamma_2)$ and the corresponding equilibrium flow f of $\mathcal{G}^{\boldsymbol{\Gamma}}$. Then, all traffic goes through link e_1 and thus $C(f) = r_\epsilon\ell_\epsilon(r_\epsilon)$. At the optimal flow o, let o_1 be the traffic routed on link e_1. Then, $C(o) = (r_\epsilon - o_1)(\gamma+1)\ell_\epsilon(r_\epsilon) + o_1\ell_\epsilon(o_1)$. Therefore,

$$\mathrm{PoA}(\mathcal{G}^{\boldsymbol{\Gamma}}) = \frac{r_\epsilon\ell_\epsilon(r_\epsilon)}{(r_\epsilon-o_1)(\gamma+1)\ell_\epsilon(r)+o_1\ell_\epsilon(o_1)} = \frac{1}{1-\frac{o_1(\ell_\epsilon(r_\epsilon)-\ell_\epsilon(o_1))-\gamma(r_\epsilon-o_1)\ell_\epsilon(r_\epsilon)}{r_\epsilon\ell_\epsilon(r_\epsilon)}}$$

Since o is the optimal flow of $\mathcal{G}$, $o_1 \in [0, r_\epsilon]$ is chosen so that $C(o)$ is minimized, or equivalently, that $\frac{r_\epsilon\ell_\epsilon(r_\epsilon)}{(r_\epsilon-o_1)(\gamma+1)\ell(r_\epsilon)+o_1\ell_\epsilon(o_1)}$ is maximized. Therefore, o_1 maximizes

$$\frac{o_1(\ell_\epsilon(r_\epsilon)-\ell_\epsilon(o_1))-\gamma(r_\epsilon-o_1)\ell_\epsilon(r_\epsilon)}{r_\epsilon\ell_\epsilon(r_\epsilon)} \tag{6}$$

We recall that $\beta_\gamma(\mathcal{D}) = \sup_{\ell\in\mathcal{D},x\geq y\geq 0} \frac{y(\ell(x)-\ell(y))-\gamma(x-y)\ell(x)}{x\ell(x)}$. Using the definition of sup, we let ℓ_ϵ be any latency function in $\mathcal{D}$ such that for some x_0 ,

$$\sup_{x_0\geq y\geq 0} \frac{y(\ell_\epsilon(x_0)-\ell_\epsilon(y))-\gamma(x_0-y)\ell_\epsilon(x_0)}{x_0\ell_\epsilon(x_0)} \geq \beta_\gamma(\mathcal{D}) - \frac{\epsilon(1-\beta_\gamma(\mathcal{D}))^2}{1-\epsilon(1-\beta_\gamma(\mathcal{D}))}$$

Setting $r_\epsilon = x_0$, and since o_1 maximizes (6), we obtain that

$$\frac{o_1(\ell_\epsilon(r_\epsilon)-\ell_\epsilon(o_1))-\gamma(r_\epsilon-o_1)\ell_\epsilon(r_\epsilon)}{r_\epsilon\ell_\epsilon(r_\epsilon)} \geq \beta_\gamma(D) - \frac{\epsilon(1-\beta_\gamma(\mathcal{D}))^2}{1-\epsilon(1-\beta_\gamma(\mathcal{D}))}$$

Therefore,

$$\mathrm{PoA}_\gamma(\mathcal{D}) \geq \mathrm{PoA}(\mathcal{G}^{\boldsymbol{\Gamma}}) \geq \frac{1}{1-\beta_\gamma(\mathcal{D}) + \frac{\epsilon(1-\beta_\gamma(\mathcal{D}))^2}{1-\epsilon(1-\beta_\gamma(D))}} = \rho_\gamma(\mathcal{D}) - \epsilon$$

□

A.5 Missing Technical Details of the Proof of Lemma 2

Formally, let $0 \leq x_{\max} \leq r_1$ be the maximum value (amount of flow) such that:

(a) There exist a γ-modification $\boldsymbol{\Gamma}'_1$ of subinstance $\mathcal{G}_1$ such that with traffic rate $r_1 - x_{\max}$, the Nash flow f'_1 of $\mathcal{G}_1^{\boldsymbol{\Gamma}'_1}$ is such that for any edge $e \in G_1$: if $f_{1e} > o_e$, then $f_{1e} \geq f'_{1e} \geq o_e$ and $\gamma'_e = \gamma_e (= \gamma)$, if $f_{1e} \leq o_e$ and $\gamma_e = 0$, then $f'_{1e} \leq f_{1e} (< o_e)$ and $\gamma'_e = \gamma_e (= 0)$, if $f_{1e} = o_e$ and $\gamma_e > 0$, then $f'_{1e} = f_{1e} (= o_e)$ and $0 \leq \gamma'_e \leq \gamma_e$.

(b) There exist a γ-modification $\boldsymbol{\Gamma}'_2$ of subinstance $\mathcal{G}_2$ such that with traffic rate $r_2 + x_{\max}$, the Nash flow f'_2 of $\mathcal{G}_2^{\boldsymbol{\Gamma}'_2}$ is such that for any edge $e \in G_2$: if $f_{2e} \geq o_e$ and $\gamma_e = \gamma$, then $f'_{2e} \geq f_{2e} (\geq o_e)$ and $\gamma'_e = \gamma_e (= \gamma)$, if $f_{2e} < o_e$, then $f_{2e} \leq f'_{2e} \leq o_e$ and $\gamma'_e = \gamma_e (= 0)$, if $f_{2e} = o_e$ and $\gamma_e < \gamma$, then $f'_{2e} = f_{2e} (= o_e)$ and $\gamma_e \leq \gamma'_e \leq \gamma$.

(c) $L(G_1, (1+\boldsymbol{\Gamma}'_1)\ell, r_1 - x_{\max}) \geq L(G_2, (1+\boldsymbol{\Gamma}'_2)\ell, r_2 + x_{\max})$

Claim. The set X of all x that satisfy (a), (b) and (c) is closed subset of $\mathbb{R}_{\geq 0}$ and thus the supremum lies inside X. Moreover, the maximum $x_{\max} \in X$ is positive and if $x_{\max} < r_1$, its corresponding γ-modification and Nash flows are such that for some edge e, either inequality (c) is tight or one of the *crucial* inequalities in (a) or (b) is tight, i.e. [$e \in G_1$ and ($f_{1e} > o_e$ and $f'_{1e} = o_e$) or ($f_{1e} = o_e$ and $\gamma_e > 0$ and $\gamma'_e = 0$)] or [$e \in G_2$ and ($f_{2e} < o_e$ and $f'_{2e} = o_e$) or ($f_{2e} = o_e$ and $\gamma_e < \gamma$ and $\gamma'_e = \gamma$)].

The proof is essentially a consequence of the more general results in [8, Section 3]. Since here we have a slightly different setting with restricted edge modifications, we give a formal proof of this claim at the section. Before we prove the claim, let us see that it indeed implies the lemma.

If inequality (c) is tight, then we are done as combining f'_1 with f'_2 and $\boldsymbol{\Gamma}'_1$ with $\boldsymbol{\Gamma}'_2$, we get the flow-moficiation pair $(f, \boldsymbol{\Gamma})$ for which properties (i) and (ii) hold (because of (a) and (b)).

If inequality (c) is not tight and $x_{\max} < r_1$, then we set $f_1 = f'_1$ and $f_2 = f'_2$, $r_1 = r_1 - x_{\max}$ and $r_2 = r_2 + x_{\max}$, $\boldsymbol{\Gamma}_1 = \boldsymbol{\Gamma}'_1$ and $\boldsymbol{\Gamma}_2 = \boldsymbol{\Gamma}'_2$ and repeat the procedure.

The above steps are finite and in the final step, inequality (c) holds with equality or G_1 is empty of flow ($x_{\max} = r_1$). To see this, let an edge be optimal under f if $f_e = o_e$ and observe that at any step (that ended without (c) being tight and without G_1 being empty of flow), because of the claim, at least one edge $e \in G_1$ got from non-optimal flow to optimal flow by losing flow, or got its $\gamma_e = 0$ and is "allowed" to become non-optimal again (in later steps) by losing flow[10], or at least one edge $e \in G_2$ got from non-optimal to

[10] Note that such edges will remain non-optimal, because in later steps, they may only lose more flow.

optimal by gaining flow, or got its $\gamma_e = \gamma$ and is "allowed" to become non-optimal again (in later steps) by gaining flow[11]. Thus, in a finite number of steps, if none of these steps ended with (c) tight, all the edges of G_1 will have $f'_{1e} \leq o_e$ and $\gamma_e = 0$ and all the edges of G_2 will have $f'_{2e} \geq o_e$ and $\gamma_e = \gamma$ and thus the rerouting may continue unrestricted (without any crucial inequality form (a) or (b)) and eventually stop either with $L(G_1, (1+\boldsymbol{\Gamma}'_1)\ell, r_1 - x_{\max}) = L(G_2, (1+\boldsymbol{\Gamma}'_2)\ell, r_2 + x_{\max})$ or with $x_{\max} = r_1$ (which leaves G_1 empty of flow).

The case where $L(G_1, (1+\boldsymbol{\Gamma}'_1)\ell, r_1 - x_{\max}) = L(G_2, (1+\boldsymbol{\Gamma}'_2)\ell, r_2 + x_{\max})$ is resolved earlier, the combination $\boldsymbol{\Gamma}$ of $\boldsymbol{\Gamma}'_1$ and $\boldsymbol{\Gamma}'_2$ together with the combination f of f'_1 and f'_2 satisfy properties (i) and (ii) of the lemma, because of conditions (a) and (b). This also works for the case where $x_{\max} = r_1$ and $L(G_1, (1+\boldsymbol{\Gamma}'_1)\ell, r_1 - x_{\max}) > L(G_2, (1+\boldsymbol{\Gamma}'_2)\ell, r_2 + x_{\max})$, again because of conditions (a) and (b) and because G_1 has no flow. Thus, to complete the proof of the lemma, we have only to prove the claim.

Proof of Claim. First we prove that: (I) if we have a flow f' and a γ-modification (combination of a $\boldsymbol{\Gamma}'_1$ and a $\boldsymbol{\Gamma}'_2$) that satisfy (a) and (b) of the set X and inequality (c), without any of the crucial inequalities of (a) and (b) or inequality (c) being tight, then we can reroute an amount of flow $\epsilon > 0$ (small enough) from G_1 to G_2 and change some of the γ'_e's of some edges of the edges with $f_e = o_e$ so as the new flow, combined with the new γ-modification also satisfies inequalities in (a) and (b) and inequality (c).

Consider graph $G_1^{\boldsymbol{\Gamma}'_1}$ and remove any edge e that has $f'_e = o_e$ and $0 < \gamma'_e$. Consider graph $G_2^{\boldsymbol{\Gamma}'_2}$ and remove any edge e that has $f'_e = o_e$ and $\gamma'_e < \gamma$. Call these graphs $\overline{G}_1^{\boldsymbol{\Gamma}'_1}$ and $\overline{G}_2^{\boldsymbol{\Gamma}'_2}$ respectively. Let the set of removed edges be E_r. Remove also the flow that goes through the paths containing the removed edges. Let h_e be the flow that is missing (was removed) from edge e because of the above flow removal ($h_e = 0$ for the edges that didn't lose flow). For all edges $e \in \overline{G}_1^{\boldsymbol{\Gamma}'_1}$ or $e \in \overline{G}_2^{\boldsymbol{\Gamma}'_2}$ let $\ell'_e(x) = \ell_e(x + h_e)$. Change all cost functions of $\overline{G}_1^{\boldsymbol{\Gamma}'_1}$ and $\overline{G}_2^{\boldsymbol{\Gamma}'_2}$ from $\ell_e(x)$ to $\ell'_e(x_e)$. Flow $f'' : f''_e = f'_e - h_e$ is a Nash flow for $\overline{G}_1^{\boldsymbol{\Gamma}'_1}$ and $\overline{G}_2^{\boldsymbol{\Gamma}'_2}$ (with the changed cost functions).

For an $\epsilon > 0$ small enough, because of continuity, we can suitably remove from $\overline{G}_1^{\boldsymbol{\Gamma}'_1}$ a flow of volume ϵ and push it through $\overline{G}_2^{\boldsymbol{\Gamma}'_2}$ resulting to f^*, so as under f^* all flow carrying paths in $\overline{G}_1^{\boldsymbol{\Gamma}'_1}$ share equal costs, all flow carrying paths in $\overline{G}_2^{\boldsymbol{\Gamma}'_2}$ share equal costs and for any any edge $e \in \overline{G}_1^{\boldsymbol{\Gamma}'_1}$ it is $(f''_e > o_e \Rightarrow f''_e \geq f^*_e \geq o_e)$ and $(f''_e \leq o_e \Rightarrow f^*_e \leq f''_e \leq o_e)$ and for any any edge $e \in \overline{G}_2^{\boldsymbol{\Gamma}'_2}$ it is $(f''_e \geq o_e \Rightarrow f^*_e \geq f''_e \geq o_e)$ and $(f''_e < o_e \Rightarrow f''_e \leq f^*_e \leq o_e)$.[12]

Now we put back again the edges that we removed from $G_1^{\boldsymbol{\Gamma}'_1}$ and $G_2^{\boldsymbol{\Gamma}'_2}$. We will put them back so as they get the same flow as in f' and, by changing their γ'_e value, they do not destroy the equilibrium property in none of the two networks.

If we put back the edges of $\overline{G}_1^{\boldsymbol{\Gamma}'_1}$ with the γ'_e they had initially, change the flow from f^* to $f^* + h$ and bring back the latency functions they had in $G_1^{\boldsymbol{\Gamma}'_1}$, then, because the cost of the Nash flow only may got smaller (in $G_1^{\boldsymbol{\Gamma}'_1}$), the costs of the paths that go through edges of E_r (in $G_1^{\boldsymbol{\Gamma}'_1}$) will have cost greater or equal to the Nash flow cost and if we put them back with $\gamma'_e = 0$, then the costs of the paths that go through edges of E_r (in $G_1^{\boldsymbol{\Gamma}'_1}$) will have cost smaller or equal to the Nash flow cost, for ϵ small enough. By induction on the decomposition of $G_1^{\boldsymbol{\Gamma}_1}$, by continuity and because $0 < \gamma'_e$ for edges in E_r and $G_1^{\boldsymbol{\Gamma}_1}$, there are $0 \leq \gamma''_e \leq \gamma'_e$ for all edges in E_r and $G_1^{\boldsymbol{\Gamma}'_1}$ so as $f^* + h$ is a Nash flow in $G_1^{\boldsymbol{\Gamma}''_1}$, where $\boldsymbol{\Gamma}''_1$ is the γ-modification that considers γ''_e's and agrees with $\boldsymbol{\Gamma}'_1$ in all other edges

Similarly, if we put back the edges of $\overline{G}_2^{\boldsymbol{\Gamma}'_2}$ with the γ'_e they had initially, change the flow from f^* to $f^* + h$ and bring back the latency functions they had in $G_2^{\boldsymbol{\Gamma}'_2}$, then, because the cost of the Nash flow only may got bigger (in $G_2^{\boldsymbol{\Gamma}'_2}$), the costs of the paths that goes through edges of E_r (in $G_2^{\boldsymbol{\Gamma}'_2}$) will have cost smaller

[11] Note that such edges will remain non-optimal, because in later steps, they may only gain more flow.

[12] For completeness, one can give a detailed proof of this statement by induction on the decomposition of $G_1^{\boldsymbol{\Gamma}_1}$ and $G_2^{\boldsymbol{\Gamma}_2}$.

or equal than the Nash flow cost and if we put them back with $\gamma'_e = \gamma$, then the costs of the paths that go through edges of E_r (in $G_2^{\boldsymbol{\Gamma}'_2}$) will have cost greater or equal to the Nash flow cost, for ϵ small enough. By induction on the decomposition of $G_2^{\boldsymbol{\Gamma}_2}$, by continuity and because $\gamma'_e < \gamma$ for edges in E_r and $G_2^{\boldsymbol{\Gamma}'_2}$, there are $\gamma'_e \leq \gamma''_e \leq \gamma$ for all edges in E_r and $G_2^{\boldsymbol{\Gamma}'_2}$ so as $f^* + h$ is a Nash flow in $G_2^{\boldsymbol{\Gamma}''_2}$, where $\boldsymbol{\Gamma}''_2$ is the γ-modification that considers γ'''_e's and agrees with $\boldsymbol{\Gamma}'_2$ in all other edges.

Initially inequality (c) was not tight and so for ϵ small enough (c) holds under $f^* + h$ and $\boldsymbol{\Gamma}''$.

Thus (I) is proved.

To prove that the supremum $x_{\max}$ of X is $x_{\max} > 0$ we set $\boldsymbol{\Gamma}' = \boldsymbol{\Gamma}$, $f'_1 = f_1$ and $f'_2 = f_2$ and we have an Nash flow f' (combination of f'_1 and f'_2) and a γ-modification (combination of a $\boldsymbol{\Gamma}'_1$ and a $\boldsymbol{\Gamma}'_2$) that satisfy (a) and (b) of the set X, without any of the crucial inequalities of (a) and (b) being tight and thus (I) can be applied.

To prove that $x_{\max} \in X$, one can check that all convergent sequences of X converge to a point inside X (or else there would be an x_0 close enough to $x_{\max}$, for which one of the inequalities of (a) or (b) wouldn't hold). Thus, X is a closed set.

Moreover, if $x_{\max} < r_1$, there exist a γ-modification $\boldsymbol{\Gamma}'$ that may differ from $\boldsymbol{\Gamma}$ only in the optimal edges such that either (c) is tight or it makes f'_1 the Nash flow of $G_1^{\boldsymbol{\Gamma}'_1}$ and f'_2 the Nash flow of $G_2^{\boldsymbol{\Gamma}'_2}$ with one crucial inequality being tight, i.e. there is an edge: [$e \in G_1^{\boldsymbol{\Gamma}'_1}$ with ($f_{1e} > o_e \,\&\, f'_{1e} = o_e$) or ($f_{1e} = o_e \,\&\, \gamma_e > 0 \,\&\, \gamma'_e = 0$)] or [$e \in G_2^{\boldsymbol{\Gamma}'_2}$ with ($f_{2e} < o_e \,\&\, f'_{2e} = o_e$) or ($f_{2e} = o_e \,\&\, \gamma_e < \gamma \,\&\, \gamma'_e = \gamma$)], with f'_1 and f'_2 differing from f_1 and f_2 only in the non optimal edges. This is true, as in a different case, by (I), we would be able to make $x_{\max}$ strictly greater (by pushing an extra $\epsilon > 0$ amount of flow from G_1 to G_2).

The last step concludes the proof of the claim and the proof of the lemma. □

A.6 The Proof of Lemma 3

Following the inductive proof of Lemma 2, we modify the optimal flow o from the leaves to the root of the series parallel decomposition tree of the series-parallel network G. As a result, at any node of the decomposition tree, we keep a flow and a γ-modification for the subnetwork corresponding to this node that satisfy properties (i) and (ii) of Lemma 2.

Let us first consider the series composition of subnetworks G_1 and G_2, for which, by induction, we have computed a γ-enforceable flow f_i and a modification $\boldsymbol{\Gamma}_i$, for $i \in \{1, 2\}$, that satisfy properties (i) and (ii) of Lemma 2. Then, we just combine f_1 with f_2 and $\boldsymbol{\Gamma}_1$ with $\boldsymbol{\Gamma}_2$ and obtain a γ-enforceable flow f and a modification $\boldsymbol{\Gamma}$ that satisfy properties (i) and (ii) of Lemma 2 for the subinstance corresponding to the composition of G_1 and G_2 in series.

For the parallel composition of subnetworks G_1 and G_2, again, by induction, we have computed, for $i \in \{1, 2\}$, a γ-enforceable flow f_i of rate r_i, with $r_1 + r_2 = r$, and a modification $\boldsymbol{\Gamma}_i$ that satisfy properties (i) and (ii) of Lemma 2. Then, we follow the steps in the proof of the claim, in proof of Lemma 2. We repeatedly compute $x_{\max}$ of the arising set X, and eventually (as proved in the claim), we obtain flow-modification pairs $(f'_1, \boldsymbol{\Gamma}'_1)$ and $(f'_2, \boldsymbol{\Gamma}'_2)$ that make (c) tight, or get G_1 empty of flow. Combining f_1 with f_2 and $\boldsymbol{\Gamma}_1$ with $\boldsymbol{\Gamma}_2$, we finally obtain a flow-modification pair $(f, \boldsymbol{\Gamma})$ for which properties (i) and (ii) of Lemma 2 are satisfied.

We compute $x_{\max}$ using binary search. Next, we discuss how to perform binary search for $x_{\max} \in X$ in the interval $[0, r_1]$. For $x = \frac{x_{\text{low}} + x_{\text{high}}}{2} \in [x_{\text{low}}, x_{\text{high}}]$, for a $[x_{\text{low}}, x_{\text{high}}] \subseteq [0, r_1]$, we compute the Nash flow in subinstance $\mathcal{G}_1^{\boldsymbol{\Gamma}_1}$ with flow $r_1 - x$ and in subinstance $\mathcal{G}_2^{\boldsymbol{\Gamma}_2}$ with flow $r_2 + x$, without the "locked" edges, i.e. edge set E_r. If for any edge, one of the inequalities of (a) or (b) is violated or if (c) is violated, then we move to the interval $[x_{\text{low}}, \frac{x_{\text{low}} + x_{\text{high}}}{2}]$. Otherwise, we compute (e.g, via Linear Programming) the modifications γ_e's for edges in E_r so that the equilibrium property is not destroyed. If for the edges in E_r, one of the inequalities of (a) or (b) is violated by some γ_e, then we move to the interval $[x_{\text{low}}, \frac{x_{\text{low}}+x_{\text{high}}}{2}]$, else we move to interval $[\frac{x_{\text{low}}+x_{\text{high}}}{2}, x_{\text{high}}]$. We continue in the same manner until we find an x for which (c) and all inequalities of (a) and (b) are satisfied and at least one of them is tight within a chosen accuracy parameter $\epsilon > 0$, i.e., it is practically tight, provided that ϵ is chosen small enough.

We next show how to find the suitable flow-modification pair $(f, \boldsymbol{\Gamma})$. Let f_1', $\boldsymbol{\Gamma}_1'$ and f_2', $\boldsymbol{\Gamma}_2'$ be the pairs found by the above binary search. If $x = r_1$ or (c) is tight, then we have found a flow f, that is a combination of f_1' and f_2', and a γ-modification $\boldsymbol{\Gamma}$, that is a combination of $\boldsymbol{\Gamma}_1'$ and $\boldsymbol{\Gamma}_2'$, which satisfy properties (i) and (ii) of Lemma 2. If inequality (c) is not tight and $x < r_1$, we set $f_1 = f_1'$ and $f_2 = f_2'$, $r_1 = r_1 - x$ and $r_2 = r_2 + x$, $\boldsymbol{\Gamma}_1 = \boldsymbol{\Gamma}_1'$ and $\boldsymbol{\Gamma}_2 = \boldsymbol{\Gamma}_2'$ and repeat the procedure.

The time complexity of the above steps is strongly related to the time complexity of computing the Nash flow for each x during the binary search process, which in turn, is closely related to the class of latency functions in $\mathcal{G}$. Let T_{NE} denote the time complexity of an algorithm that computes a Nash flows in series-parallel networks with the latency functions as in $\mathcal{G}$.

To compute a specific $x_{\max}$, at any step of the above procedure, the Nash flow algorithm is called $\log(r/\epsilon)$ times if we want to specify $x_{\max}$ within accuracy ϵ. A computation of $x_{\max}$ for parallel compositions is performed at most $2m$ times (see also proof of Lemma 2), since the decomposition tree of G has at most m nodes and thus, at most m parallel compositions occur during the whole procedure. Thus, assuming the time complexity T_{NE} of the Nash flow algorithm dominates the time required for the computation of the edge modifications $\gamma_e' s$ of edges in E_r's, we get a total running time of $O(m^2 T_{\text{NE}} \log(r/\epsilon))$, as required by prove Lemma 3. □

A.7 The Proof of Theorem 7

We observe that $\{\boldsymbol{\Gamma}_0 | \boldsymbol{\Gamma}_0$ is a γ_0-modification of $\mathcal{G}\} \subseteq \{\boldsymbol{\Gamma}_p | \boldsymbol{\Gamma}_p$ is a (p,γ)-modification of $\mathcal{G}\}$, because $\|\boldsymbol{\Gamma}_0\|_p \leq \gamma$ for every γ_0-modification $\boldsymbol{\Gamma}_0$. Therefore, by definition, we have that

$$\begin{aligned}\text{PoA}_\gamma^p(\mathcal{G}) &= \min\{\text{PoA}(\mathcal{G}^{\boldsymbol{\Gamma}_p}) | \boldsymbol{\Gamma}_p \text{ is a } (p,\gamma)\text{-modification of } \mathcal{G}\} \\ &\leq \min\{\text{PoA}(\mathcal{G}^{\boldsymbol{\Gamma}_0}) | \boldsymbol{\Gamma}_0 \text{ is a } \gamma_0\text{-modification of } \mathcal{G}\} \\ &= \text{PoA}_{\gamma_0}(\mathcal{G})\end{aligned}$$

Theorem 1 provides the second part of the inequality and completes the proof. □

A.8 Theorem 8: The Proof of Claim 1

To begin with, there is a function $\ell \in \mathcal{D}$ with $\ell(0) = 0$ because $\mathcal{D}$ is of the form $\mathcal{D}_0$ and thus by assumption $\forall\, g(x) \in \mathcal{D} \Rightarrow \ell(x) := (g(x) - g(0)) \in \mathcal{D}$ and $\ell(0) = 0$.

Since the difference between $\text{PoA}_\gamma^p(\mathcal{I}_m)$ and $\text{PoA}_{\gamma_1}(\mathcal{I}_m)$ lies in the cost of the best possible equilibrium it is sufficient to show that the best equilibrium of $\mathcal{I}_m$, in terms of social cost, under any (p,γ)-modification and under any γ_1-modification coincide.

To do so, let us consider the convex program that minimizes the social cost objective among feasible flows in equilibrium. We only care for equilibrium that can route flow through e_m. If we know a priory that the last edge can not receive flow the claim holds trivially. So the equilibrium condition is $(1+\gamma_e)\ell(f_e) = L$, $\forall e$, where $L = (1+\gamma_1)\ell(\frac{r}{m-1})$, i.e. the cost of the last edge.

$$\min\, C(f) = \sum_{e \in E} f_e \ell_e(f_e)$$

$$\sum_{e \in E} f_e - r = 0 \quad (h) \tag{7}$$

$$-f_e \leq 0 \qquad \forall e \in E \quad (g_e) \tag{8}$$

$$\sqrt[p]{\sum_{e \in A} \gamma_e^p} - \gamma \leq 0 \quad (\nu) \tag{9}$$

$$-\gamma_e \leq 0 \qquad \forall e \in A \quad (\mu_e) \tag{10}$$

$$(1+\gamma_e)\ell(f_e) - L = 0 \qquad \forall e \in A \quad (\omega_e) \tag{11}$$

where $A = E \setminus \{e_m\}$, meaning the set of edges with non-constant latency functions.

Obviously, there is no point in modifying the cost of the last edge because it has a constant cost so increasing it can only worsen the quality of the equilibrium. So all the (p, γ) andγ-modifications only affect the first $m-1$ edges but not the last one.

The stationarity conditions together with dual feasibility are given by

$$\ell(f_e) + f_e \ell'(f_e) + h - g_e + \omega_e(1+\gamma_e)\ell'(f_e) = 0 \qquad \forall e \in A \tag{12}$$

$$L + h - g_{e_m} = 0 \tag{13}$$

$$\nu \left(\frac{\gamma_e}{\sqrt[p]{\sum_{e \in A} \gamma_e^p}} \right)^{p-1} - \mu_e + \omega_e \ell(f_e) = 0 \qquad \forall e \in A \tag{14}$$

$$g_e, \mu_e, \nu \geq 0 \tag{15}$$

and the complementary slackness conditions are given by

$$g_e f_e = 0 \qquad \forall e \in E \tag{16}$$

$$\mu_e \gamma_e = 0 \qquad \forall e \in A \tag{17}$$

$$\nu \left(\sqrt[p]{\sum_{e \in A} \gamma_e^p} - \gamma \right) = 0 \tag{18}$$

If $g_{e_m} > 0$ then from (16) we get that $f_{e_m} = 0$ and as we already mentioned the claim holds trivially. So we now focus in the case with $g_{e_m} = 0$.

Since $\ell(0) = 0$ there is no feasible solution with $f_e = 0$, $e \in A$ because then (11) cannot hold. Thus, from complementary slackness $g_e = 0$, $\forall e \in A$.

Additionally from the stationarity conditions we get:

- $h = -L$.
- $\omega_e = \frac{L - \ell(f_e) - f_e \ell'(f_e)}{(1+\gamma_e)\ell'(f_e)}$, where $\ell'(f_e) > 0$ since $\mathcal{D}$ is of the form $\mathcal{D}_0$ and ℓ is not a constant.

If there exists $\mu_e > 0$ for some e, then from (17) we get $\gamma_e = 0$. So from (14), (15) and because $\ell(f_e) > 0$ since $f_e > 0$, we get that $\omega_e > 0$. But, since $\gamma_e = 0$, from (11) we get that $\ell(f_e) = L \Rightarrow \omega_e < 0$, which is a contradiction. Thus, $\mu_e = 0$, $\forall e \in A$.

Now, if $\gamma_e = 0$ for some e, then $\omega_e = 0$. So $\ell(f_e) + f_e \ell'(f_e) = L = (1+\gamma_e)\ell(f_e) \Rightarrow f_e \ell'(f_e) = 0$ which is impossible since $f_e, \ell'(f_e) > 0$. Thus, $\gamma_e > 0$, $\forall e \in A$.

Additionally if $\nu = 0 \Rightarrow \omega_e = 0$, $\forall e$ and $\ell(f_e) > 0$ because $f_e > 0$. But in that case $\ell(f_e) + f_e \ell'(f_e) = L \Rightarrow f_{e_i} = f_{e_j}$, $\forall e_i, e_j \in A$ and thus $f_e \leq \frac{r}{m-1}$. So, from (11) $\gamma_e = \gamma_1$ and $f_e = \frac{r}{m-1}$, $\forall e \in A$ and the claim holds.

It remains to examine the case with $\nu > 0 \Rightarrow \omega_e < 0$, $\forall e$.

From (18) it is $\sqrt[p]{\sum_{e\in A}\gamma_e^p}=\gamma$, so from (14) we get that $\nu=-\omega_e\ell(f_e)\left(\frac{\gamma}{\gamma_e}\right)^{p-1}$. We use the equilibrium condition of primal feasibility, (11), to eliminate γ_e and we get that $\nu=\ell^{p+1}(f_e)\frac{\ell(f_e)+f_e\ell'(f_e)-L}{L\ell'(f_e)}\frac{\gamma^{p-1}}{\left(L-\ell(f_e)\right)^{p-1}}$.

Now, let us symbolize with f_0 the flow so that $\ell(f_0)+f_0\ell'(f_0)=L$. Since $\omega_e<0$, $\forall e\in A$ we know that $f_e>f_0$, $\forall e\in A$. Note that this implies that $r>(m-1)f_0$. If $r\leq(m-1)f_0$ then we know a priory that no flow can be routed through e_m in equilibrium and as we already mentioned the proof of the claim is trivial.

Now we will show that $N(f_e)=\ell^{p+1}(f_e)\frac{\ell(f_e)+f_e\ell'(f_e)-L}{L\ell'(f_e)}\frac{\gamma^{p-1}}{\left(L-\ell(f_e)\right)^{p-1}}$ is strictly increasing and thus one-to-one function for $f_e\geq f_0$. It is trivial to see that $N_1(f_e)=\ell^{p+1}(f_e)$ and $N_2(f_e)=\frac{\gamma^{p-1}}{\left(L-\ell(f_e)\right)^{p-1}}$ are strictly increasing for strictly increasing ℓ and $f_e>0$. For $N_3(f_e)=\frac{\ell(f_e)+f_e\ell'(f_e)-L}{L\ell'(f_e)}$ we have $N_3'(f_e)=\frac{2\left(\ell'(f_e)\right)^2+\ell''(f_e)(L-\ell(f_e))}{L\left(\ell'(f_e)\right)^2}$.

We will exploit the fact that ℓ is semi-convex to prove that $N_3'(f_e)\geq0$, $\forall f_e\geq f_0$. So, we know that $(x\ell(x))''=2\ell'(x)+x\ell''(x)\geq0\Leftrightarrow\ell''(x)\geq-\frac{2\ell'(x)}{x}$ for $x>0$. Now, since $\ell(x)\leq L$ because of equilibrium condition we get that $2\left(\ell'(f_e)\right)^2+\ell''(f_e)(L-\ell(f_e))\geq2\left(\ell'(f_e)\right)^2-\frac{2\ell'(x)}{x}(L-\ell(f_e))=\frac{2\ell'(f_e)}{f_e}\left(\ell(f_e)+f_e\ell'(f_e)-L\right)$. We only care about the sign of $\ell(f_e)+f_e\ell'(f_e)-L$ since $\frac{2\ell'(f_e)}{f_e}\geq0$, $\forall f_e>0$ because ℓ is strictly increasing. Now remember that $\ell(f_0)+f_0\ell'(f_0)=L$ and since $f_e\ell(f_e)$ is convex it follows that $\left(f_e\ell(f_e)\right)'=\ell(f_e)+f_e\ell'(f_e)$ is increasing. So for $f_e>f_0\Rightarrow\ell(f_e)+f_e\ell'(f_e)\geq\ell(f_0)+f_0\ell'(f_0)=L\Rightarrow\ell(f_e)+f_e\ell'(f_e)-L\geq0\Rightarrow N_3'(f_e)\geq0$.

Now $N(f_e)=N_1(f_e)N_2(f_e)N_3(f_e)$ and $N_1(f_0),N_2(f_0),N_3(f_0)\geq0$ so their product is also strictly increasing for $f_e\geq f_0$.

Demanding $N(f_e)=\nu$, $\forall e\in A$ and respecting primal feasibility, we get that the only available solution is $f_e=\frac{r}{m-1}$, $\forall e\in A$, $f_m=0$ and $\gamma_e=\gamma_1=\frac{\gamma}{\sqrt[p]{m-1}}$, $\forall e\in A$, because N is one-to-one function.

So, finally we proved that the vector $\boldsymbol{u_0}=((f_e)_{e\in E},(\gamma_e)_{e\in A})=(\frac{r}{m-1},\ldots,\frac{r}{m-1},0,\gamma_1,\ldots,\gamma_1)$ is the only KKT point of the mathematical program stated above. Since $C(f)$ is a continuous function in a compact set we know that it receives a minimum value. Apart from that the set $\{\nabla h(\boldsymbol{u_0}),\nabla g_m(\boldsymbol{u_0}),\nabla\nu(\boldsymbol{u_0}),\nabla\omega_e(\boldsymbol{u_0})\}$ is linearly independent for any $m\geq3$, so we know that the necessary KKT conditions hold in the point the minimizes $C(f)$. Thus $\boldsymbol{u_0}$ minimizes $C(f)$.

The case for $m=2$ is trivial and does not require KKT conditions.

Now note that $\boldsymbol{u_0}$ can be achieved by a γ_1-modification, and since it consists the best (p,γ)-modification it is also the best γ_1-modification. Thus $\mathrm{PoA}_\gamma^p(\mathcal{I}_m)=\mathrm{PoA}_{\gamma_1}(\mathcal{I}_m)$. □

A.9 Theorem 8: The Proof of Claim 2

Let $\epsilon>0$ and consider a γ_1-modifiable instance $\mathcal{G}$ of m parallel links, flow rate $r_{\epsilon,m}$ (to be fixed later) of aversion type $a^1=1$ and cost functions: an arbitrary cost function $\ell_1(x)=\ell_{\epsilon,m}(x)$ (to be fixed later) in class $\mathcal{D}$ for the first $m-1$ links and the constant function $\ell_2(x)=(1+\gamma_1)\ell_{\epsilon,m}(\frac{r_{\epsilon,m}}{m-1})$ for the last link.

Let f^{min} denote the best Nash flow under any γ_1-modification of $\mathcal{I}_m$. In f^{min} the last edge cannot carry flow, so the optimal configuration is to split equally the flow in the rest of the edges. Thus $C(f^{min})=r_{\epsilon,m}\ell_{\epsilon,m}(\frac{r}{m-1})$.

Additionally, let o denote the optimal flow for $\mathcal{I}_m$. The first $m-1$ edges receive equal amount of flow denoted by o_1 and the rest of the flow is routed through the last edge. That is because the first $m-1$ edges have the same marginal cost tolls, $\ell_{\epsilon,m}(x)+x\ell_{\epsilon,m}'(x)$. So to obtain equilibrium we need to route the same amount of traffic, o_1 through every edge. Then

$$\text{PoA}_{\gamma_1}(\mathcal{I}_m) = \frac{r_{\epsilon,m}\ell_{\epsilon,m}(\frac{r_{\epsilon,m}}{m-1})}{(m-1)o_1\ell_{\epsilon,m}(o_1) + [r_{\epsilon,m} - (m-1)o_1](1+\gamma_1)\ell_{\epsilon,m}(\frac{r_{\epsilon,m}}{m-1})}$$

Setting $u = \frac{r_{\epsilon,m}}{m-1}$ we simplify the notation and we get

$$\text{PoA}_{\gamma_1}(\mathcal{I}_m) = \frac{1}{1 - \frac{o_1(\ell_{\epsilon,m}(u) - \ell_{\epsilon,m}(o_1)) - \gamma_1(u-o_1)\ell_{\epsilon,m}(u)}{u\ell_{\epsilon,m}(u)}}$$

Flow o is optimal and thus $o_1 \in [0, \frac{r_{\epsilon,m}}{m-1}]$ is exactly the value that minimizes social cost and eventually maximizes $\text{PoA}_{\gamma_1}(\mathcal{I}_m)$ and so maximizes

$$\frac{o_1(\ell_{\epsilon,m}(u) - \ell_{\epsilon,m}(o_1)) - \gamma_1(u-o_1)\ell_{\epsilon,m}(u)}{u\ell_{\epsilon,m}(u)} \tag{19}$$

Recall that $\beta_{\gamma_1}(\mathcal{D}) = \sup_{\ell \in \mathcal{D}, x \geq y \geq 0} \frac{y(\ell(x)-\ell(y)) - \gamma_1(x-y)\ell(x)}{x\ell(x)}$. Using the definition of sup, let $\ell_{\epsilon,m}$ be a cost function such that for some x_0 it is $\sup_{x_0 \geq y \geq 0} \frac{y(\ell_{\epsilon,m}(x_0) - \ell_{\epsilon,m}(y)) - \gamma_1(x_0-y)\ell_{\epsilon,m}(x_0)}{x_0\ell_{\epsilon,m}(x_0)} \geq \beta_{\gamma_1}(\mathcal{D}) - \frac{\epsilon/2(1-\beta_{\gamma_1}(\mathcal{D}))^2}{1-\epsilon/2(1-\beta_{\gamma_1}(\mathcal{D}))}$.

Additionally, we can assume that $\ell_{\epsilon,m}(0) = 0$ because if $\ell_{\epsilon,m}(0) > 0$ then for $g_{\epsilon,m}(x) := \ell_{\epsilon,m}(x) - \ell_{\epsilon,m}(0)$ we have $\sup_{x_0 \geq y \geq 0} \frac{y(g_{\epsilon,m}(x_0) - g_{\epsilon,m}(y)) - \gamma_1(x_0-y)g_{\epsilon,m}(x_0)}{x_0 g_{\epsilon,m}(x_0)} \geq \sup_{x_0 \geq y \geq 0} \frac{y(\ell_{\epsilon,m}(x_0) - \ell_{\epsilon,m}(y)) - \gamma_1(x_0-y)\ell_{\epsilon,m}(x_0)}{x_0\ell_{\epsilon,m}(x_0)}$ and $g_{\epsilon,m}(0) = 0$.

Setting $r_{\epsilon,m} = x_0$, and because o_1 maximizes (19), it is $\frac{o_1(\ell_{\epsilon,m}(u) - \ell_{\epsilon,m}(o_1)) - \gamma_1(u-o_1)\ell_{\epsilon,m}(u)}{u\ell_{\epsilon,m}(u)} \geq \beta_{\gamma_1}(\mathcal{D}) - \frac{\epsilon/2(1-\beta_{\gamma_1}(\mathcal{D}))^2}{1-\epsilon/2(1-\beta_{\gamma_1}(\mathcal{D}))}$ and thus

$$\text{PoA}_{\gamma_1}(\mathcal{I}_m) \geq \frac{1}{1 - \beta_{\gamma_1}(\mathcal{D}) + \frac{\epsilon/2(1-\beta_{\gamma_1}(\mathcal{D}))^2}{1-\epsilon/2(1-\beta_{\gamma_1}(\mathcal{D}))}} = \rho_{\gamma_1}(\mathcal{D}) - \epsilon/2$$

□

A.10 Theorem 8: The Proof of Claim 3

It is $\beta_{\gamma_0}(\mathcal{D}) = \sup_{\ell \in \mathcal{D}, x \geq y \geq 0} \frac{y(\ell(x)-\ell(y)) - \gamma_0(x-y)\ell(x)}{x\ell(x)} =$

$$\sup_{\ell \in \mathcal{D}, x \geq y \geq 0} \frac{y(\ell(x)-\ell(y)) - \gamma_1(x-y)\ell(x) + \gamma_1(x-y)\ell(x) - \gamma_0(x-y)\ell(x)}{x\ell(x)} =$$

$$\sup_{\ell \in \mathcal{D}, x \geq y \geq 0} \Big(\frac{y(\ell(x)-\ell(y)) - \gamma_1(x-y)\ell(x)}{x\ell(x)} + \frac{(\gamma_1-\gamma_0)(x-y)}{x} \Big) \leq$$

$$\sup_{\ell \in \mathcal{D}, x \geq y \geq 0} \frac{y(\ell(x)-\ell(y)) - \gamma_1(x-y)\ell(x)}{x\ell(x)} + \sup_{\ell \in \mathcal{D}, x \geq y \geq 0} \frac{(\gamma_1-\gamma_0)(x-y)}{x} = \beta_{\gamma_1}(\mathcal{D}) + (\gamma_1 - \gamma_0)$$

Let $\delta(m) := \gamma_1 - \gamma_0 = \frac{\gamma}{\sqrt[p]{m-1}} - \frac{\gamma}{\sqrt[p]{m}} = \gamma \frac{m^{1/p} - (m-1)^{1/p}}{[m(m-1)]^{1/p}}, m \geq 2$.

Now we can write clearly the previous inequality as

$$\beta_{\gamma_1}(\mathcal{D}) \geq \beta_{\gamma_0}(\mathcal{D}) - \delta(m)$$

$$\Updownarrow$$

$$\frac{1}{1-\beta_{\gamma_1}(\mathcal{D})} \geq \frac{1}{1-\beta_{\gamma_0}+\delta(m)} \tag{20}$$

$\delta(m)$ is strictly decreasing for $m \geq 2$ and $\lim_{m\to+\infty}\delta(x) = 0$. So, by definition $\forall\,\eta > 0\,\exists m_0$: $\delta(m_0) \leq \eta$. Selecting $\eta := \eta(\epsilon) = \frac{\epsilon/2(1-\beta_{\gamma_0}(\mathcal{D}))^2}{1-\epsilon/2(1-\beta_{\gamma_0}(\mathcal{D}))}$, for ϵ small enough so that $\eta(\epsilon) > 0$ for a given γ, we get that:

For any $\epsilon > 0$, small enough there is $m_\epsilon \geq 2$ such that $\delta(m_\epsilon) \leq \eta(\epsilon)$.

Inequality (20) for $m = m_\epsilon$ becomes

$$\frac{1}{1-\beta_{\gamma_1}(\mathcal{D})} \geq \frac{1}{1-\beta_{\gamma_0}+\delta(m_\epsilon)} \geq \frac{1}{1-\beta_{\gamma_0}+\eta(\epsilon)} = \frac{1}{1-\beta_{\gamma_0}} - \frac{\epsilon}{2} \Longleftrightarrow \rho_{\gamma_1}(\mathcal{D}) \geq \rho_{\gamma_0}(\mathcal{D}) - \epsilon/2$$

$\square$