Improving Approximate Pure Nash Equilibria in Congestion Games

Vipin Ravindran Vijayalakshmi[1(✉)] and Alexander Skopalik[2]

[1] Chair of Management Science, RWTH Aachen, Aachen, Germany
vipin.rv@oms.rwth-aachen.de
[2] Mathematics of Operations Research, University of Twente, Enschede, Netherlands
a.skopalik@utwente.nl

Abstract. Congestion games constitute an important class of games to model resource allocation by different users. As computing an exact [18] or even an approximate [34] pure Nash equilibrium is in general PLS-complete, Caragiannis et al. [9] present a polynomial-time algorithm that computes a $(2 + \epsilon)$-approximate pure Nash equilibria for games with linear cost functions and further results for polynomial cost functions. We show that this factor can be improved to $(1.61 + \epsilon)$ and further improved results for polynomial cost functions, by a seemingly simple modification to their algorithm by allowing for the cost functions used during the best response dynamics be different from the overall objective function. Interestingly, our modification to the algorithm also extends to efficiently computing improved approximate pure Nash equilibria in games with arbitrary non-decreasing resource cost functions. Additionally, our analysis exhibits an interesting method to optimally compute universal load dependent taxes and using linear programming duality prove tight bounds on the PoA under universal taxation, e.g., 2.012 for linear congestion games and further results for polynomial cost functions. Although our approach yield weaker results than that in Bilò and Vinci [6], we remark that our cost functions are locally computable and in contrast to [6] are independent of the actual instance of the game.

Keywords: Congestion games · Approximate pure Nash equilibria · Price of anarchy · Universal taxes

1 Introduction

Congestion games constitute an important class of games that succinctly represents a game theoretic model for resource allocation among non-cooperative users. A canonical example for this is the road transportation network, where the time needed to commute is a function on the total amount of traffic in the network. A congestion game is a cost minimization game defined by a set of

V. R. Vijayalakshmi—This work is supported by the German research council (DFG) Research Training Group 2236 UnRAVeL

X. Chen et al. (Eds.): WINE 2020, LNCS 12495, pp. 280–294, 2020.
https://doi.org/10.1007/978-3-030-64946-3_20

resources E, a set of N players with strategies $S_1, \ldots, S_N \subseteq 2^E$, and for each resource $e \in E$, a cost function $f_e : \mathbb{N} \mapsto \mathbb{R}_+$. Congestion games were first introduced by Rosenthal [28], and using a potential function argument proved that it belongs to a class of games in which a pure Nash equilibrium always exists, i.e., the game always consists of a self-emerging solution in which no user is able to improve by unilaterally deviating.

Convergence to Pure Nash Equilibria. Fabrikant et al. [18] show that computing a pure Nash equilibrium is PLS-complete. They show that regardless of the order in which local search is performed, there are initial states from where it could take exponential number of steps before the game converges to a pure Nash equilibrium. Also, they show PLS-completeness for network congestion games with asymmetric strategy spaces. As a positive result, Fabrikant et al. [18] present a polynomial time algorithm to compute a pure Nash equilibrium in certain restricted strategy spaces e.g., symmetric network congestion games. Ackermann et al. [1] show that network congestion games with linear cost functions are PLS-complete. However, if the set of strategies of each player consists of the bases of a matroid over the set of resources, then they show that the lengths of all best response sequences are polynomially bounded in the number of players and resources. This alludes to studying *approximate* pure Nash equilibria in congestion games.

To our knowledge, the concept of α-approximate equilibria[1] was introduced by Roughgarden and Tardos [29] in the context of non-atomic selfish routing games. An α-approximate pure Nash equilibrium is a state in which none of the users can unilaterally deviate to improve by a factor of at least α. Orlin et al. [25] show that every local search problem in PLS admits a fully polynomial time ϵ-approximation scheme. Although their approach can be applied to congestion games, this does not yield an approximate pure Nash equilibrium, but rather only an approximate local optimum of the potential function. In case of congestion games, Skopalik and Vöcking [34] show that in general for arbitrary cost functions, finding a α-approximate pure Nash equilibrium is PLS-complete, for any $\alpha > 1$. However, for polynomial cost function (with non-negative coefficients) of maximum degree d, Caragiannis et al. [9] present an approximation algorithm. They present a polynomial-time algorithm that computes $(2 + \epsilon)$-approximate pure Nash equilibria for games with linear cost functions and an approximation guarantee of $d^{O(d)}$ for polynomial cost functions of maximum degree d. Interestingly, they use the convergence of subsets of players to a $(1 + \epsilon)$-approximate Nash equilibrium (of that subset) as a subroutine to generate a state which is an approximation of the minimal potential function value (of that subset), e.g. $2 \cdot \textsc{opt}$ for linear congestion games. This approximation factor of the minimal potential then essentially turns into the approximation factor of the approximate equilibrium. Feldotto et al. [19] using a *path-cycle decomposition* technique bound this approximation factor of the potential for arbitrary cost functions.

[1] Here we refer to the multiplicative notion of approximation. There is also a additive variant which is often denoted by ϵ-Nash.

Our Contribution. In this paper we improve the approximation guarantee achieved in the computation of approximate pure Nash equilibrium with the algorithm in Caragiannis et al. [9], using a linear programming approach which generalizes the smoothness condition in Roughgarden [30], to modify the cost functions that users experience in the algorithm. Although we only make a seemingly simple modification to their algorithm in [9], we would like to remark that the analysis is significantly involved, and does not immediately follow from [9], since the sub-game induced by the algorithm with the modified costs is not a potential game anymore. Table 1 lists the results for resource cost functions that are bounded degree polynomials of maximum degree d. Our main contribution in this paper is presented as Theorem 1.

Table 1. Approximate pure Nash equilibria of congestion games with polynomial cost functions of degree at most d.

d	Previous Approx. [9,19]	Our Approx. $\rho_d + \epsilon$
1	$2 + \epsilon$	$1.61 + \epsilon$
2	$6 + \epsilon$	$3.35 + \epsilon$
3	$20 + \epsilon$	$8.60 + \epsilon$
4	$111 + \epsilon$	$27.46 + \epsilon$
5	$571 + \epsilon$	$98.14 + \epsilon$

Theorem 1. *For every $\epsilon > 0$, the algorithm computes a $(\rho_d + \epsilon)$-approximate equilibrium for every congestion game with non-decreasing cost functions that are polynomials of maximum degree d in a number of steps which is polynomial in the number of players and $1/\epsilon$.*

Our approach also yields a simple and distributed method to compute load dependent universal taxes that improves the inefficiency of equilibria in congestion games. Table 2 lists our results for the price of anarchy (PoA) under refundable taxation for resource cost functions that are bounded degree polynomials. Bilò and Vinci [6] present an algorithm to compute load dependent taxes that improve the price of anarchy e.g., for linear congestion games from 2.5 to 2. Although our methods yield slightly weaker results, our cost functions are locally computable and in contrast to [6] are independent of the actual instance of the game. Furthermore, using linear programming duality we derive a reduction to a selfish scheduling game on identical machines, which implies a matching lower bound on the approximation factor. We would like to remark that our results for PoA were achieved independently of that in Paccagnan et al. [26] by a very similar technique.

2 Preliminaries

A strategic game denoted by the tuple $\left(\mathcal{N}, (S_u)_{u\in\mathcal{N}}, (c_u)_{u\in\mathcal{N}}\right)$ consists of a finite set of players $\mathcal{N}$, and for each player $u \in \mathcal{N}$, a finite set of strategies S_u and

Table 2. PoA under taxation in congestion games with polynomial cost functions of degree at most d.

d	PoA without taxes Aland et al. [2]	Optimal taxes Bilò and Vinci [6]	Universal taxes Ψ_d Local search w.r.t ζ_{SC}
1	2.5	2	2.012
2	9.583	5	5.10
3	41.54	15	15.56
4	267.6	52	55.46
5	1514	203	220.41

a cost function $c_u : S \to \mathbb{R}_+$ mapping a state $s \in S := S_1 \times S_2 \times \cdots \times S_N$ to the cost of player $u \in \mathcal{N}$. A congestion game is a strategic game that succinctly represents a decentralized resource allocation problem involving selfish users.

A congestion game denoted by $G = \left(\mathcal{N}, E, (S_u)_{u\in N}, (f_e)_{e\in E}\right)$ consists of a set of N players, $\mathcal{N} = \{1, 2, \ldots, N\}$, who compete over a set of resources E. Each player $u \in \mathcal{N}$ has a set of strategies denoted by $S_u \subseteq 2^E$. Each resource $e \in E$ has a non-negative and non-decreasing cost function $f_e : \mathbb{N} \mapsto \mathbb{R}_+$ associated with it. Let $n_e(s)$ denote the number of players on a resource $e \in E$ in the state s, then the cost contributed by a resource $e \in E$ to each player using it is denoted by $f_e(n_e(s))$. Therefore, the cost of a player $u \in \mathcal{N}$ in a state $s = (s_1, \ldots, s_N)$ of the game is given by $c_u(s) = \sum_{e\in E: e\in s_u} f_e(n_e(s))$. For a state s, $c_u(s'_u, s_{-u})$ denotes the cost of player u, when only u deviates. A best-response move denoted by $\mathcal{BR}_u(s)$ is a move that minimizes a player's cost while all the other players are fixed to their strategy in s. With some abuse of notation, $\mathcal{BR}_u(0)$ denotes the best response of a player u assuming that no other player participates in the game.

A state $s \in S$ is a pure Nash equilibrium (PNE), if there exists no player who could deviate to another strategy and decrease their cost, i.e., $\forall u \in \mathcal{N}$, and $\forall s'_u \in S_u$, $c_u(s) \leq c_u(s'_u, s_{-u})$. A weaker notion of PNE is the α-approximate pure Nash equilibrium for $\alpha \geq 1$, which is a state s in which no player has an improvement that decreases their cost by a factor of at least α, i.e, $\forall u \in \mathcal{N}$, and $\forall s'_u \in S_u$ $\alpha \cdot c_u(s'_u, s_{-u}) \geq c_u(s)$. For congestion games the exact potential function $\phi(s) = \sum_{e\in E} \sum_{i=1}^{n_e(s)} f_e(i)$, guarantees the existence of a PNE by proving that every sequence of unilateral improving strategies converges to a PNE. We denote social or global cost of a state s as $c(s) = \sum_{u\in\mathcal{N}} c_u(s)$ and the state that minimizes social cost is called the optimal, i.e., $s^* = \arg\min_{s\in S} c(s)$. The inefficiency of equilibria is measured using the price of anarchy (PoA) [22], which is the worst case ratio between the social cost of an equilibrium and the social optimum.

A local optimum is a state s in which there is no player $u \in \mathcal{N}$ with an alternative strategy s'_u such that $c(s'_u, s_{-u}) < c(s)$, and an α-approximate local optimum is a state s in which there is no player u who has an α-move with a strategy s'_u such that $\alpha \cdot c(s'_u, s_{-u}) < c(s)$. Let us remark that there is an interesting connection between a local optimum and a PNE. A PNE is a local optimum of the potential function ϕ, and similarly, a local optimum is a Nash equilibrium of a game in which we change the resource cost functions from $f(x)$ to the marginal contribution to social cost, e.g., to $f'(x) = xf(x) - (x-1)f(x-1)$. Analogous to the PoA, the *stretch* of a congestion game is the worst case ratio between the value of the potential function at an equilibrium and the potential minimizer.

3 Approximate Equilibria in Congestion Games

In this section we aim at improving the approximation factor of an approximate pure Nash equilibria in congestion games with arbitrary non-decreasing resource cost functions. We extend an algorithm based on Caragiannis et al. [9] to compute an approximate pure Nash equilibrium in congestion games with polynomial cost functions with non-negative coefficients. A key element of this algorithm is the so called stretch of a (sub-) game. This is the worst case ratio of the potential function at an equilibrium and the global minimum of the potential.

This algorithm generates a sequence of improving moves that converges to an approximate PNE in polynomial number of best-response moves. The idea is to divide the players into blocks based on their costs and hence their prospective ability to drop the potential of the game. In each phase of the algorithm, players of two consecutive blocks are scheduled to make improving moves starting with the blocks of players with high costs. One block only makes q-moves, which are improvements by a factor of at least q which is close to 1. The other block does p-moves, where p is slightly larger than the stretch of a q-approximate equilibrium, and slightly smaller than the final approximation factor.

The key idea here is that blocks first converge to a q-approximate equilibrium, and thereby generate a state with a stretch of approximately p. Later, when players of a block are allowed to do p-moves, there is not much potential left to move. In particular, there is no significant influence on players of blocks that moved earlier possible. This finally results in the approximation factor of roughly p. We modify the algorithm in [9] by changing the cost seen by the players during their q-moves to be the modified cost generated using a linear programming approach, to achieve a significantly smaller stretch, and this results in an improved approximation factor. For the sake of completeness we present the algorithm as Algorithm 1, but note that only the definition of $\theta(q)$ using $\lambda := \max_{e \in E} \lambda_e$, the definition of p in Line 1, and the use of the modified cost functions in Line 11 has been changed. Before we analyze the correctness of the algorithm, we describe how the modified cost functions can be computed.

Modified Cost Functions

After a long series of papers in which various authors (e.g. [2,3,14]) show upper bounds on the price of anarchy, Roughgarden exhibited that most of them essentially used the same technique, which is formalized as (λ, μ)-smoothness [30]. A game is called (λ, μ)-smooth, if for every pair of outcomes s, s^*, it holds that,

Algorithm 1. Computing a $\lambda(1+\epsilon)$-approximate pure Nash equilibria in congestion games.

Input: Congestion game $G = \left(\mathcal{N}, E, (S_u)_{u\in\mathcal{N}}, (f_e)_{e\in E}\right)$ and $\epsilon > 0$.
Output: A state of G in $\lambda(1+\epsilon)$-approximate pure Nash equilibrium.

Set $q = \left(1 + \frac{1}{N^c}\right)$, $p = \left(\frac{1}{\theta(q)} - \frac{1+q+2\lambda}{N^c}\right)^{-1}$, $c = 10\log\left(\frac{\lambda}{\epsilon}\right)$, $\Delta = \max_{e\in E}\frac{f_e(N)}{f_e(1)}$ and $\theta(q) = \frac{\lambda}{1+\frac{1-q}{q}N\lambda}$, where $\lambda := \max_{e\in E}\lambda_e$
foreach $u \in \mathcal{N}$ **do**
  set $\ell_u = c_u\left(\mathcal{BR}_u(0)\right)$;
end for
Set $\ell_{min} = \min_{u\in\mathcal{N}}\ell_u$, $\ell_{max} = \max_{u\in\mathcal{N}}\ell_u$ and $\hat{z} = 1 + \lceil \log_{2\Delta N^{2c+2}}\left(\ell_{max}/\ell_{min}\right)\rceil$;
Assign players to blocks $B_1, B_2, \cdots, B_{\hat{z}}$ such that $u \in B_i \Leftrightarrow \ell_u \in \left(\ell_{max}\left(2\Delta N^{2c+2}\right)^{-i}, \ell_{max}\left(2\Delta N^{2c+2}\right)^{-i+1}\right]$;
foreach $u \in N$ **do**
  set the player u to play the strategy $s_u \leftarrow \mathcal{BR}_u(0)$;
end for
for phase $i \leftarrow 1$ to $\hat{z}-1$ such that $B_i \neq \emptyset$ **do**
  while $\exists u \in B_i$ with a p-move w.r.t the original cost f or $\exists u \in B_{i+1}$ with a q-move w.r.t to modified cost f' **do**
    u deviates to that best-response strategy $s_u \leftarrow \mathcal{BR}(s_1, \cdots, s_n)$.
  end while
end for

$\sum_{u\in\mathcal{N}} c_u(s_u^*, s_{-u}) \leq \lambda \cdot c(s^*) + \mu \cdot c(s)$. The price of anarchy of a (λ,μ)-smooth game with $\lambda > 0$ and $\mu < 1$ is then at most $\frac{\lambda}{1-\mu}$. Observe that the original smoothness definition can be extended to allow for an arbitrary objective function $h(s)$ instead of the social cost function $c(s) = \sum_{u\in\mathcal{N}} c_u(s)$.

Definition 1. *A game is* (λ,μ)*-smooth* with respect to an objective function h, *if for every pair of outcome* s, s^*, $\lambda\cdot h(s^*) \geq \sum_{u\in\mathcal{N}} c_u(s_u^*, s_{-u}) - \sum_{u\in\mathcal{N}} c_u(s) + (1-\mu)h(s)$.

From the definition above, we restate the central smoothness theorem [30].

Theorem 2. *Given a* (λ,μ)*-smooth game* G *with* $\lambda > 0$, $\mu < 1$, *and an objective function* h, *then for every equilibrium* s *and the global optimum* s^*, $h(s) \leq \frac{\lambda}{1-\mu}h(s^*)$.

The proof is analogous to Roughgarden's proof [30]. We remark that a variant to our extension of Roughgarden's smoothness framework is independently introduced as *generalized smoothness* in [11–13].

In the following we study games in which we change the cost functions c_u experienced by the players. By scaling the cost functions appropriately, we always ensure that we can satisfy the above inequality with $\mu = 0$. Observe that, given a game $G = \left(\mathcal{N}, (S_u)_{u\in\mathcal{N}}, (c_u)_{u\in\mathcal{N}}\right)$, we can determine new cost functions c'_u for which the value of λ is minimized, for all pairs of solutions s, s^*. However, observe that since the state space S grows exponentially in the number of players, this would be computationally inefficient. Therefore, we typically have to work with games in which the players' costs and the objective function h can be represented in a succinct way. In congestion games, the players cost and the global objective function are implicitly defined by the resource cost function. In the following, we allow for an arbitrary, additive objective function $h(s)$, i.e., of the form $h(s) = \sum_{e \in E} h_e(n_e(s))$, and we can conveniently restate the smoothness condition as follows.

Lemma 1. *A congestion game is* $(\lambda, 0)$*-smooth with respect to an objective function* $h(s) = \sum_{e \in E} h_e(n_e(s))$*, if for every non-decreasing cost function* $f'_e : \mathbb{N} \mapsto \mathbb{R}_+$ *and for every* $0 \leq n, m \leq N$*,* $\lambda \cdot h_e(m) \geq m f'_e(n+1) - n f'_e(n) + h_e(n)$*.*

Lemma 1 immediately gives rise to the following optimization problem: Given an objective function $h(s) = \sum_{e \in E} h_e(n_e(s))$, find functions f'_e that minimize λ. For a resource objective function h_e and a bound on the number of players N this can be easily solved by the following linear program LP_h with the variables $f'_e(0), \ldots, f'_e(N+1)$ and λ_e.

$$\begin{aligned} &\min \lambda_e \\ \lambda_e \cdot h_e(m) - m f'_e(n+1) + n f'_e(n) &\geq h_e(n) \qquad \text{for all } n \in [0, N], m \in [0, N] \\ f'_e(n+1) &\geq f'_e(n) \qquad \text{for all } n \in [0, N] \\ f'_e(n) &\geq 0 \qquad \text{for all } n \in [0, N+1] \end{aligned}$$

Henceforth, we use $f' = (f'_e)_{e \in E}$ whenever we refer to cost functions that are the solution to an optimization problem and denote the players cost by $c'_u(s) = \sum_{e \in s_u} f'_e(n_e(s))$. Observe that LP_h is compact, i.e, the number of constraints and variables are polynomially bounded in the number of players. Hence, we state the following theorem.

Theorem 3. *Optimal resource cost functions* f'_e *for objective functions* h_e *can be computed in polynomial time.*

Improving the Approximation Factor

In order to achieve a better approximation factor than that in Caragiannis et al. [9] we modify the algorithm in [9] by changing the cost functions seen by the players during their q-moves to be the modified cost generated by the linear program LP_ϕ arising from Lemma 1 with the potential as its objective function. This results in an improved approximation factor $\lambda(1+\epsilon)$ for $\epsilon > 0$, where $\lambda := \max_{e \in E} \lambda_e$ is the optimal solution value of LP_ϕ that we state below. Unfortunately, it is not possible to simply use the LP above with the potential function as its objective function, since Algorithm 1 uses the potential function argument for a subset of players $F \subseteq \mathcal{N}$. More precisely, it needs that the approximation factor also holds for an arbitrary subset of players and its induced subgame. Let us denote by $n_e^F(s)$ the number of players in F that use the resource e in the state s. Define the potential of this subset as the potential in the subgame induced by these players in s, i.e, $\phi^F(s) := \sum_{i=1}^{n_e^F(s)} f_e(i + n_e^{\mathcal{N} \setminus F}(s))$. With slight abuse of notation, we remark that $\phi^F(s)$ and $\phi_F(s)$ are equivalent. Now consider an arbitrary subset of players $F \subseteq \mathcal{N}$ and a state s.

Then, $G_s^F := (F, E, (S_u)_{u \in F}, (f_e^F)_{e \in E})$ is the subgame induced by freezing the remaining players from $\mathcal{N} \setminus F$, with $f_e^F(x) := f_e(x + n_e^{\mathcal{N} \setminus F}(s))$, where $n_e^{\mathcal{N} \setminus F}(s)$ is the number of players outside of F on resource e in the state s. Henceforth, for our purposes, the following definition is a stronger notion of the $(\lambda, 0)$-smoothness.

Definition 2. *A strategic game is strongly* $(\lambda, 0)$*-smooth with respect to an objective function* h*, and for some* $\lambda > 0$*, if for every subset* $F \subseteq \mathcal{N}$ *and for every* $s, s^* \in S$*,* $\lambda \cdot h^F(s^*) \geq \sum_{u \in F} c'_u(s^*_u, s_{-u}) - \sum_{u \in F} c'_u(s) + h^F(s)$*, where* $h^F(s) := \sum_{e \in E} h_e(n_e(s)) - h_e(n_e^{\mathcal{N} \setminus F}(s))$*.*

We would like to remark that all future references to $(\lambda, 0)$-smoothness in Sect. 3 imply strong $(\lambda, 0)$-smoothness. As a consequence of Definition 2, we state the following lemma.

Lemma 2. *For every congestion game* G *with non-decreasing cost functions* $f'_e : \mathbb{N} \mapsto \mathbb{R}_+$*, which is* $(\lambda, 0)$*-smooth with respect to the potential function* ϕ_e *for every subgame* G_s^F *induced by an arbitrary subset* $F \subseteq \mathcal{N}$*, and arbitrary states* $s, s^* \in S$*, i.e.,* $\lambda \cdot \phi_e^F(s^*) - n_e^F(s^*) \cdot f'_e(n_e(s) + 1) + n_e^F(s) \cdot f'_e(n_e(s)) \geq \phi_e^F(s)$*, with* $\lambda > 0$*, is also strongly* $(\lambda, 0)$*-smooth.*

This subset property is of particular importance for the algorithm to compute an approximate equilibrium, but may be of independent interest as well. We are not aware of other approximation algorithms that can guarantee this property as well. From Lemma 2, for any resource $e \in E$, the modified cost functions f'_e are computed by the following linear program LP_ϕ.

$$\min \lambda_e$$

$$\begin{aligned} \lambda_e \sum_{i=z+1}^{m+z} f_e(i) - m f'_e(n+z+1) + n f'_e(n+z) &\geq \sum_{i=z+1}^{n+z} f_e(i) && \forall (n+z), m \in [0, N] \\ f'_e(n+1) &\geq f'_e(n) && \forall n \in [0, N] \\ f'_e(n) &\geq 0 && \forall n \in [0, N+1] \end{aligned}$$

We are now ready to prove Theorem 1, by restating it as follows.

Theorem 4. *For every constant* $\epsilon > 0$*, Algorithm 1 computes a* $\lambda(1+\epsilon)$*-approximate equilibrium for every congestion game with non-decreasing cost functions, and* $\lambda = \max_{e \in E} \lambda_e$*, in number of steps which is polynomial in the number of players,* $\Delta := \frac{f(N)}{f(1)}$ *and* $1/\epsilon$*.*

The proof of the theorem follows the proof scheme of Caragiannis et al. [9], which we have to rework to accommodate for our modification. The complete proof of the theorem is omitted due to space constraints (see full version [33]). Note that for cost functions which are polynomials of maximum degree d with non negative coefficients, Δ is polynomial in the number of players. In the following, we sketch the main proof idea. Here, we have to take into account that the game played by the players from $B_i \cup B_{i+1}$ in phase i is no longer a potential game as the players use different cost functions. However, we can show that the strong smoothness constraints of the LP guarantees that the values of the new cost functions can be conveniently bounded.

Lemma 3. *Let f' to be the modified cost functions generated by the LP_ϕ and f to be the original cost functions. Then for all $i \geq 1$, $f_e(i) \leq f'_e(i) \leq \lambda f_e(i)$.*

To bound the stretch of any (sub-)game in a q-approximate equilibrium the following lemma is useful. We remark that for this lemma, the property that the induced subgames are also smooth (Lemma 2) is crucial.

Lemma 4. *Let s be any q-approximate equilibrium with respect to the modified cost function, and s^* be a strategy profile with minimal potential. Then for every $F \subseteq \mathcal{N}$, $\phi_F(s) \leq \theta(q) \cdot \phi_F(s^*)$.*

We now bound the potential of the set of players $R_i \subseteq B_i \cup B_{i+1}$ that move in phase i. Most importantly, the players of B_i, were in an q-approximate equilibrium with respect to c'_u at the end of the previous round. Hence, for every subset of B_i, we can exploit Lemma 4 to obtain a small upper bound on the potential amongst players R_i participating in a phase i at the beginning of the phase. For a phase i, let $b_i := \ell_{max} \left(2\Delta N^{2c+2}\right)^{-i+1}$ and s^i denote the state of the game after the execution of phase i.

Lemma 5. *For every phase $i \geq 2$, it holds that $\phi_{R_i}(s^{i-1}) \leq \frac{b_i}{N^c}$.*

To analyze convergence, we have to take into account the fact that players use different latency functions. However, it turns out that the Rosenthal potential with respect to the modified cost functions can serve as an approximate potential function, i.e., it also decreases for the p-moves of players using the original cost functions.

Lemma 6. *Let $u \in \mathcal{N}$ be a player that makes a p-move with respect to the original cost function f. Then, $p \cdot c_u(s'_u, s_{-u}) - c_u(s) \geq q \cdot c'_u(s'_u, s_{-u}) - c'_u(s)$, where c_u and c'_u are the cost of the player u with respect to f and f', respectively.*

Using Lemma 5 and Lemma 6, we can bound the runtime which has to be slightly larger and has to depend on Δ to allow for arbitrary non-decreasing functions.

Lemma 7. *The algorithm terminates after at most $\mathcal{O}(\lambda \Delta^3 N^{5c+5})$ best-response moves.*

The next lemma shows that when players involved in phases $i \geq 2$ make their moves, they do not increase the cost of players in the blocks $B_1, B_2, \cdots, B_{i-1}$ significantly.

Lemma 8. *Let u be a player in the block B_t, where $t \leq \hat{z}-2$. Let s'_u be a strategy different from the one assigned to u by the algorithm at the end of the phase t. Then, for each phase $i \geq t$, it holds that, $c_u(s^i) \leq p \cdot c_u(s'_u, s^i_{-u}) + \frac{2p+1}{N^c} \sum_{k=t+1}^{i} b_k$.*

As no players' costs and alternatives is significantly influenced by moves in later blocks, they remain in an approximate equilibrium which can be used to finally prove the correctness of the algorithm.

Lemma 9. *The state computed by the algorithm is a* $p\left(1+\frac{5}{N^c}\right)$*-approximate equilibrium.*

Linear and Polynomial Cost Functions. We now turn to the important class of polynomial cost functions with non-negative coefficients. We can show that for polynomials of small degree, it is sufficient to restrict the attention to the first $K = 150$ values of the cost functions. Hence, we only need to solve a linear program of constant size. The following lemma states that for the larger values of n and appropriate values of λ_d and ν, we can easily obtain $(\lambda_d, 0)$-smoothness by choosing $f'(n) = \nu n^d$. We further note, that for a given $\lambda_d > 0$, and for each n and z we only need to consider a limited range for m.

Lemma 10. *For* $d \leq 5$ *and* $n \geq 150$*, the function* $f'(n) = \nu n^d$ *with* $\nu = \sqrt[d+1]{\lambda_d}$ *is* $(\lambda_d, 0)$*-smooth with respect to the potential function* ϕ *for an appropriate* λ_d*.*

Lemma 11. *For fixed* n, z*, if* $\lambda_d \cdot \sum_{i=z+1}^{m+z} i^d - mf'(n+z+1) + nf'(n+z) \geq \sum_{i=z+1}^{n+z} i^d$ *is true* $\forall m \leq (n+z+1)^2(d+1)$*, it also holds* $\forall m > (n+z+1)^2(d+1)$*.*

By Lemma 10 and 11 it remains to solve a linear program of constant size to obtain our results ρ_d as listed in Table 1 for $d \leq 5$.

Corollary 1. *For every congestion game with polynomial cost functions of degree* $d \leq 5$*, and for every constant* $\epsilon > 0$*, the algorithm computes a* $(\rho_d + \epsilon)$*-approximate pure Nash equilibrium in polynomial time.*

Lower Bound. Any feasible solution to the linear program LP_h emerging from Lemma 1 are cost functions $f'_e : \mathbb{N} \mapsto R_+$ that guarantees that the objective value associated with the function h is at most λ. We can show that this is in fact optimal. That is, LP_h is not only optimizing the smoothness inequality, but also that there exists no other resource cost function that can guarantee a smaller objective value than λ. To that end, we consider the dual of LP_h (LPD_h) and show that for every feasible solution of the dual, we can construct an instance of a selfish scheduling game on identical machines with an objective value that is equal to the value of the dual LP solution, regardless of the actual cost function of the game.

Lemma 12. *Every optimal solution of* LPD_h *with objective value* λ *can be turned into an instance of selfish scheduling on identical machines with an objective value of* $\lambda - \epsilon$ *for an arbitrary* $\epsilon > 0$*.*

4 Extensions

The smoothness framework introduced by Roughgarden [30] also extends to equilibrium concepts such as mixed Nash and (coarse) correlated equilibria. The same is true for our variant with respect to an arbitrary objective function h. We now look at an extension of Lemma 1 for computing load dependent universal taxes in congestion games.

Load Dependent Universal Taxes. One of the many approaches used to improve the PoA is the introduction of taxes. For a set of resources E, the load dependent tax function t, is the excess cost incurred by the user on a resource $e \in E$ with cost $f(x)$, e.g., $f'(x) = f(x) + t(x)$. We remark that the taxes we consider in this work are refundable, and do not contribute to the overall cost of the game.

Meyers and Schulz [24] studied the complexity of computing an optimal solution in a congestion game and prove NP-hardness. Makarychev and Sviridenko [23] give the best known approximation algorithm using randomized rounding on a natural feasibility LP with approximation factor $\mathcal{B}_{d+1}$ which is the $d+1^{\text{th}}$ Bell number, where d is the maximum degree of the polynomial cost function. Interestingly, the same was later achieved using load dependent taxes by Bilò and Vinci [6], where they apply the *primal-dual* method [4] to upper bound the PoA under refundable taxation in congestion games. They determine a load specific taxation to show that the PoA is at most $[O(d/\log d)]^{d+1}$ under refundable taxation. However, we remark that the load dependent taxes computed in [6] aren't universal, i.e, they are sensitive to the instance of the game.

We give a rather simple approach to locally (on resource) compute load dependent *universal* taxes. Table 2 lists the improved PoA bounds under refundable taxation using our technique for congestion games with resource cost functions that are bounded degree polynomials of maximum degree d. By the smoothness argument [30] the new bounds immediately extends to mixed, (coarse) correlated equilibria and outcomes generated by no-regret sequences. Moreover, since the linear program that computes the cost or tax function does only depend on the original cost function of that resource, the computed taxes are robust against perturbations of the instance such as adding or removing of resources or players.

Optimal Universal Taxes. We seek to compute universal load dependent taxes that minimize the PoA under refundable taxation. We consider the following optimization problem. For an objective function $h(s) = \sum_{e \in E} n_e(s) \cdot f_e(n_e(s))$, find functions f'_e that satisfies Lemma 1 minimizing λ. For a resource objective function $h_e(n_e(s)) = n_e(s) \cdot f_e(n_e(s))$ and a bound on the number of players N, this can be easily solved by the following linear program LP_{SC} with the variables $f'_e(0), \ldots, f'_e(N+1)$ and λ_e.

$$\min \lambda_e$$
$$\lambda_e \cdot h_e(m) - m f'_e(n+1) + n f'_e(n) \geq h_e(n) \qquad \text{for all } n \in [0, N], m \in [0, N]$$

$$f'_e(n+1) \geq f'_e(n) \qquad \forall n \in [0, N]$$
$$f'_e(n) \geq 0 \qquad \text{for all } n \in [0, N+1]$$

Observe that we can solve LP_{SC} locally for each resource with cost function f_e. For the LP solution λ_e and $f'_e(n)$, define the tax function as $t_e(n) := f'_e(n) - f_e(n)$. The resulting price of anarchy under taxation is then $\lambda := \max_{e \in E} \lambda_e$. From Lemma 12 we remark that the taxes computed by LP_{SC} are optimal. Evidently our lower bound of 2.012 for congestion games with linear cost functions matches the price of anarchy bound for selfish scheduling games on identical machines [10]. For any (distributed) local search algorithm (such as Bjelde et al. [8]) that seeks to minimizes the social cost $c(s) = \sum_{e \in E} n_e(s) f_e(n_e(s))$, we define $\zeta_{\text{SC}}(s) := \sum_{e \in E} \sum_{i=1}^{n_e(s)} f'_e(i)$ as a pseudo-potential function. Then, from Lemma 1 it is guaranteed that every local optimum with respect to $\zeta_{\text{SC}}(s)$ has an approximation factor of at most $\lambda := \max_{e \in E} \lambda_e$ with respect to the social cost $c(s)$. Using approximate local search by Orlin et al. [25], we can compute a solution close to that in polynomial time, and more so to state the following.

Corollary 2. *For every congestion game the ϵ-local search algorithm using $\zeta_{\text{SC}}(s)$, produces a $\lambda(1+\epsilon)$ local optimum in running time polynomial in the input length and $1/\epsilon$.*

Linear and Polynomial Cost Functions. For the interesting case of polynomial resource cost functions of maximum degree d, similar to Sect. 3, we show that for polynomials of small degree, it is sufficient to restrict the attention to the first 1154 values of the cost functions. Hence, we only need solve a linear program of constant size. We further note, that for a fixed $\lambda_d > 0$, and for each n we only need to consider a limited range for m in the LP_{SC}.

Lemma 13. *For $d \leq 5$ and $n \geq 1154$, the function $f'(n) = \nu n^d$ with $\nu = \sqrt[d+1]{(d+1)\lambda_d}$ is $(\lambda_d, 0)$-smooth with respect to $h(n) = n^{d+1}$ and an appropriate λ_d.*

Lemma 14. *For a fixed n, if $\lambda_d \cdot m^{d+1} - mf(n+1) + nf(n) \geq n^{d+1}$ is true for all $m \leq (n+1)^2$, it also holds for all $m > (n+1)^2$.*

As a consequence of Lemma 13 and 14 it only remains to solve a linear program of constant size for each $d \leq 5$ to obtain our results Ψ_d (listed in Table 2). Our results match the recent results that were obtained independently by Paccagnan et al. [26].

Corollary 3. *For every congestion game with polynomial cost functions of degree $d \leq 5$, each cost function f'_e can be computed in constant time and the resulting game is $(\Psi_d, 0)$-smooth with respect to social cost.*

5 Conclusion and Open Problems

The most interesting question which was the initial motivation for this work is the complexity of approximate equilibria. We find it very surprising that the technique yields such a significant improvement, e.g., for linear congestion games from 2 to 1.61, by using essentially the same algorithm of Caragiannis et al. [9]. However, the algorithmic technique is limited only by the lower bound for approximation factor of the stretch implied in Roughgarden [31]. Hence, further significant improvements may need new algorithmic ideas. On the lower bound side, not much is known for linear or polynomial congestion games. The only computational lower bound for approximate equilibria is from Skopalik and Vöcking [34] using unnatural, and very steep cost functions.

We believe that the technique of perturbing the instance of an (optimization) problem such that a simple local search heuristic (or an equilibrium) guarantees an improved approximation ratio can be applied in other settings as well. It would be interesting to see, whether one can achieve similar results for variants and generalizations of congestion games such as weighted [3], atomic- or integer-splittable [27,32] congestion games, scheduling games [16,17,20], etc. Considering other heuristics such as greedy or one-round walks [5,7,15,21] would be another natural direction.

References

1. Ackermann, H., Röglin, H., Vöcking, B.: On the impact of combinatorial structure on congestion games. J. ACM **55**(6), 25:1–25:22 (2008)
2. Aland, S., Dumrauf, D., Gairing, M., Monien, B., Schoppmann, F.: Exact price of anarchy for polynomial congestion games. SIAM J. Comput. **40**(5) (2011)
3. Awerbuch, B., Azar, Y., Epstein, A.: The price of routing unsplittable flow. In: Proceedings of the Thirty-Seventh Annual ACM Symposium on Theory of Computing. STOC (2005)
4. Bilò, V.: A unifying tool for bounding the quality of non-cooperative solutions in weighted congestion games. Theor. Comp. Syst. **62**(5), 1288–1317 (2018). https://doi.org/10.1007/978-3-642-38016-7_18
5. Bilò, V., Fanelli, A., Flammini, M., Moscardelli, L.: Performance of one-round walks in linear congestion games. Theory Comput. Syst. **49**(1), 24–45 (2011). https://doi.org/10.1007/s00224-010-9309-0
6. Bilò, V., Vinci, C.: Dynamic taxes for polynomial congestion games. In: Proceedings of the 2016 ACM Conference on Economics and Computation. EC (2016)
7. Bilò, V., Vinci, C.: On the impact of singleton strategies in congestion games. In: 25th Annual European Symposium on Algorithms (ESA) (2017)
8. Bjelde, A., Klimm, M., Schmand, D.: Brief announcement: approximation algorithms for unsplittable resource allocation problems with diseconomies of scale. In: Proceedings of the 29th ACM Symposium on Parallelism in Algorithms and Architectures. SPAA 2017, ACM, New York, NY, USA (2017)
9. Caragiannis, I., Fanelli, A., Gravin, N., Skopalik, A.: Efficient computation of approximate pure Nash equilibria in congestion games. In: Proceedings of the 2011 IEEE 52nd Annual Symposium on Foundations of Computer Science. FOCS (2011)
10. Caragiannis, I., Flammini, M., Kaklamanis, C., Kanellopoulos, P., Moscardelli, L.: Tight bounds for selfish and greedy load balancing. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006. LNCS, vol. 4051, pp. 311–322. Springer, Heidelberg (2006). https://doi.org/10.1007/11786986_28

11. Chandan, R., Paccagnan, D., Marden, J.R.: When smoothness is not enough: toward exact quantification and optimization of the price-of-anarchy. In: 2019 IEEE 58th Conference on Decision and Control (CDC) (2019)
12. Chandan, R., Paccagnan, D., Ferguson, B.L., Marden, J.R.: Computing optimal taxes in atomic congestion games. In: Proceedings of the 14th Workshop on the Economics of Networks, Systems and Computation. NetEcon (2019)
13. Chandan, R., Paccagnan, D., Marden, J.R.: Optimal mechanisms for distributed resource-allocation. Preprint (2019). http://arxiv.org/abs/1911.07823
14. Christodoulou, G., Koutsoupias, E.: The price of anarchy of finite congestion games. In: Proceedings of the Thirty-seventh Annual ACM Symposium on Theory of Computing. STOC 2005, ACM, New York, NY, USA (2005)
15. Christodoulou, G., Mirrokni, V.S., Sidiropoulos, A.: Convergence and approximation in potential games. Theoret. Comput. Sci. **438** (2012)
16. Cole, R., Correa, J., Gkatzelis, V., Mirrokni, V., Olver, N.: Decentralized utilitarian mechanisms for scheduling games. Games Econ. Behav. **92**, 306–326 (2015)
17. Correa, J., Queyranne, M.: Efficiency of equilibria in restricted uniform machine scheduling with total weighted completion time as social cost. Naval Res. Logistics **59**(5), 384–395 (2012)
18. Fabrikant, A., Papadimitriou, C., Talwar, K.: The complexity of pure Nash equilibria. In: Proceedings of the Thirty-Sixth Annual ACM Symposium on Theory of Computing. STOC 2004, ACM, New York, NY, USA (2004)
19. Feldotto, M., Gairing, M., Skopalik, A.: Bounding the potential function in congestion games and approximate pure Nash equilibria. In: Liu, T.-Y., Qi, Q., Ye, Y. (eds.) WINE 2014. LNCS, vol. 8877, pp. 30–43. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-13129-0_3
20. Gairing, M., Lücking, T., Mavronicolas, M., Monien, B.: Computing Nash equilibria for scheduling on restricted parallel links. Theory of Computing Systems **47**(2), 405–435 (2010). https://doi.org/10.1007/s00224-009-9191-9
21. Klimm, M., Schmand, D., Tönnis, A.: The online best reply algorithm for resource allocation problems. In: Fotakis, D., Markakis, E. (eds.) SAGT 2019. LNCS, vol. 11801, pp. 200–215. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-30473-7_14
22. Koutsoupias, E., Papadimitriou, C.: Worst-case equilibria. In: Proceedings of the 16th Annual Conference on Theoretical Aspects of Computer Science. STACS (1999)
23. Makarychev, K., Sviridenko, M.: Solving optimization problems with diseconomies of scale via decoupling. In: Proceedings of the 2014 IEEE 55th Annual Symposium on Foundations of Computer Science. FOCS (2014)
24. Meyers, C.A., Schulz, A.S.: The complexity of welfare maximization in congestion games. Networks **59**(2), 252–260 (2012)
25. Orlin, J.B., Punnen, A.P., Schulz, A.S.: Approximate local search in combinatorial optimization. In: Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms. SODA (2004)
26. Paccagnan, D., Chandan, R., Ferguson, B.L., Marden, J.R.: Incentivizing efficient use of shared infrastructure: optimal tolls in congestion games. Preprint (2020). http://arxiv.org/abs/1911.09806
27. Rosenthal, R.W.: The network equilibrium problem in integers. Networks **3**(1), 53–59 (1973)
28. Rosenthal, R.W.: A class of games possessing pure-strategy Nash equilibria. Int. J. Game Theory **2**(1), 65–67 (1973). https://doi.org/10.1007/BF01737559

29. Roughgarden, T., Tardos, E.: How bad is selfish routing? In: Proceedings 41st Annual Symposium on Foundations of Computer Science, pp. 93–102, November 2000
30. Roughgarden, T.: Intrinsic robustness of the price of anarchy. In: Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing. STOC (2009)
31. Roughgarden, T.: Barriers to near-optimal equilibria. In: Proceedings of the 2014 IEEE 55th Annual Symposium on Foundations of Computer Science. FOCS (2014)
32. Roughgarden, T., Schoppmann, F.: Local smoothness and the price of anarchy in atomic splittable congestion games. In: Proceedings of the Twenty-second Annual ACM-SIAM Symposium on Discrete Algorithms. SODA (2011)
33. Skopalik, A., Vijayalakshmi, V.R.: Improving approximate pure Nash equilibria in congestion games. Preprint (2020). https://arxiv.org/abs/2007.15520
34. Skopalik, A., Vöcking, B.: Inapproximability of pure Nash equilibria. In: Proceedings of the 40th Annual ACM Symposium on Theory of Computing. STOC (2008)