Efficient computation of approximate pure Nash equilibria in congestion games[‡]

Ioannis Caragiannis*, Angelo Fanelli†, Nick Gravin† and Alexander Skopalik†
*Research Academic Computer Technology Institute & Department of Computer Engineering and Informatics, University of Patras, 26500 Rio, Greece.
Email: caragian@ceid.upatras.gr
†Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore.
Email: angelo.fanelli@ntu.edu.sg, ngravin@gmail.com, skopalik@cs.rwth-aachen.de

Abstract— Congestion games constitute an important class of games in which computing an exact or even approximate pure Nash equilibrium is in general PLS-complete. We present a surprisingly simple polynomial-time algorithm that computes $O(1)$-approximate Nash equilibria in these games. In particular, for congestion games with linear latency functions, our algorithm computes $(2+\epsilon)$-approximate pure Nash equilibria in time polynomial in the number of players, the number of resources and $1/\epsilon$. It also applies to games with polynomial latency functions with constant maximum degree d; there, the approximation guarantee is $d^{O(d)}$. The algorithm essentially identifies a polynomially long sequence of best-response moves that lead to an approximate equilibrium; the existence of such short sequences is interesting in itself. These are the first positive algorithmic results for approximate equilibria in non-symmetric congestion games. We strengthen them further by proving that, for congestion games that deviate from our mild assumptions, computing ρ-approximate equilibria is PLS-complete for any polynomial-time computable ρ.

1. Introduction

Among other solution concepts, the notion of the pure Nash equilibrium plays a central role in Game Theory. It characterizes situations with non-cooperative deterministic players in which no player has any incentive to unilaterally deviate from the current situation in order to achieve a higher payoff. Questions related to their existence and efficient computation have been extensively addressed in the context of congestion games. In these games, pure Nash equilibria are guaranteed to exist through potential function arguments: any pure Nash equilibrium corresponds to a local minimum of a potential function. Unfortunately, this proof of existence is inefficient and computing a local minimum for this function is a computationally-hard task. This statement has been made formal in the work of Fabrikant et al. [14] where it is proved that the problem of computing a pure Nash equilibrium is PLS-complete.

Such negative complexity results significantly question the importance of pure Nash equilibria as solution concepts that characterize the behavior of rational players. Approximate pure Nash equilibria, which characterize situations where no player can *significantly improve* her payoff by unilaterally deviating from her current strategy, could serve as alternative solution concepts[1] provided that they can be computed efficiently. In this paper, we study the complexity of computation of approximate pure Nash equilibria in congestion games and prove the first positive algorithmic results for important (and quite general) classes of congestion games. Our main result is a polynomial-time algorithm that computes $O(1)$-approximate pure Nash equilibria in congestion games under mild restrictions.

1.1. Problem statement and related work

Congestion games were introduced by Rosenthal [20]. In a congestion game, players compete over a set of resources. Each resource incurs a latency to all players that use it; this latency depends on the number of players that use the resource according to a resource-specific, non-negative, and non-decreasing latency function. Among a given set of strategies (over sets of resources), each player aims to select one selfishly, trying to minimize her individual total cost, i.e., the sum of the latencies on the resources in her strategy. Typical examples include network congestion games where the network links correspond to the resources and each player has alternative paths that connect two nodes as strategies. Congestions games in which players have the same set of available strategies are called symmetric.

Rosenthal [20] proved that congestion games admit a potential function with the following remarkable property: the difference in the potential value between two states (i.e., two snapshots of strategies) that differ in the strategy of a single player equals to the difference of the cost experienced by this player in these two states. This immediately implies the existence of a pure Nash equilibrium. Any sequence

‡ This work was partially supported by the grant NRF-RF2009-08 "Algorithmic aspects of coalitional games" and the EC-funded STREP Project FP7-ICT-258307 EULER.

[1]Actually, approximate pure Nash equilibria may be more desirable as solution concepts in practical decision making settings since they can accommodate small modeling inaccuracies due to uncertainty (e.g., see the arguments in [9]).

of improvement moves by the players strictly decreases the value of the potential and a state corresponding to a local minimum of the potential will eventually be reached; this corresponds to a pure Nash equilibrium. Monderer and Shapley [18] proved that any game that admits such a cost-revealing (or exact) potential function is isomorphic to a congestion game.

The existence of a potential function allows us to view the problem of computing a pure Nash equilibrium as a local search problem [13], i.e., as the problem of computing a local minimum of the potential function. Fabrikant et al. [14] proved that the problem is PLS-complete (informally, as hard as it could be given that there is an associated potential function). This negative result applies to symmetric congestion games as well as to non-symmetric network congestion games. Ackermann et al. [1] studied the impact of combinatorial structure of congestion games to complexity and extended such negative results to games with linear latency functions. One consequence of PLS-completeness results is that almost all the states of the game are such that any sequence of players' improvement moves that originates from these states must be exponentially long (in terms of the number of players) in order to reach a pure Nash equilibrium. Efficient algorithms are known only for special cases. For example, in symmetric network congestion games, Fabrikant et al. [14] show that the Rosenthal's potential function can be (globally) minimized efficiently by a flow computation.

The above negative results have led to the study of the complexity of approximate Nash equilibria. A ρ-approximate pure Nash equilibrium is a state, from which no player has an incentive to deviate so that she decreases her cost by a factor larger than ρ. Skopalik and Vöcking [22] show that, in general, the problem is still PLS-complete for any polynomially computable ρ. Efficient algorithms are known only for special cases. For symmetric congestion games, Chien and Sinclair [8] prove that the $(1+\epsilon)$-improvement dynamics converges to a $(1+\epsilon)$-approximate Nash equilibrium after a polynomial number of steps; this result holds under additional mild assumptions on the latency functions (a "bounded jump" property) and the participation of the players in the dynamics. Skopalik and Vöcking [22] prove that this approach cannot be generalized. They present non-symmetric congestion games with latency functions satisfying the bounded-jump property, so that every sequence of approximate improvement moves from a given initial state to an approximate equilibrium is exponentially long. Daskalakis and Papadimitriou [12] present algorithms for the broader class of anonymous games assuming that the number of players' strategies is constant; for congestion games, this assumption is very restrictive. Efficient algorithms for approximate equilibria have been recently obtained for other classes of games such as constraint satisfaction [4], [19], network formation [2], and facility location games [5].

In light of these negative results, several authors have considered other properties of the dynamics of congestion games. The papers [3], [15] consider the question of whether efficient states (in the sense that the total cost of the players, or social cost, is small compared to the optimum one) can be reached by best-response moves. Recall that such states are not necessary approximate Nash equilibria. Fanelli et al. [15] proved that congestion games with linear latency functions converge to states that approximate the optimal social cost within a constant factor after an almost linear (in the number of players) number of best response moves under mild assumptions on the participation of each player in the dynamics. Negative results in [3] indicate that these assumptions are necessary in order to obtain convergence in subexponential time. However, Awerbuch et al. [3] show that using almost unrestricted sequences of $(1+\epsilon)$-improvement best-response moves in congestion games with polynomial latency functions, the players rapidly converge to efficient states. Similar approaches have been followed in the context of other games as well, such as multicast [6], [7], cut [11], and valid-utility games [17].

A notion that is historically related to congestion games (but rather loosely connected to our work) is that of the price of anarchy, introduced by Koutsoupias and Papadimitriou [16]. The price of anarchy captures the impact of selfishness on efficiency and is defined as the worst-case ratio of the social cost in any pure Nash equilibrium and the social optimum (see [21] and the references therein for tight bounds on congestion games). Christodoulou et al. [10] extended the notion of the price of anarchy to approximate equilibria and provided tight bounds for congestion games with polynomial latency functions.

1.2. Our contribution

We present the first polynomial-time algorithm that computes $O(1)$-approximate pure Nash equilibria in *non-symmetric* congestion games with polynomial latency functions of constant maximum degree. In particular, our algorithm computes $(2+\epsilon)$-approximate pure Nash equilibria in congestion games with linear latency functions, and $d^{O(d)}$ approximate equilibria for polynomial latency functions of maximum degree d. The algorithm is surprisingly simple. Essentially, starting from a specific initial state, it computes a sequence of best-response player moves of length that is bounded by a polynomial in the number of players and $1/\epsilon$. To the best of our knowledge, the existence of such short sequences was not known before and is interesting in itself. The sequence consists of phases so that the players that participate in each phase experience costs that are polynomially related. This is crucial in order to obtain convergence in polynomial time. Another interesting part of our algorithm is that, within each phase, it coordinates the best response moves according to two different (but simple) criteria; this is the main tool that guarantees that the effect of a phase to previous ones is negligible and, eventually, an approximate equilibrium is reached. The parameters used by the algorithm and its approximation guarantee have a nice relation to properties of Rosenthal's potential function. Our bounds are marginally higher than the worst-case ratio of the potential value at an almost exact pure Nash equilibrium over the globally optimum potential value.

We remark that, following the classical definition of polynomial latency functions in the literature, we assume that they have non-negative coefficients. We show that this is a necessary limitation. In particular, by significantly extending the reduction of [22], we prove that the problem of computing a ρ-approximate equilibrium in congestion games with linear latency functions with negative offsets is PLS-complete. This negative statement also applies to games with polynomial latency functions with non-negative coefficients and maximum degree that is polynomial in the number of players.

1.3. Roadmap

We begin with definitions and preliminary results and observations in Section 2. The description of the algorithm then appears in Section 3. The analysis of the algorithm is presented in Section 4. We conclude with a discussion that includes the statement of our PLS-completeness result and open problems in Section 5.

2. Definitions and preliminaries

A *congestion game* $\mathcal{G}$ is represented by the tuple $(N, E, (\Sigma_u)_{u\in N}, (f_e)_{e\in E})$. There is a set of n *players* $N = \{1, 2, ..., n\}$ and a set of *resources* E. Each player u has a set of available *strategies* Σ_u; each strategy s_u in Σ_u consists of a non-empty set of resources, i.e., $s_u \subseteq 2^E$. A snapshot of strategies, with one strategy per player, is called a *state* and is represented by a vector of players' strategies, e.g., $S = (s_1, s_2, ..., s_n)$. Each resource $e \in E$ has a *latency function* $f_e : \mathbb{N} \mapsto \mathbb{R}$ which denotes the latency incurred to the players using resource e; this latency depends on the number of players whose strategies include the particular resource. For a state S, let us define $n_e(S)$ to be the number of players that use resource e in S, i.e., $n_e(S) = |\{u \in N : e \in s_u\}|$. Then, the latency incurred by resource e to the players that use it is $f_e(n_e(S))$. The *cost* of a player u at a state S is the total latency she experiences at the resources in her strategy s_u, i.e., $c_u(S) = \sum_{e\in s_u} f_e(n_e(S))$. We mainly consider congestion games in which the resources have polynomial latency functions with non-negative coefficients. More precisely, the latency function of resource e is $f_e(x) = \sum_{k=0}^{d} a_{e,k}x^k$ with $a_{e,k} \geq 0$. The special case of linear latency functions (i.e., $d = 1$) is of particular interest. Observe that for polynomials with non-negative coefficients and maximum degree d, we have $f_e(x + 1) \leq 2^d f_e(x)$ and $f_e(x) \leq n^d f_e(1)$ for every positive integer x.

Players act selfishly; each of them aims to select a strategy that minimizes her cost, given the strategies of the other players. Given a state $S = (s_1, s_2, ..., s_n)$ and a strategy s'_u for player u, we denote by (S_{-u}, s'_u) the state obtained from S when player u *deviates* to strategy s'_u. For a strategy S, an *improvement move* (or, simply, a *move*) for player u is the deviation to any strategy s'_u that (strictly) decreases her cost, i.e., $c_u(S_{-u}, s'_u) < c_u(S)$. For $q \geq 1$, such a move is called a *q-move* if it satisfies $c_u(S_{-u}, s'_u) < \frac{c_u(S)}{q}$. A *best-response move* is a move that minimizes the cost of the player (of course, given the strategies of the other players). So, from state S, a move of player u to strategy s'_u is a best-response move (and is denoted by $\mathcal{BR}_u(S)$) when $c_u(S_{-u}, s'_u) = \min_{s\in\Sigma_u} c_u(S_{-u}, s)$. With some abuse in notation, we use $\mathcal{BR}_u(\mathbf{0})$ to denote the best-response of player u assuming that no other player participates in the game.

A state S is called a *pure Nash equilibrium* (or, simply, an *equilibrium*) when $c_u(S) \leq c_u(S_{-u}, s'_u)$ for every player $u \in N$ and every strategy $s'_u \in \Sigma_u$. In this case, we say that no player has (any incentive to make) a move. Similarly, a state is called a *q-approximate pure Nash equilibrium* (henceforth called, simply, a *q-approximate equilibrium*) when no player has a q-move.

Congestion games are potential games. They admit a *potential function* $\Phi : \prod_u \Sigma_u \mapsto \mathbb{R}$, defined over all states of the game, with the following property: for any two state S and (S_{-u}, s'_u) that differ only in the strategy of player u, it holds that $\Phi(S_{-u}, s'_u) - \Phi(S) = c_u(S_{-u}, s'_u) - c_u(S)$. Clearly, local minima of the potential function corresponds to states that are pure Nash equilibria. The function $\Phi(S) = \sum_{e\in E}\sum_{j=1}^{n_e(S)} f_e(j)$ (first used by Rosenthal [20]) is such a potential function. A nice property of this particular potential function is that the potential value at a state lies between the sum of latencies incurred by the resources and the total cost of the players.

Claim 2.1. *For any state S of a congestion game with a set of players N, a set of resource E, and latency functions $(f_e)_{e\in E}$, it holds that*

$$\sum_{e\in E} f_e(n_e(S)) \leq \Phi(S) \leq \sum_{u\in N} c_u(S).$$

Proof: The first inequality follows easily by the definition of function Φ. The second one can be obtained by the following derivation:

$$\begin{aligned} \Phi(S) &= \sum_{e\in E}\sum_{j=1}^{n_e(S)} f_e(j) \\ &\leq \sum_{e\in E} n_e(S)\cdot f_e\Big(n_e(S)\Big) \\ &= \sum_{u\in N}\sum_{e\in s_u} \cdot f_e\Big(n_e(S)\Big) = \sum_{u\in N} c_u(S). \end{aligned}$$

□

In the rest of the paper, the term potential function is used specifically for Rosenthal's potential function.

We now present a simple observation which will be used extensively in the analysis of our algorithm. Consider a sequence of moves in which only players from a subset F of N participate while players in $N \setminus F$ are frozen to their strategies throughout the whole sequence. We will think of this sequence as a sequence of moves in a *subgame* played among the players of F on the resources of E. In this subgame, each player in F has the same set of strategies as in the original game; players of $N \setminus F$ do not participate in the subgame, although they contribute to the latency of the resources at which they have been frozen. Thus, the *modified* latency function of resource e is then $f_e^F(x) = f_e(x + t_e)$, where t_e stands for the number of players of $N \setminus F$ on resource e. Then, it is not hard to see that the subgame is a congestion game as well. Clearly, if f_e is a linear (respectively, polynomial of maximum degree d) function with non-negative coefficients, so is f_e^F and the bound established in Lemma 2.3 (respectively, Lemma 2.5, see below) also holds for the subgame. From the perspective of a player in F, nothing changes. At any state S, such a player experiences the same cost in both games and therefore has the same incentive to move, regardless whether we view S as a state of the original game or the subgame. However, one should be careful with the definition of the potential for the subgame (denoted by Φ_F) and use the modified latency functions f_e^F instead of f_e. Throughout the paper, for a subset of players $F \subseteq N$, we use the notation $n_e^F(S)$ to denote the number of players in F that use resource e at state S.

Claim 2.2. *Let S be a state of the congestion game with a set of players N and let $F \subseteq N$. Then, $\Phi(S) \leq \Phi_F(S) + \Phi_{N\setminus F}(S)$ and $\Phi(S) \geq \Phi_F(S)$.*

Proof: We use the definition of the potential function for the original game and the subgames, the definitions of the modified latency functions $f_e^F(x) = f_e(x + n_e^{N\setminus F}(S))$ and $f_e^{N\setminus F}(x) = f_e(x + n_e^F(S))$, and the equality $n_e(S) = n_e^F(S) + n_e^{N\setminus F}(S)$ to obtain

$$
\begin{aligned}
\Phi(S) &= \sum_{e\in E} \sum_{j=1}^{n_e(S)} f_e(j) \\
&= \sum_{e\in E} \sum_{j=1}^{n_e^F(S)} f_e(j) + \sum_{e\in E} \sum_{j=n_e^F(S)+1}^{n_e(S)} f_e(j) \\
&\leq \sum_{e\in E} \sum_{j=1}^{n_e^F(S)} f_e(j + n_e^{N\setminus F}(S)) \\
&\quad + \sum_{e\in E} \sum_{j=n_e^F(S)+1}^{n_e(S)} f_e(j) \\
&= \sum_{e\in E} \sum_{j=1}^{n_e^F(S)} f_e^F(j) + \sum_{e\in E} \sum_{j=1}^{n_e^{N\setminus F}(S)} f_e(j + n_e^F(S)) \\
&= \sum_{e\in E} \sum_{j=1}^{n_e^F(S)} f_e^F(j) + \sum_{e\in E} \sum_{j=1}^{n_e^{N\setminus F}(S)} f_e^{N\setminus F}(j) \\
&= \Phi_F(S) + \Phi_{N\setminus F}(S),
\end{aligned}
$$

as desired for the first part of the claim. For the second part, we have

$$
\begin{aligned}
\Phi(S) &= \sum_{e\in E} \sum_{j=1}^{n_e(S)} f_e(j) \\
&\geq \sum_{e\in E} \sum_{j=n_e^{N\setminus F}(S)+1}^{n_e(S)} f_e(j) \\
&= \sum_{e\in E} \sum_{j=1}^{n_e^F(S)} f_e(j + n_e^{N\setminus F}(S)) \\
&= \sum_{e\in E} \sum_{j=1}^{n_e^F(S)} f_e^F(j) \\
&= \Phi_F(S).
\end{aligned}
$$

□

The approximation guarantee of our algorithm for congestion games of a particular class (e.g., with linear latency functions) is strongly related to the worst-case ratio (among *all* congestion games in the class) between the potential of an approximate equilibrium (the factor of approximation may be picked to be close to 1) and the minimum potential value. Below, we present upper bounds on this quantity; these upper bounds are used as parameters by our algorithm. The next lemma deals with the case of linear latency functions.

Lemma 2.3. *Consider a congestion game with linear latency functions. Let $q \in [1, 2)$ and let S be a q-approximate equilibrium. Then, $\Phi(S) \leq \frac{2q}{2-q}\Phi(S^*)$, where S^* is a state of the game with minimum potential.*

Proof: In the proof, we will need the following technical claim.

Claim 2.4. *For every non-negative integers x, y, it holds true $xy \leq \frac{1}{2}x^2 - \frac{1}{2}x + y^2$.*

Proof: For $x = 1$, the claim clearly holds. Otherwise, observe that $\frac{1}{2}x^2 - \frac{1}{2}x \geq \frac{1}{4}x^2$. Then $0 \leq (\frac{x}{2} - y)^2 = \frac{1}{4}x^2 + y^2 - xy \leq \frac{1}{2}x^2 - \frac{1}{2}x + y^2 - xy$ and claim follows. □

For each player u we denote by s_u and s_u^* the strategies she uses at states S and S^*, respectively. Using the q-approximate equilibrium condition, that is $c_u(S) \leq q \cdot$

$c_u(S_{-u}, s_u^*)$, we obtain

$$\begin{aligned}&\sum_{e\in s_u}(a_{e,1}\cdot n_e(S)+a_{e,0})\\ \le\ & q\sum_{e\in s_u^*}(a_{e,1}\cdot n_e(S_{-u},s_u^*)+a_{e,0})\end{aligned}$$

for each player $u \in N$. Summing over all players, we get that their total cost is

$$\begin{aligned}&\sum_{u\in N}\sum_{e\in s_u}(a_{e,1}\cdot n_e(S)+a_{e,0})\\ \le\ & q\sum_{u\in N}\sum_{e\in s_u^*}(a_{e,1}\cdot n_e(S_{-u},s_u^*)+a_{e,0}) \qquad (1)\\ \le\ & q\sum_{u\in N}\sum_{e\in s_u^*}(a_{e,1}\cdot (n_e(S)+1)+a_{e,0})\\ =\ & q\sum_{e\in E}(a_{e,1}\cdot n_e(S^*)(n_e(S)+1)\\ & +a_{e,0}\cdot n_e(S^*)) \qquad (2)\end{aligned}$$

In the following, we use the definitions of the potential and the latency functions, the fact that $q \ge 1$, inequality (2) $\left(n_e(S)\cdot n_e(S^*) \le \frac{1}{2}n_e(S)^2 - \frac{1}{2}n_e(S) + n_e(S^*)^2\right)$, and Claim 2.4 to obtain

$$\begin{aligned}&\Phi(S)\\ =\ & \sum_{e\in E}\sum_{j=1}^{n_e(S)} f_e(j)\\ =\ & \sum_{e\in E}\left(\frac{1}{2}a_{e,1}\cdot n_e(S)^2+\frac{1}{2}a_{e,1}\cdot n_e(S)+a_{e,0}\cdot n_e(S)\right)\\ =\ & \sum_{e\in E}\left(\frac{1}{2}a_{e,1}\cdot n_e(S)^2+\frac{1}{2}a_{e,0}\cdot n_e(S)\right)\\ & +\sum_{e\in E}\frac{1}{2}a_{e,1}\cdot n_e(S)+\sum_{e\in E}\frac{1}{2}a_{e,0}\cdot n_e(S)\\ \le\ & \frac{1}{2}\sum_{u\in N}\sum_{e\in s_u}\Big(a_{e,1}\cdot n_e(S)+a_{e,0}\Big)+\frac{q}{2}\sum_{e\in E}a_{e,1}n_e(S)\\ & +\frac{q}{2}\sum_{e\in E}a_{e,0}n_e(S)\\ \le\ & \frac{q}{2}\sum_{e\in E}\Big(a_{e,1}\cdot n_e(S^*)(n_e(S)+1)+a_{e,0}\cdot n_e(S^*)\Big)\\ & +\frac{q}{2}\sum_{e\in E}a_{e,1}\cdot n_e(S)+\frac{q}{2}\sum_{e\in E}a_{e,0}\cdot n_e(S)\\ \le\ & \frac{q}{2}\sum_{e\in E}a_{e,1}\left(\frac{1}{2}n_e(S)^2+\frac{1}{2}n_e(S)+n_e(S^*)^2+n_e(S^*)\right)\\ & +\frac{q}{2}\sum_{e\in E}a_{e,0}\cdot n_e(S^*)+\frac{q}{2}\sum_{e\in E}a_{e,0}\cdot n_e(S)\\ \le\ & \frac{q}{2}\sum_{e\in E}\left(\frac{1}{2}a_{e,1}\cdot n_e(S)^2+\frac{1}{2}a_{e,1}\cdot n_e(S)+a_{e,0}\cdot n_e(S)\right)\\ & +q\sum_{e\in E}\left(\frac{1}{2}a_{e,1}\cdot n_e(S^*)^2+\frac{1}{2}a_{e,1}\cdot n_e(S^*)\right.\\ & \left.+a_{e,0}\cdot n_e(S^*)\right)\\ =\ & \frac{q}{2}\Phi(S)+q\Phi(S^*),\end{aligned}$$

and, equivalently, $\Phi(S) \le \frac{2q}{2-q}\Phi(S^*)$. □

Our next (rather rough) bound applies to polynomial latency functions of maximum degree d. It is obtained by observing that the desired ratio is at most $(d+1)$ times the known upper bound of $d^{O(d)}$ for the price of anarchy of 2-approximate equilibria [10].

Lemma 2.5. *Consider a congestion game with polynomial latency functions of maximum degree d, where $d \ge 2$. Let $q \in [1, 2]$ and let S be a q-approximate equilibrium. Then, $\Phi(S)/\Phi(S^*) \in d^{O(d)}$, where S^* is a state of the game with the minimum potential.*

Proof: Observe that $\Phi(S) \le \sum_{u\in N} c_u(S)$ (see Claim 2.1). We will also show that $\Phi(S^*) \ge \frac{1}{d+1}\sum_{u\in N} c_u(S^*)$. The desired bound then follows by the fact that the price of anarchy of 2-approximate equilibria is at most $d^{O(d)}$. Notice that the price of anarchy is at least $\sum_{u\in N} c_u(S)/\sum_{u\in N} c_u(S^*)$.

We will use the property $\int_0^y f(x)dx \le \sum_{j=1}^{y} f(j)$, that holds for every non-decreasing function $f : [0, y] \to R$ and integer $y \ge 1$. We prove the desired inequality as follows.

$$\begin{aligned}\Phi(S^*) &= \sum_{e\in E}\sum_{j=1}^{n_e(S^*)} f_e(j)\\ &\ge \sum_{e\in E}\int_0^{n_e(S^*)} f_e(x)dx\\ &= \sum_{e\in E}\int_0^{n_e(S^*)}\sum_{k=0}^{d} a_{e,k}\cdot x^k dx\\ &= \sum_{e\in E}\sum_{k=0}^{d}\frac{a_{e,k}}{k+1}n_e(S^*)^{k+1}\\ &\ge \frac{1}{d+1}\sum_{e\in E}\sum_{k=0}^{d}a_{e,k}\cdot n_e(S^*)^{k+1}\\ &= \frac{1}{d+1}\sum_{e\in E}n_e(S^*)f_e(S^*)\\ &= \frac{1}{d+1}\sum_{u\in N}c_u(S^*).\end{aligned}$$

□

3. The algorithm

In this section we describe our algorithm. It takes as input a congestion game $\mathcal{G}$ with n players and polynomial latency

input : A congestion game $\mathcal{G} = (N, E, (\Sigma_i)_{i\in N}, (f_e)_{e\in E})$ with n players and polynomial latency functions of maximum degree d

output: A state of $\mathcal{G}$

1 Set $q = 1 + n^{-\psi}$ and $p = \left(\frac{1}{\theta_d(q)} - \frac{2}{n^{\psi}}\right)^{-1}$;

2 **foreach** $u \in N$ **do** set $\ell_u = c_u\left(\mathcal{BR}_u(\mathbf{0})\right)$;

3 Set $\ell_{\min} = \min_{u\in N} \ell_u$, $\ell_{\max} = \max_{u\in N} \ell_u$, and set $m = 1 + \lceil \log_{2^{d+1}n^{2\psi+d+1}} (\ell_{\max}/\ell_{\min}) \rceil$;

4 (Implicitly) partition players into *blocks* $B_1, B_2, \ldots, B_m$, such that $u \in B_i \Leftrightarrow \ell_u \in \left(\ell_{\max}\left(2^{d+1}n^{2\psi+d+1}\right)^{-i}, \ell_{\max}\left(2^{d+1}n^{2\psi+d+1}\right)^{-i+1}\right]$;

5 **foreach** $u \in N$ **do** set u to play the strategy $s_u \leftarrow \mathcal{BR}_u(\mathbf{0})$;

6 **for** *phase* $i \leftarrow 1$ **to** $m - 1$ *such that* $B_i \neq \emptyset$ **do**

7   **while** *there exists a player* u *that either belongs to* B_i *and has a* p*-move or belongs to* B_{i+1} *and has a* q*-move* **do**

8   u deviates to the best-response strategy $s_u \leftarrow \mathcal{BR}_u(s_1, \ldots, s_n)$.

9   **end**

10 **end**

Algorithm 1. Computing approximate equilibria in congestion games.

functions of maximum degree d and produces a state of $\mathcal{G}$. The algorithm uses a constant parameter $\psi > 0$ and two more parameters q and p. Denote by $\theta_d(q)$ the upper bound on the worst-case ratio (among all possible congestion games with polynomial latency functions of degree d) between the potential of any q-approximate equilibrium and the minimum potential value that are provided by Lemmas 2.3 and 2.5, i.e., $\theta_1(q) = \frac{2q}{2-q}$ and $\theta_d(q) = d^{O(d)}$ for $d \geq 2$. We set the parameter q to be slightly larger than 1 (in particular, $q = 1+n^{-\psi}$) and parameter p to be slightly larger than $\theta_d(q)$ (in particular, $p = \left(\frac{1}{\theta_d(q)} - \frac{2}{n^{\psi}}\right)^{-1}$).

The algorithm considers the *optimistic cost* ℓ_u of each player u, given by $\ell_u = \min_{s_u\in\Sigma_u} \sum_{e\in s_u} f_e(1)$; this is the minimum cost that u could experience assuming that no other player participates in the game. Let $\ell_{\max}$ denote the maximum optimistic cost among all players. The algorithm partitions the players into *blocks* $B_1, B_2, \ldots, B_m$; block B_i contains player u if and only if $\ell_u \in (b_{i+1}, b_i]$, where $b_i = \ell_{\max}\left(2^{d+1}n^{2\psi+d+1}\right)^{-i+1}$. It initializes each player u to choose strategy $\mathcal{BR}_u(\mathbf{0})$. Then, the algorithm coordinates best-response moves by the players as follows. By considering i in the increasing order (from 1 to $m-1$), it executes phase i provided that block B_i is non-empty. When at phase i, the algorithm lets players in B_i make best-response p-moves and players in B_{i+1} make best-response q-moves while this is possible. The algorithm is depicted in the following table.

We remark that step 2 partitions the players into at most n non-empty blocks. Then, the for-loop at lines 6-10 enumerates only phases i such that B_i is non-empty, i.e., it considers at most n phases.

We conclude this section with two remarks that will be treated formally in the next section. First, the selection of the boundaries of each block to be polynomially-related is crucial in order to bound the number of steps. Second, but more importantly, we notice that each player in block B_i does not move after phase i. At the end of this phase, the algorithm guarantees that none of these players has a p-move to make. The most challenging part of the analysis will be to show that the players do not have any $p(1 + 4n^{-\psi})$-move to make after any subsequent phase. In this respect, the definition of the phases, the selection of parameter p and its relation to $\theta_d(q)$ play the crucial role.

4. Analysis of the algorithm

This section is devoted to proving our main result.

Theorem 4.1. *For every constant $\psi > 0$, the algorithm computes a ρ_d-approximate equilibrium for every congestion game with polynomial latency functions of constant maximum degree d and n players, where $\rho_1 = 2 + O(n^{-\psi})$ and $\rho_d \in d^{O(d)}$. Moreover, the number of player moves is at most polynomial in n.*

Proof: We denote by S^0 the state computed at step 5 of the algorithm (player u plays strategy $\mathcal{BR}_u(\mathbf{0})$) and by S^i the state after the execution of phase i for $i \geq 1$. Within each phase i, we denote by R_i the set of players that make at least one move during the phase. Recall that the players of block B_i are those with optimistic cost $\ell_u \in (b_{i+1}, b_i]$ and that $b_i = 2^{d+1}n^{2\psi+d+1}b_{i+1}$, for $i = 1, ..., m$.

The proof of the theorem follows by a series of lemmas. The most crucial one is Lemma 4.3 where we show that the potential $\Phi_{R_i}(S^{i-1})$ of the subgame among the players in R_i at the beginning of phase $i \geq 2$ is significantly smaller than b_i. In general, players that move during phase i experience cost that is polynomially related to b_i and each of them decreases her cost (and, consequently, the potential) by a quantity that is also polynomially related to b_i. This argument is used in Lemma 4.4 (together with Lemma 4.3) in order to show that the number of steps of the algorithm is polynomial in n. More importantly, Lemma 4.3 is used in the proof of Lemma 4.5 in order to show that players in block B_i are not affected significantly after phase i (notice that players in B_i do not move after phase i). Using this lemma, we conclude in Lemma 4.6 that the players are in a $p(1+4n^{-\psi})$-approximate equilibrium after the execution of the algorithm. The statement of the theorem then follows by taking into account the parameters of the algorithm.

Let us warm up with the following lemma (to be used in the proof of Lemma 4.3) that relates the potential $\Phi_{R_i}(S^i)$ with the latency the players in R_i experience when they make their last move within phase i.

Lemma 4.2. *Let $c(u)$ denote the cost of player $u \in R_i$ just after making her last move within phase i. Then,*

$$\Phi_{R_i}(S^i) \leq \sum_{u\in R_i} c(u).$$

Proof: We denote by s_u the strategy of player u at state S^i. We rank the players that use resource e in S^i according to the timing of their last moves (using consecutive integers $1, 2, ...$). We denote by $\text{rank}_e(u)$ the number of players in R_i with the smaller ranking than u on resource e. Then, we get $c(u) \geq \sum_{e\in s_u} f_e^{R_i}(\text{rank}_e(u))$, since any resource e in s_u is occupied by at least $\text{rank}_e(u)$ players from R_i at state S^i: u and the players with ranks $1, 2, ..., \text{rank}_e(u) - 1$ that made their last move before u. Hence, by the definition of the potential function (expressed using the modified latency functions for the subgame among the players of R_i), we have

$$\begin{aligned}
\Phi_{R_i}(S^i) &= \sum_{e\in E} \sum_{j=1}^{n_e^{R_i}(S^i)} f_e^{R_i}(j) \\
&= \sum_{e\in E} \sum_{u\in R_i : e\in s_u} f_e^{R_i}(\text{rank}_e(u)) \\
&= \sum_{u\in R_i} \sum_{e\in s_u} f_e^{R_i}(\text{rank}_e(u)) \\
&\leq \sum_{u\in R_i} c(u),
\end{aligned}$$

and the lemma follows. □

We now present the key lemma of our proof.

Lemma 4.3. *For every phase $i \geq 2$, it holds that $\Phi_{R_i}(S^{i-1}) \leq \frac{b_i}{2^d n^\psi}$.*

Proof: Assume the contrary, that $\Phi_{R_i}(S^{i-1}) > \frac{b_i}{2^d n^\psi}$. We will show that state S^{i-1} would not be a q-approximate equilibrium for the players in $R_i \cap B_i$, which contradicts the definition of phase $i-1$ of the algorithm.

First observe that a player u in B_{i+1} is assigned to the strategy $\mathcal{BR}_u(\mathbf{0})$ in the beginning of the algorithm and does not move during the first $i-1$ phases. Hence, by the definition of the latency functions, she does not experience a cost more than $n^d b_{i+1}$ at state S^{i-1}. Hence, the potential $\Phi_{R_i\cap B_{i+1}}(S^{i-1})$, which is upper-bounded by the total cost of players in $R_i \cap B_{i+1}$, satisfies

$$\Phi_{R_i\cap B_{i+1}}(S^{i-1}) \leq n^{d+1} b_{i+1}. \tag{3}$$

We now use the fact $\Phi_{R_i}(S^{i-1}) \leq \Phi_{R_i\cap B_i}(S^{i-1}) + \Phi_{R_i\cap B_{i+1}}(S^{i-1})$ (see Claim 2.2), inequality (3), and the assumption $\Phi_{R_i}(S^{i-1}) > \frac{b_i}{2^d n^\psi}$ to obtain

$$\begin{aligned}
\Phi_{R_i\cap B_i}(S^{i-1}) &\geq \Phi_{R_i}(S^{i-1}) - \Phi_{R_i\cap B_{i+1}}(S^{i-1}) \\
&> \frac{b_i}{2^d n^\psi} - n^{d+1} b_{i+1} \\
&= \left(\frac{2^{d+1} n^{2\psi+d+1}}{2^d n^\psi} - n^{d+1}\right) b_{i+1} \\
&\geq n^{\psi+d+1} b_{i+1}. \qquad (4)
\end{aligned}$$

Further, we consider the dynamics of the subgame among the players in R_i at phase i. For each player u in R_i, we denote by $c(u)$ the cost player u experiences just after she makes her last move in phase i. Observe that every player u in $B_i \cap R_i$ decreases the potential of the subgame among the players of R_i by at least $(p-1)c(u)$ when she performs her last p-move. Hence,

$$\begin{aligned}
& (p-1) \sum_{u\in R_i\cap B_i} c(u) \\
\leq\ & \Phi_{R_i}(S^{i-1}) - \Phi_{R_i}(S^i) \qquad (5) \\
\leq\ & \Phi_{R_i\cap B_i}(S^{i-1}) + \Phi_{R_i\cap B_{i+1}}(S^{i-1}) - \Phi_{R_i}(S^i) \\
\leq\ & \Phi_{R_i\cap B_i}(S^{i-1}) + n^{d+1} b_{i+1} - \Phi_{R_i}(S^i) \\
<\ & \left(1 + \frac{1}{n^\psi}\right) \Phi_{R_i\cap B_i}(S^{i-1}) - \Phi_{R_i}(S^i). \qquad (6)
\end{aligned}$$

The last three inequalities follow by Claim 2.2 and inequalities (3) and (4), respectively.

Furthermore, since each player u in $R_i \cap B_{i+1}$ plays a best-response during phase i, her cost after her last move will be at most the cost she would experience by deviating to strategy $\mathcal{BR}_u(\mathbf{0})$, which is at most $n^d b_{i+1}$. Then, the total cost of the players of $R_i \cap B_{i+1}$ is at most $n^{d+1} b_{i+1}$. Now, using Lemma 4.2, the last observation, inequalities (6) and (4), we obtain

$$\begin{aligned}
\Phi_{R_i}(S^i) &\leq \sum_{u\in R_i} c(u) \\
&= \sum_{u\in R_i\cap B_{i+1}} c(u) + \sum_{u\in R_i\cap B_i} c(u) \\
&< n^{d+1} b_{i+1} + \frac{1}{p-1}\left(1 + \frac{1}{n^\psi}\right) \Phi_{R_i\cap B_i}(S^{i-1}) \\
&\quad - \frac{1}{p-1} \Phi_{R_i}(S^i) \\
&\leq \frac{1}{p-1}\left(1 + \frac{p}{n^\psi}\right) \Phi_{R_i\cap B_i}(S^{i-1})
\end{aligned}$$

$$-\frac{1}{p-1}\Phi_{R_i}(S^i)$$

which implies that

$$\Phi_{R_i}(S^i) \quad < \quad \left(\frac{1}{p}+\frac{1}{n^{\psi}}\right)\Phi_{R_i\cap B_i}(S^{i-1}). \tag{7}$$

Now, let S^* be the state in which the players in $R_i \cap B_i$ play their strategies in S^i and the players in $R_i \cap B_{i+1}$ (as well as every other player) play their strategies in S^{i-1}. Consider the deviation of each player u in $R_i \cap B_{i+1}$ from her strategy in S^i to her strategy $\mathcal{BR}_u(\mathbf{0})$ in S^*. Recall that the cost each player u in $R_i \cap B_{i+1}$ experiences when playing strategy $\mathcal{BR}_u(\mathbf{0})$ is at most $n^d b_{i+1}$ which means that the increase her deviation incurs to the potential of the subgame among the players in R_i is at most $n^d b_{i+1}$. Hence,

$$\Phi_{R_i}(S^*) \quad \leq \quad \Phi_{R_i}(S^i)+n^{d+1}b_{i+1}. \tag{8}$$

Now, using the fact that $\Phi_{R_i\cap B_i}(S^*) \leq \Phi_{R_i}(S^*)$, together with inequalities (8), (7), and (4), we have

$$\begin{aligned}
\Phi_{R_i\cap B_i}(S^*) &\leq \Phi_{R_i}(S^*)\\
&\leq \Phi_{R_i}(S^i)+n^{d+1}b_{i+1}\\
&< \left(\frac{1}{p}+\frac{2}{n^{\psi}}\right)\Phi_{R_i\cap B_i}(S^{i-1})\\
&= \frac{1}{\theta_d(q)}\Phi_{R_i\cap B_i}(S^{i-1}).
\end{aligned}$$

The last inequality implies that the global minimum of the potential value of the subgame among the players of $R_i \cap B_i$ (when all other players are frozen to their strategies in S^{i-1}) is strictly smaller than $\frac{1}{\theta_d(q)}\Phi_{R_i\cap B_i}(S^{i-1})$. Due to the definition of $\theta_d(q)$ and Lemmas 2.3 and 2.4, this contradicts the fact that S^{i-1} is a q-approximate equilibrium for the players in $R_i \cap B_i$. □

We are ready to bound the number of best-response moves. As a matter of fact, our upper bound is dominated by the number of best-response moves in the very first phase of the algorithm. We remark that a weaker result could be obtained without resorting to Lemma 4.3.

Lemma 4.4. *The algorithm terminates after at most $O\left(n^{5\psi+3d+3}\right)$ best-response moves.*

Proof: We will upper-bound the total number of moves during the execution of the algorithm. After the first n best-response moves in line 5, the number of phases executed by the algorithm is at most n. At the beginning of the first phase, the latency of any player in R_1 is at most $n^d b_1$ (due to the definition of block B_1 and of the latency functions). Hence, $\Phi_{R_1}(S^0) \leq \sum_{u\in R_1} c_u(S^0) \leq n^{d+1}b_1$. The minimum latency experienced by any player in R_1 is at least b_3, so each move in this step decreases the potential Φ_{R_1} by at least $(q-1)b_3$. So the total number of moves is at most $\frac{n^{d+1}b_1}{(q-1)b_3} = 2^{2d+2}n^{5\psi+3d+3}$.

At the beginning of any other phase $i \geq 2$, we have that $\Phi_{R_i}(S^{i-1}) \leq \frac{b_i}{2^d n^{\psi}}$ (by Lemma 4.3). The minimum latency experienced by any player in R_i is at least b_{i+2}, so each move in this step decreases the potential Φ_{R_i} by at least $(q-1)b_{i+2}$. So the total number of moves is at most $\frac{b_i}{2^d n^{\psi}(q-1)b_3} = 2^{d+2}n^{4\psi+2d+2}$.

In total, we have $O\left(n^{5\psi+3d+3}\right)$ best-response moves. □

The proof of the following lemma strongly relies on Lemma 4.3. Intuitively, Lemma 4.3 implies that the cost experienced by any player of R_i while moving during phase i is considerably lower than the cost of players in blocks $B_1, \ldots, B_{i-1}$ (who are not supposed to move anymore). The latter means that, for every player u in $B_1, \ldots, B_{i-1}$, after phase i, neither the cost of u may increase considerably, nor the cost that u could experience by a possible deviation may decrease considerably.

Lemma 4.5. *Let u be a player in the block B_t, where $t \leq m-2$. Let s'_u be a strategy different from the one assigned to u by the algorithm at the end of phase t. Then, for each phase $i \geq t$, it holds that*

$$c_u(S^i) \quad \leq \quad p\cdot c_u(S^i_{-u}, s'_u) + \frac{p+1}{n^{\psi}}\sum_{k=t+1}^{i} b_k.$$

Proof: We will prove the lemma using induction on i. For $i = t$, the claim follows by the definition of phase i of the algorithm. Assume that the claim is true for a phase i with $t \leq i \leq m-2$. In the following, we show that the claim is true for the phase $i+1$ as well.

First, we show that if

$$c_u(S^{i+1}) \quad \leq \quad c_u(S^i) + \frac{b_{i+1}}{n^{\psi}} \tag{9}$$

and

$$c_u(S^i_{-u}, s'_u) \quad \leq \quad c_u(S^{i+1}_{-u}, s'_u) + \frac{b_{i+1}}{n^{\psi}} \tag{10}$$

then the claim holds. By the hypothesis of induction, we have

$$c_u(S^i) \quad \leq \quad p\cdot c_u(S^i_{-u}, s'_u) + \frac{p+1}{n^{\psi}}\sum_{k=t+1}^{i} b_k.$$

Combining the above three inequalities, we obtain that

$$\begin{aligned}
c_u(S^{i+1}) &\leq c_u(S^i) + \frac{b_{i+1}}{n^{\psi}}\\
&\leq p\cdot c_u(S^i_{-u}, s'_u) + \frac{p+1}{n^{\psi}}\sum_{k=t+1}^{i} b_k + \frac{b_{i+1}}{n^{\psi}}\\
&\leq p\cdot c_u(S^{i+1}_{-u}, s'_u) + \frac{p+1}{n^{\psi}}\sum_{k=t+1}^{i+1} b_k,
\end{aligned}$$

as desired.

In order to complete the proof of the inductive step we are left to prove (9) and (10). We do so by proving that if one of these two inequalities does not hold, this would violate the statement of Lemma 4.3.

Assume that (9) does not hold, i.e., $c_u(S^{i+1}) > c_u(S^i) + \frac{b_{i+1}}{n^\psi}$ for some player u of block B_t, where $t \leq i$. We will show that the potential $\Phi_{R_{i+1}}(S^{i+1})$ at state S^{i+1} of the subgame among the players in R_{i+1} is larger than $\frac{b_{i+1}}{2^d n^\psi}$. Since the potential decreases during phase $i+1$, $\Phi_{R_{i+1}}(S^i)$ should also be larger than $\frac{b_{i+1}}{2^d n^\psi}$, contradicting Lemma 4.3. Indeed, since player u does not move during phase $i+1$, the increase in her cost from state S^i to state S^{i+1} implies the existence of a set of resources $C \subseteq s_u$ in her strategy with the following properties: each resource $e \in C$ is also used by at least one player of R_{i+1} in state S^{i+1} and, furthermore, $\sum_{e\in C} f_e(n_e(S^{i+1})) > \frac{b_{i+1}}{n^\psi}$. By Claim 2.1, we obtain that $\Phi_{R_{i+1}}(S^{i+1}) > \frac{b_{i+1}}{n^\psi}$.

Similarly, assume that (10) does not hold for a player u of block B_t and a strategy s'_u that is different from s_u, the strategy assigned to u in phase t, i.e., $c_u(S^i_{-u}, s'_u) > c_u(S^{i+1}_{-u}, s'_u) + \frac{b_{i+1}}{n^\psi}$. Recall that player u does not move during phase $i+1$. This implies that there exists a set of resources $C \subseteq s'_u$ with the following properties: each resource $e \in C$ is used by at least one player of R_{i+1} in state S^i and, furthermore, $\sum_{e\in C} f_e(n_e(S^i_{-u}, s'_u)) \geq \frac{b_{i+1}}{n^\psi}$. Hence, by Claim 2.1 and the definition of the latency functions, we have $\Phi_{R+1}(S^i) \geq \sum_{e\in C} f_e(n_e(S^i)) \geq \sum_{e\in C} \frac{1}{2^d} f_e(n_e(S^i_{-u}, s'_u)) > \frac{b_{i+1}}{2^d n^\psi}$. Again, this contradicts Lemma 4.3.

Hence, (9) and (10) hold and the proof of the inductive step is complete. □

The next lemma follows easily by Lemma 4.5, the definition of b_i's, and the definition of the last phase of the algorithm.

Lemma 4.6. *The state computed by the algorithm is a $p\left(1+\frac{4}{n^\psi}\right)$-approximate equilibrium.*

Proof: We have to show that in the state S^{m-1}, computed by the algorithm after the last phase, no player has an incentive to deviate to another strategy in order to decrease her cost by a factor of $p\left(1+\frac{4}{n^\psi}\right)$. The claim is certainly true for the players in the blocks B_{m-1} and B_m by the definition of the last phase of the algorithm. Let u be a player in block B_t with $t \leq m-2$ and let s'_u be any strategy different from the one assigned to u by the algorithm after phase t. We apply Lemma 4.5 to player u. By the definition of b_i's, we have $\sum_{k=t+1}^m b_k \leq 2b_{t+1}$. Also, $c_u(S^{m-1}_{-u}, s'_u) \geq b_{t+1}$, since u belongs to block B_t. Hence, Lemma 4.5 implies that

$$\begin{aligned} c_u(S^{m-1}) &\leq p \cdot c_u(S^{m-1}_{-u}, s'_u) + \frac{2(p+1)}{n^\psi} c_u(S^{m-1}_{-u}, s'_u) \\ &\leq p\left(1+\frac{4}{n^\psi}\right) c_u(S^{m-1}_{-u}, s'_u), \end{aligned}$$

as desired. The last inequality follows since $p \geq 1$. □

By the definition of the parameters q and p in our algorithm, we obtain that the state computed is a ρ_d-approximate equilibrium with

$$\rho_d \leq \left(\frac{1}{\theta_d(q)} - \frac{2}{n^\psi}\right)^{-1}\left(1+\frac{4}{n^\psi}\right),$$

where $\theta_1(q) = \frac{2q}{2-q}$, $\theta_d(q) \in d^{O(d)}$ and $q = 1 + n^{-\psi}$. By making simple calculations, we obtain that $\rho_1 \leq 2 + O(n^{-\psi})$ and $\rho_d \in d^{O(d)}$. This completes the proof of Theorem 4.1. □

5. Discussion and open problems

We remark that the number of best response moves computed by our algorithm depends neither on the number of the resources nor on the number of strategies per player. In fact, our algorithm delegates to the players the computation of their best-response move; the overall running time then depends also on the time required by the players to compute a best-response move from any state of the game and (pseudo-)state $\mathbf{0}$. Of course, the players are expected to be able to do this computation efficiently.

The guarantee of our algorithm depends strongly on the fact that the latency functions have non-negative coefficients. Is this a severe limitation? We answer this question negatively in the next theorem where we prove that the problem of computing approximate equilibria is PLS-complete for congestion games with linear latency functions that have negative offsets (but incurring non-negative latency to any player using the corresponding resource).

Theorem 5.1. *Finding an ρ-approximate equilibrium in a congestion game with linear laency functions with negative coefficients is PLS-complete, for every polynomial-time computable $\rho > 1$.*

The reduction yields a congestion game in which every resource is contained in strategies of at most two players. It can also be turned into a congestion game with polynomial latency functions that have degree polynomial in n. The proof is a rework of the reduction in [22] and has to be omitted due to space constraints.

Our work reveals several open problems. The most challenging one is whether the guarantee for approximate equilibria that can be computed efficiently can be improved. For example, can we compute $(1+\epsilon)$-approximate equilibria in congestion games with linear latency functions in polynomial time for every (polynomially small) $\epsilon > 0$? We believe that this is not the case and our algorithm is close to optimal in this sense. It would be very interesting to see how the best possible approximation guarantee relates to the worst-case ratio of the potential at an almost exact equilibrium over the minimum potential. Here, we point out that we have examples of congestion games for which the upper bound of 2, provided by Lemma 2.3, is tight when q approaches 1. Extending this question to polynomial latencies is interesting as well. Note that a nice consequence of our work is that, besides being approximate equilibria, the states computed have low price of anarchy as well (e.g., at least $7.33 + O(\epsilon)$ for linear latency functions according to the bounds in [10]). Providing improved guarantees for the social cost of approximate equilibria that can be computed efficiently or related trade-offs is another interesting line of research. Finally, we strongly believe that our techniques could be applicable to other potential games as well. Typical examples include constraint satisfaction games such as the cut and parity games studied in [4]; we plan to consider such games in future work.

References

[1] H. Ackermann, H. Röglin, and B. Vöcking. On the impact of combinatorial structure on congestion games. *Journal of the ACM*, 55(6), 2008.

[2] E. Anshelevich and B. Caskurlu. Exact and approximate equilibria for optimal group network formation. In *Proceedings of the 17th Annual Symposium on Algorithms (ESA)*, LNCS 5757, Springer, pages 239–250, 2009.

[3] B. Awerbuch, Y. Azar, A. Epstein, V. S. Mirrokni, and A. Skopalik. Fast convergence to nearly optimal solutions in potential games. In *Proceedings of the 9th ACM Conference on Electronic Commerce (EC)*, ACM, pages 264–273, 2008.

[4] A. Bhalgat, T. Chakraborty, and S. Khanna. Approximating pure Nash equilibrium in cut, party affiliation, and satisfiability games. In *Proceedings of the 11th ACM Conference on Electronic Commerce (EC)*, ACM, pages 73–82. 2010.

[5] J. Cardinal and M. Hoefer. Non-cooperative facility location and covering games. *Theoretical Computer Science*, 411(16-18): 1855–1876, 2010.

[6] M. Charikar, H. J. Karloff, C. Mathieu, J. Naor, and M. E. Saks. Online multicast with egalitarian cost sharing. In *Proceedings of the 20th Annual ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, ACM, pages 70–76, 2008.

[7] C. Chekuri, J. Chuzhoy, L. Lewin-Eytan, J. Naor, and A. Orda. Non-cooperative multicast and facility location games. *IEEE Journal on Selected Areas in Communications*, 25(6): 1193–1206, 2007.

[8] S. Chien and A. Sinclair. Convergence to approximate Nash equilibria in congestion games. *Games and Economic Behavior*, 71(2): 315-327, 2011.

[9] N. Christin, J. Grossklags, and J. Chuang. Near rationality and competitive equilibria in networked systems. In *Proceedings of the ACM SIGCOMM Workshop on Practice and Theory of Incentives in Networked Systems (PINS)*, ACM, pages 213–219, 2004.

[10] G. Christodoulou, E. Koutsoupias, and P. G. Spirakis. On the performance of approximate equilibria in congestion games. In *Proceedings of the 17th Annual Symposium on Algorithms (ESA)*, LNCS 5757, Springer, pages 251–262, 2009.

[11] G. Christodoulou, V. S. Mirrokni, and A. Sidiropoulos. Convergence and approximation in potential games. In *Proceedings of the 23rd Annual Symposium on Theoretical Aspects of Computer Science (STACS)*, LNCS 3884, Springer, pages 349–360, 2006.

[12] C. Daskalakis and C. H. Papadimitriou. Computing equilibria in anonymous games. In *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, IEEE, pages 83–93, 2007.

[13] D. S. Johnson, C. H. Papadimitriou, and M. Yannakakis. How easy is local search? *Journal of Computer and System Sciences*, 37:79–100, 1988.

[14] A. Fabrikant, C. H. Papadimitriou, and K. Talwar. The complexity of pure nash equilibria. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing (STOC)*, ACM, pages 604–612, 2004.

[15] A. Fanelli, M. Flammini, and L. Moscardelli. The speed of convergence in congestion games under best-response dynamics. In *Proceedings of the 35th International Colloquium on Automata, Languages and Programming (ICALP)*, Part 1, LNCS 5125, Springer, pages 796–807, 2008.

[16] E. Koutsoupias and C. H. Papadimitriou. Worst-case equilibria. In *Proceedings of the 16th Annual Symposium on Theoretical Aspects of Computer Science (STACS)*, LNCS 1563, Springer, pages 404–413, 1999.

[17] V. S. Mirrokni and A. Vetta. Convergence issues in competitive games. In *Proceedings of the 7th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX)*, LNCS 3122, Springer, pages 183-192, 2004.

[18] D. Monderer and L. S. Shapley. Potential games. *Games and Economic Behavior*, 14(1):124–143, 1996.

[19] T. Nguyen and E. Tardos. Approximate pure Nash equilibria via Lovász local lemma. In *Proceedings of the 5th International Workshop on Internet and Network Economics (WINE)*, LNCS 5929, Springer, pages 160–171, 2009.

[20] R. W. Rosenthal. A class of games possessing pure-strategy Nash equilibria. *International Journal of Game Theory*, 2:65–67, 1973.

[21] T. Roughgarden. Intrinsic robustness of the price of anarchy. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC)*, ACM, pages 513–522, 2009.

[22] A. Skopalik and B. Vöcking. Inapproximability of pure Nash equilibria. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC)*, ACM, pages 355–364, 2008.