Generalized Mirror Descents in Congestion Games with Splittable Flows

Po-An Chen*
National Chiao Tung University
1001 University Road
Hsinchu, Taiwan
poanchen@nctu.edu.tw

Chi-Jen Lu
Academia Sinica
128 Academia Road, Section 2
Nankang, Taipei 115, Taiwan
cjlu@iis.sinica.edu.tw

ABSTRACT

Different types of dynamics have been studied in repeated game play, and one of them which has received much attention recently consists of those based on "no-regret" algorithms from the area of machine learning. It is known that dynamics based on generic no-regret algorithms may not converge to Nash equilibria in general, but to a larger set of outcomes, namely coarse correlated equilibria. Moreover, convergence results based on generic no-regret algorithms typically use a weaker notion of convergence: the convergence of the average plays instead of the actual plays. Some work has been done showing that when using a specific no-regret algorithm, the well-known multiplicative updates algorithm, convergence of actual plays to equilibria can be shown and better quality of outcomes can be reached for atomic congestion games and load balancing games. Are there more cases of natural no-regret dynamics that perform well in suitable classes of games in terms of convergence and quality of outcomes? We answer this question positively by showing that when each player individually employs the *mirror-descent* algorithm, a well-known generic no-regret algorithm, the *actual* plays converge quickly to equilibria in nonatomic congestion games. This gives rise to a family of algorithms, including the multiplicative updates algorithm and the gradient descent algorithm as well as many others. Furthermore, we show that our dynamics achieves good bounds on the quality of outcomes measured by two different social costs: the average individual cost and the maximum individual cost.

Categories and Subject Descriptors

Theory of computation [**Theory and algorithms for application domains**]: Algorithmic game theory and mechanism design—*Convergence and learning in games*

Keywords

Mirror-descent algorithm; No-regret dynamics; Convergence

*Supported in part by NSC 102-2221-E-009-061-MY2.

Appears in: *Alessio Lomuscio, Paul Scerri, Ana Bazzan, and Michael Huhns (eds.), Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014), May 5-9, 2014, Paris, France.*

1. INTRODUCTION

Nash equilibrium is a widely-adopted solution concept in game theory, which is used for predicting the outcomes of systems consisting of self-interested players. We are interested in repeated games, and a Nash equilibrium describes a steady state in which the system would stay once it is reached. However, this raises the issue of how such a state can be reached. In fact, for a general game, computing a Nash equilibrium is believed to be hard (according to the PPAD-hardness results), so an equilibrium may not be reached in a reasonable amount of time in general, and the outcomes we have observed may all be far out of any equilibrium, which would render the study on equilibria meaningless. To address this issue, a line of research is to consider natural dynamics which players have incentive to follow, and study how the system evolves according to such dynamics. One natural dynamics is the best or better response dynamics, in which a deviating player at each time makes a best or better change in his/her strategy to improve his/her payoff given the current choice of the other players. It is well-known that such dynamics leads to pure Nash equilibria in congestion games. However, a player may not have incentive to play this way because making such deviations may not be beneficial if other players also deviate at the same time.

One may argue that a plausible incentive for a player is to maximize his/her average payoff through the time, and dynamics based on "no-regret" algorithms from the area of online learning [9] have thus been proposed in the study. For a nonatomic routing game, it is known that if each player plays any arbitrary no-regret algorithm, the "time-averaged" flow (and flows at *most* time steps) converges to some type of approximate Nash equilibrium [7]. For a "socially concave" game, a similar time-averaged convergence result is also known [11][1]. However, convergence to a Nash or approximate Nash equilibrium is not always the case in general, and playing arbitrary no-regret algorithms can result in a larger set of outcomes than Nash equilibria, namely *coarse correlated equilibria*. Nevertheless, if one only cares about the outcome quality and the quality is measured by the average individual cost, it is known that the total price of anarchy achieved by such no-regret algorithms can still match the price of anarchy at Nash equilibrium in special games, such as atomic congestion games [8] or an even wider class of smooth games [16]. On the other hand, there are

[1]Note that if we change convexity to concavity and costs to utilities in our paper, games that we consider here are not socially concave.

broad classes of games and natural measures of outcome quality for which large gaps are known between no-regret outcomes and Nash equilibria. Furthermore, all the convergence results mentioned above are about the convergence of the time-averaged strategy instead of the actual strategy.[2] That is, what converges to a Nash equilibrium is the average of all the strategies played so far, instead of the actual strategy played at the moment. Such results are useful if the goal is to solve the computational problem of computing an approximate Nash equilibrium, but they may not tell us much about how the system actually evolves. In particular, even though the time-averaged strategy converges to an equilibrium, the actual strategy may not converge and may be far away from an equilibrium.

Although it is nice to be able to have general positive results on what generic no-regret algorithms can achieve, one may wonder if going from generic no-regret algorithms to specific ones could yield stronger results, in terms of convergence or quality of outcomes. One of the best known no-regret algorithms is the *Multiplicative Updates* (MU) algorithm. Kleinberg et al. [15] studied this for atomic congestion games in the full-information setting, in which players have full information about the cost functions so that they can determine the cost of every other strategy they could have used given other players strategies at the current round. It was shown that if each player employs such an MU algorithm, the actual joint strategy profile of players converges to a pure Nash equilibrium with high probability for most games. Note that here it is the actual joint strategy profile, instead of the time-averaged one, which converges. Furthermore, since the set of pure Nash equilibria can be a very small subset of correlated equilibria, the price of total anarchy achieved this way can be much smaller than that by a generic no-regret algorithm.

In another work [14], Kleinberg et al. studied the smaller class of load balancing games, but in the more stringent partial-information setting of the "bulletin board" model, in which players only know the actual cost value of each edge according to the actual strategies played at the current round. They showed that if all the players play according to a *common* distribution (mixed strategy) and update the distribution using such an MU algorithm, the common distribution converges to some symmetric Nash equilibrium of the *nonatomic* version of the game. As a result, the price of total anarchy achieved this way is also considerably smaller than that by a generic no-regret one. However, their analysis relies crucially on the assumption that all the players at each round play according to the same distribution. This assumption may not be reasonable in other settings or in other games, which makes the applicability of their analysis somewhat limited. On the other hand, the analysis in [15] can do without the assumption and deal with general asymmetry in players' probability distributions, but it only works in the full-information model.

These results, which form a good comparison and complement to each other, along with the results on generic no-regret plays motivate our quest for other classes of learning dynamics in possibly other classes of games and settings. Are there more cases of natural no-regret dynamics that perform well in suitable classes of games in terms of convergence time and quality of outcomes?

[2]Although it was also shown in [7] that flows at most time steps are close to equilibria, the guarantee is still *not* on the convergence of the actual plays.

Our Contributions. We answer this question positively. We provide a family of such dynamics in the bulletin model for the class of nonatomic congestion games with cost functions of bounded slopes. More precisely, we show that in such a game, if each player individually plays some type of the *mirror-descent* algorithm [5], a well-known general no-regret algorithm, then their joint strategy profile quickly converges to a Wardrop equilibrium. We also show that our dynamics achieves good bounds on the quality of outcomes measured by two different social costs: the average individual cost and the maximum individual cost.

The mirror-descent algorithm in fact can be seen as a family of algorithms. By instantiating it properly, one can recover the MU algorithm, the gradient-descent algorithm, as well as many others, and our result establishes the fast convergence of all these algorithms at once. Let us stress that as in [15, 14], our notion of convergence is the stronger one: what converges is the actual joint strategy profile, instead of the time-averaged one as in [7, 11]. Note that in the congestion game, different players naturally have different sets of strategies, so it is no longer reasonable to assume that all the players use the same distribution to play as in [14]. Therefore, we allow players to use different distributions and moreover, we allow players to update according to different learning rates. Still, we manage to prove the convergence, just as [15] but in the more difficult bulletin model and with a concrete bound on convergence time.

Furthermore, we provide bounds on the price of total anarchy achieved by our dynamics, in terms of the average individual cost and the maximum individual cost. Using the average individual cost as the social cost, we show that the ratio between the social cost achieved by our dynamics and the optimal one approaches some constant, which depends on the slopes of the cost functions. Using the maximum individual cost as the social cost, we show that the ratio between the social cost achieved by our dynamics and the optimal one also approaches the same constant in symmetric games. In each case, there is a tradeoff between the ratio we can achieve and the time it takes: by letting the system evolve for a longer time, it will get closer to an equilibrium, and the resulting ratio will approach closer to that constant.

Our main technical contribution is the convergence of our dynamics to an approximate equilibrium. To show this, we consider a smooth *convex* potential function of the game which has the joint strategy profile of players as its input. The interesting observation is that although each player individually applies the mirror descent algorithm to his/her own strategy using costs related only to him/her, we show that the updates performed by all the players collectively can be seen as following some generalized mirror descent process on the potential function. The generalized mirror descent allows different step sizes in different dimensions, and we need this generalization because we allow different learning rates for different players. The standard mirror descent, on the other hand, has the same step size across all the dimensions, so that it moves at each time in exactly the opposite direction of the gradient vector. It is known that doing the standard mirror descent on a smooth convex function leads to a fast convergence to its minimum [6]. However, our generalized mirror descent no longer moves in the opposite direction of the gradient vector as different step sizes have different scaling effects in different dimensions, and therefore it is not clear if the process would still converge. Interestingly, we show that a similar convergence result can also be achieved, which may have independent interest of its own. Finally, let us remark that the standard mirror descent algorithm, instead of the generalized one, has also been used for different problems in game theory: for finding market equilibria in Fisher markets [6] and convex potential markets [10]. Our convergence result for the generalized mirror descent algorithm is an extension of that for the standard one.

We provide definitions and some preliminaries in Section 2. Then, the convergence result is presented in Section 3, which is followed by the outcome quality bounds in Section 4. We summarize with conclusions and future work in Section 5.

2. PRELIMINARIES

In this paper, we consider the following congestion game described by $(N, E, (\mathcal{S}_i)_{i\in N}, (c_e)_{e\in E})$, where N is the set of players, E is the set of edges (resources), $\mathcal{S}_i \subseteq 2^E$ is the collection of allowed paths (subsets of resources) for player i, and c_e is the cost function of edge e, which is a nondecreasing function of the amount of load on it. Let us assume that $N = \{1, \ldots, n\}$, $|E| = m$, and each player has a load of $1/n$ (so the total load is 1). The strategy of each player i is to distribute her load over her allowed paths, which can be represented by a $|\mathcal{S}_i|$-dimensional vector $x_i = (x_{i,s})_{s\in\mathcal{S}_i}$, where $x_{i,s} \in [0, 1]$ is the amount of the load player i puts on the path s. Note that $\sum_{s\in\mathcal{S}_i} x_{i,s} = 1/n$ and let $\mathcal{K}_i$ be the feasible set for all such vectors x_i. Then the strategies of all players can be jointly represented by a vector

$$x = (x_1, ..., x_n) = ((x_{1,s})_{s\in\mathcal{S}_1}, ..., (x_{n,s})_{s\in\mathcal{S}_n}) \in \mathbb{R}^d,$$

where $d = \sum_{i\in N} |\mathcal{S}_i|$, and let $\mathcal{K} = \mathcal{K}_1 \times \cdots \times \mathcal{K}_n$ be the feasible set for all such vectors x. We call x_i the flow of player i and x the flow of the system.[3] Note that an edge $e \in E$ can be shared by different paths, and the aggregated load on e, denoted by $\ell_e(x)$, is $\sum_{s:e\in s}\sum_{i\in N} x_{i,s}$. Then the cost of a path s is defined as $c_s(x) = \sum_{e\in s} c_e(\ell_e(x))$, and the individual cost of player i is defined as $C_i(x) = \sum_{s\in\mathcal{S}_i} x_{i,s}c_s(x)$. The game is called an *atomic splittable congestion game* in [17] and others when the number of players is finite.

An alternative *nonatomic* definition of the game is that each player consists of a huge (infinite) number of agents (or see each player as a group of players of the same type). The agents of player i split the load of player i so that each has a small (infinitesimal) amount Δ of load. Each agent of player i now must choose one single path s from $\mathcal{S}_i$ and put that Δ amount of load all on s, and the cost for that is $\Delta c_s(x)$. Now $\sum_{s\in\mathcal{S}_i} x_{i,s}c_s(x)$ becomes the total cost of all agents of player i, where $x_{i,s}$ is the total load on path s from all agents of player i. It is not hard to check that the two definitions of the game coincide when Δ approaches 0. We will mainly use expressions from the first definition yet essentially with an infinite number of infinitesimal agents for each player of a nonatomic congestion game, as it makes our presentation simpler.

[3]Although we borrow the terms such as edge, path, and flow from routing games, the congestion games are more general as there are no underlying graphs and a path can be just any arbitrary subset of edges.

Such a game admits the following potential function:[4]

$$\Phi(x) = \sum_e \int_0^{\ell_e(x)} c_e(y)dy. \quad (1)$$

To see that this is indeed a potential function, note that if some player deviates an infinitesimal fraction of load from s to s' (where $x_{i,s} > 0$) such that $c_s(x) > c_{s'}(x')$ (where x is almost the same as x' except for the small fraction of moved load), then $\partial\Phi(x)/\partial x_{i,s} > \partial\Phi(x')/\partial x_{i,s'}$, which means that the rate of decrease in Φ is larger than the rate of increase in Φ and thus the resulting Φ decreases. We will need the following, which we prove in Appendix A.

PROPOSITION 1. *The function Φ defined in (1) is convex.*

As in [14], we assume that the cost functions satisfy the property that for any $y \in [0, 1]$ and any $e \in E$, $c_e(0) = 0$, $c_e(1) \le 1$, $c'_e(y) \ge A > 0$ and $0 \le c''_e(y) \le B$, where A, B are positive constants. By Lemma 3.8 of [14], for constants $b_0 = A$ and $b_1 = B + 1$, the cost functions satisfy the condition that

$$b_0 y \le c_e(y) \le b_1 y, \text{ for any } y \in [0, 1]. \quad (2)$$

Then we have the following, which we prove in Appendix B.

PROPOSITION 2. *For any $\xi \in \mathcal{K}$, $\nabla^2\Phi(\xi) \preceq \alpha I$ with $\alpha = dmb_1$.*

We consider two types of social cost functions. The first is the average individual cost function, defined as

$$C_A(x) = \sum_e \ell_e(x)c_e(\ell_e(x)),$$

and the second is the maximum individual cost function, defined as

$$C_M(x) = \max_{s\in\mathcal{S}} \sum_{e\in s} c_e(\ell_e(x)), \text{ where } \mathcal{S} = \bigcup_{i\in N} \mathcal{S}_i.$$

Using them, we measure the quality of outcome for a flow $x \in \mathcal{K}$ in the following two ways. The first is the ratio $C_A(x)/C_A(x^*)$, where $x^* = \arg\min_{z\in\mathcal{K}} C_A(z)$, and the second is the ratio $C_M(x)/C_M(\hat{x})$, where $\hat{x} = \arg\min_{z\in\mathcal{K}} C_M(z)$.

3. DYNAMICS AND CONVERGENCE

We consider the setting in which the players play the game iteratively in the following way. At step t, each player i plays the strategy x_i^t by sending the amount $x_{i,s}^t$ of load on path s for each $s \in \mathcal{S}_i$. After that, she gets to know the vector $\hat{c}_i^t = (c_s(x^t))_{s\in\mathcal{S}_i}$ of cost values, where $c_s(x^t) = \sum_{e\in s} c_e(\ell_e(x^t))$ is the cost value on the path s at that step. With this, she updates her next strategy x_i^{t+1} in some way and then proceeds to the next iteration. In the alternative definition of the game, the corresponding setting is that at step t, each agent of player i sends its load of Δ all on some path $s \in \mathcal{S}_i$, which is chosen according to some distribution. We assume that all agents of player i start with the same initial distribution and update their distributions at each step t using the same algorithm according to the same information $\hat{c}_i^t$. Then we can conclude that their distributions at step t

[4]Note that our convergence result will be proved more generally for any convex potential function satisfying certain properties.

are all the same,[5] which basically can be described by the flow x_i^t of player i, due to the law of large number as the number of agents is huge. Thus, the settings for the two definitions of the game also match.

We have not specified how the players or agents of players update their next strategies. Different update algorithms may make the whole system evolve in rather different ways, and we would like to understand if there are update algorithms which players or agents of players have incentive to adopt that can lead to desirable outcomes for the whole system. One can argue that a plausible incentive for a player is to minimize her regret. Two well-known no-regret algorithms are the gradient descent algorithm and the multiplicative update algorithm, both of which can be seen as special cases of a more general algorithm called mirror descent algorithm (see e.g. [9] for more detail). Inspired by this, we consider the following update rule for player i or agents of player i:

$$\begin{aligned} x_i^{t+1} &= \arg\min_{z_i \in \mathcal{K}_i} \left\{ \eta_i \langle \hat{c}_i^t, z_i \rangle + \mathcal{B}^{R_i}(z_i, x_i^t) \right\} && (3) \\ &= \arg\min_{z_i \in \mathcal{K}_i} \mathcal{B}^{R_i}(z_i, x_i^t - \eta_i \hat{c}_i^t). && (4) \end{aligned}$$

Here, $\eta_i > 0$ is some learning rate, $R_i : \mathcal{K}_i \to \mathbb{R}$ is some regularization function, and $\mathcal{B}^{R_i}(\cdot,\cdot)$ is the Bregman divergence with respect to R_i defined as

$$\mathcal{B}^{R_i}(u_i, v_i) = R_i(u_i) - R_i(v_i) - \langle \nabla R_i(v_i), u_i - v_i \rangle$$

for $u_i, v_i \in \mathcal{K}_i$. This gives rise to a family of update rules for different choices of the function R_i. For example, it is well-known that by choosing $R_i(u_i) = \|u_i\|_2^2/2$, one recovers the gradient descent algorithm, while by choosing $R_i(u_i) = \sum_s (u_{i,s} \ln u_{i,s} - u_{i,s})$, one recovers the multiplicative update algorithm. Using a similar argument as in [14], one can show that this algorithm, with a properly chosen R_i, is indeed a no-regret algorithm for each agent i, and this provides an incentive for the agents to use the algorithm. The requirement on these R_i's which we need is that the function Φ is "smooth" with respect to them in the following sense.

Definition 1. *We say that Φ is λ-smooth with respect to $(R_1, \ldots, R_n)$ if for any two inputs $x = (x_1, \ldots, x_n)$ and $x' = (x'_1, \ldots, x'_n)$ in $\mathcal{K}$,*

$$\Phi(x') \le \Phi(x) + \langle \nabla \Phi(x), x' - x \rangle + \lambda \sum_{i=1}^{n} \mathcal{B}^{R_i}(x'_i, x_i). \quad (5)$$

Our main result in this section is the following, which shows that if each player (or agent of a player) uses such an update algorithm, the system quickly converges, in the sense that the value of the potential function $\Phi(x^t)$ quickly approaches the minimum $\Phi(q)$, where $q = \arg\min_{z \in \mathcal{K}} \Phi(z)$. Implications of $\Phi(x^t)$ being close to $\Phi(q)$ will be given in Section 4, including x^t being an approximate equilibrium and achieving social efficiency.

Theorem 3. *Consider any nonatomic congestion game of n players, with a potential function Φ which is λ-smooth with respect to some $(R_1, \ldots, R_n)$. Let $q = (q_1, \ldots, q_n) = \arg\min_{z \in \mathcal{K}} \Phi(z)$. Now suppose that each player i starts from some initial strategy x_i^0, with $\mathcal{B}^{R_i}(q_i, x_i^0) \le \gamma$, and updates her strategy according to the rule in (3), with $\eta_i \in [\eta, 1/\lambda]$ for some η. Then for any $\varepsilon \in (0,1)$ there exists some $T_\varepsilon \le n\gamma/(\eta\varepsilon)$ such that for any $t \ge T_\varepsilon$, $\Phi(x^t) \le \Phi(q) + \varepsilon$.*

[5]The distributions of agents from different players are still different in general.

From this, we have the following, which we will prove in Section 3.2.

Corollary 4. *Consider any nonatomic congestion game of n players with parameters given in Section 2, and let $\lambda = mb_1 d$. Now if each player i plays the gradient descent algorithm by starting from any $x_i^0 \in \mathcal{K}_i$ and using any $\eta_i \in [\eta, 1/\lambda]$, then $T_\varepsilon \le 2/(n\eta\varepsilon)$. Furthermore, if each player i plays the multiplicative update algorithm by starting from a uniform x_i^0 (same load on each allowed path) and using any $\eta_i \in [\eta, 1/\lambda]$, then $T_\varepsilon \le (n \ln(dn))/(\eta\varepsilon)$.*

Remark 1. *According to Corollary 4, playing the gradient descent algorithm guarantees a faster convergence time. In particular, if each player i uses $\eta_i = 1/\lambda$, then adopting the gradient descent algorithm leads to a convergence time $T_\varepsilon \le 2mb_1 d/(n\varepsilon)$, while adopting the multiplicative update algorithm leads to $T_\varepsilon \le (mb_1 dn \ln(dn))/\varepsilon$.*

To prove Theorem 3, the key observation is that the updates by all players collectively can be seen as doing a generalized version of the mirror descent, with different step sizes in different dimensions, on the potential function Φ defined in (1). To see this, note that for any $i \in N$ and $s \in \mathcal{S}_i$, the s'th entry of $\hat{c}_i^t$ is

$$c_s(x^t) = \sum_{e \in s} c_e(\ell_e(x^t)) = \frac{\partial \Phi(x^t)}{\partial x_{i,s}},$$

which means that the d-dimensional vector $(\hat{c}_i^t)_{i \in N}$ is in fact equal to $\nabla \Phi(x^t)$, the gradient of Φ at x^t. That is, if we write $\nabla \Phi(x^t) = (\nabla_1 \Phi(x^t), \ldots, \nabla_n \Phi(x^t)$, with $\nabla_i \Phi(x^t)$ being the portion of $\nabla \Phi(x^t)$ corresponding to player i, then the update rule of (3) and (4) becomes the following:

$$\begin{aligned} x_i^{t+1} &= \arg\min_{z_i \in \mathcal{K}_i} \left\{ \eta_i \langle \nabla_i \Phi(x^t), z_i \rangle + \mathcal{B}^{R_i}(z_i, x_i^t) \right\} && (6) \\ &= \arg\min_{z_i \in \mathcal{K}_i} \mathcal{B}^{R_i}(z_i, x_i^t - \eta_i \nabla_i \Phi(x^t)). && (7) \end{aligned}$$

Observe that when all the η_i's are identical, the collective update of all players moves the whole system exactly in the direction of $-\nabla \Phi(x^t)$, and this becomes the standard mirror descent algorithm which has the same step size across all dimensions. It is known that doing such a mirror descent on a smooth convex function leads to a fast convergence to its minimum [6]. On the other hand, we consider the more general case in which different players can have different learning rates, and this corresponds to a more general mirror descent algorithm which allows different step sizes in different dimensions. Because the different step sizes have different scaling effects in different dimensions, the collective update now no longer moves the whole system in the direction of $-\nabla \Phi(x^t)$, and it is not clear if a similar convergence result can be obtained. Interestingly, the following theorem shows that doing such a generalized mirror descent algorithm on a general smooth convex function still gives us a fast convergence to its minimum.

Theorem 5. *Suppose $\mathcal{K} = \mathcal{K}_1 \times \cdots \times \mathcal{K}_n$, with each $\mathcal{K}_i$ being a convex set. Let $\Phi : \mathcal{K} \to \mathbb{R}$ be any convex function which is λ-smooth with respect to some $(R_1, \ldots, R_n)$ and let*

$q = (q_1, \ldots, q_n) = \arg\min_{z \in \mathcal{K}} \Phi(z)$. *Suppose we start from some* $x^0 = (x_1^0, \ldots, x_n^0)$, *with each* $\mathcal{B}^{R_i}(q_i, x_i^0) \le \gamma$, *and then use the update rule in (6), with each* $\eta_i \in [\eta, 1/\lambda]$ *for some* η. *Then for any* $\varepsilon \in (0,1)$, *there exists some* $T_\varepsilon \le n\gamma/(\eta\varepsilon)$ *such that for any* $t \ge T_\varepsilon$, $\Phi(x^t) \le \Phi(q) + \varepsilon$.

We will prove Theorem 5 in Section 3.1. Now note that Theorem 3 follows immediately from Theorem 5 since our potential function Φ is convex by Proposition 1. On the other hand, Theorem 5 works for a general convex function (not restricted to the specific potential function given in (1)), which may have independent interest of its own.

3.1 Proof of Theorem 5

Our proof follows closely that in [6] for the special case in which all the η_i's are identical. To simplify our notation, let us denote the gradient vector $\nabla\Phi(x^t)$ by $g^t = (g_1^t, \ldots, g_n^t)$, with $g_i^t = \nabla_i \Phi(x^t)$.

Using the assumption that for each i, $\eta_i \le 1/\lambda$ and thus $\lambda \le 1/\eta_i$, the λ-smoothness condition implies that

$$\Phi(x^{t+1}) \le \Phi(x^t) + \langle g^t, x^{t+1} - x^t\rangle + \sum_{i=1}^{n} \frac{1}{\eta_i} \mathcal{B}^{R_i}(x^{t+1}, x^t), \quad (8)$$

because each $\mathcal{B}^{R_i}(x^{t+1}, x^t)$ is nonnegative. Then we need the following two lemmas, which we will prove later.

LEMMA 6. *For any integer* $t \ge 0$, $\Phi(x^{t+1}) \le \Phi(x^t)$.

LEMMA 7. *For any integer* $T \ge 1$,

$$\sum_{t=0}^{T-1} \left(\Phi(x^{t+1}) - \Phi(q)\right) \le \sum_{i=1}^{n} \frac{1}{\eta_i} \mathcal{B}^{R_i}(q_i, x_i^0).$$

Combining these two lemmas together, we obtain

$$\begin{aligned} T\left(\Phi(x^T) - \Phi(q)\right) &\le \sum_{t=0}^{T-1} \left(\Phi(x^{t+1}) - \Phi(q)\right) \\ &\le \sum_{i=1}^{n} \frac{1}{\eta_i} \mathcal{B}^{R_i}(q_i, x_i^0) \\ &\le \frac{n\gamma}{\eta}. \end{aligned}$$

Dividing both sides by T gives us

$$\Phi(x^T) - \Phi(q) \le \frac{n\gamma}{\eta T} \le \varepsilon,$$

when $T \ge n\gamma/(\eta\varepsilon)$, and we have the theorem. It remains to prove the two lemmas, which we do next.

PROOF OF LEMMA 6. We know from (8) that

$$\Phi(x^{t+1}) \le \Phi(x^t) + \sum_{i=1}^{n} \left(\langle g_i^t, x_i^{t+1} - x_i^t\rangle + \frac{1}{\eta_i} \mathcal{B}^{R_i}(x_i^{t+1}, x_i^t) \right).$$

To bound the sum above, note that according to the definition of x_i^{t+1} in (6), we have

$$\begin{aligned} &\langle g_i^t, x_i^{t+1} - x_i^t\rangle + \frac{1}{\eta_i} \mathcal{B}^{R_i}(x_i^{t+1}, x_i^t) \\ \le\; &\langle g_i^t, x_i^t - x_i^t\rangle + \frac{1}{\eta_i} \mathcal{B}^{R_i}(x_i^t, x_i^t) \\ =\; &0. \end{aligned}$$

Applying this to the above bound on $\Phi(x^{t+1})$, Lemma 6 follows. □

PROOF OF LEMMA 7. We know from (8) that for any $t \ge 0$, $\Phi(x^{t+1})$ is at most

$$\Phi(x^t) + \langle g^t, x^{t+1} - x^t\rangle + \sum_{i=1}^{n} \frac{1}{\eta_i} \mathcal{B}^{R_i}(x^{t+1}, x^t),$$

where the second term above can be expressed as

$$\begin{aligned} \langle g^t, x^{t+1} - x^t\rangle &= \langle g^t, q - x^t\rangle + \langle g^t, x^{t+1} - q\rangle \\ &= \langle g^t, q - x^t\rangle + \sum_{i=1}^{n} \langle g_i^t, x_i^{t+1} - q_i\rangle. \end{aligned}$$

Since $\Phi(x^t) + \langle g^t, q - x^t\rangle \le \Phi(q)$ for a convex Φ, we thus know that $\Phi(x^{t+1})$ is at most

$$\Phi(q) + \sum_{i=1}^{n} \left(\langle g_i^t, x_i^{t+1} - q_i\rangle + \frac{1}{\eta_i} \mathcal{B}^{R_i}(x^{t+1}, x^t) \right). \quad (9)$$

To bound the sum above, we rely on the following.

PROPOSITION 8. *For each* i, $\langle g_i^t, x_i^{t+1} - q_i\rangle$ *is at most*

$$\frac{1}{\eta_i} \left(\mathcal{B}^{R_i}(q_i, x_i^t) - \mathcal{B}^{R_i}(q_i, x_i^{t+1}) - \mathcal{B}^{R_i}(x_i^{t+1}, x_i^t) \right).$$

PROOF. According to the definition of x_i^{t+1} in (6), it is also the minimizer of the function

$$L(z) = \eta_i \langle g_i^t, z - q_i\rangle + \mathcal{B}^{R_i}(z, x_i^t)$$

over $z \in \mathcal{K}_i$, since $\langle g_i^t, -q_i\rangle$ is a constant independent of z. Then from a well-known fact in convex optimization, we know that

$$\langle \nabla L(x_i^{t+1}), q_i - x_i^{t+1}\rangle \ge 0.$$

Since $\nabla L(x_i^{t+1}) = \eta_i g_i^t + \nabla R_i(x_i^{t+1}) - \nabla R_i(x_i^t)$, we have

$$\eta_i \langle g_i^t, x_i^{t+1} - q_i\rangle \le \left\langle \nabla R_i(x_i^{t+1}) - \nabla R_i(x_i^t), q_i - x_i^{t+1} \right\rangle. \quad (10)$$

Then according to the definition of $\mathcal{B}^{R_i}(\cdot)$, we have

$$\begin{aligned} &\mathcal{B}^{R_i}(q_i, x_i^t) \\ &\quad = R_i(q_i) - R_i(x_i^t) - \langle \nabla R_i(x_i^t), q_i - x_i^t\rangle, \\ &\mathcal{B}^{R_i}(q_i, x_i^{t+1}) \\ &\quad = R_i(q_i) - R_i(x_i^{t+1}) - \langle \nabla R_i(x_i^{t+1}), q_i - x_i^{t+1}\rangle, \text{ and} \\ &\mathcal{B}^{R_i}(x_i^{t+1}, x_i^t) \\ &\quad = R_i(x_i^{t+1}) - R_i(x_i^t) - \langle \nabla R_i(x_i^t), x_i^{t+1} - x_i^t\rangle. \end{aligned}$$

By subtracting the second and the third equalities from the first, we obtain

$$\begin{aligned} &\mathcal{B}^{R_i}(q_i, x_i^t) - \mathcal{B}^{R_i}(q_i, x_i^{t+1}) - \mathcal{B}^{R_i}(x_i^{t+1}, x_i^t) \\ &\quad = \left\langle \nabla R_i(x_i^{t+1}) - \nabla R_i(x_i^t), q_i - x_i^{t+1} \right\rangle. \end{aligned}$$

Substituting this into (10) proves the proposition. □

Combining the bound from this proposition with the upper bound on $\Phi(x^{t+1})$ in (9), we obtain

$$\Phi(x^{t+1}) \le \Phi(q) + \sum_{i=1}^{n} \frac{1}{\eta_i} \left(\mathcal{B}^{R_i}(q_i, x_i^t) - \mathcal{B}^{R_i}(q_i, x_i^{t+1}) \right).$$

This implies that

$$\begin{aligned}\sum_{t=0}^{T-1}&\left(\Phi(x^{t+1})-\Phi(q)\right)\\ &\leq \sum_{i=1}^{n}\frac{1}{\eta_i}\sum_{t=0}^{T-1}\left(\mathcal{B}^{R_i}(q,x_i^t)-\mathcal{B}^{R_i}(q,x_i^{t+1})\right)\\ &\leq \sum_{i=1}^{n}\frac{1}{\eta_i}\mathcal{B}^{R_i}(q_i,x_i^0),\end{aligned}$$

which proves Lemma 7.

3.2 Proof of Corollary 4

Let us first consider the case that each player plays the gradient descent algorithm. Note that this corresponds to choosing $R_i(u_i)=\|u_i\|_2^2/2$ for each i, and one can show that $\mathcal{B}^{R_i}(u_i,v_i)=\|u_i-v_i\|_2^2/2$, for $u_i,v_i\in\mathcal{K}_i$. Then, we have

$$\mathcal{B}^{R_i}(q_i,x_i^0)=\|q_i-x_i^0\|_2^2/2\leq\|q_i-x_i^0\|_1^2/2$$

which is at most

$$\left(\|q_i\|_1+\|x_i^0\|_1\right)^2/2\leq 2/n^2.$$

Therefore, we can choose $\gamma=2/n^2$ to have $\mathcal{B}^{R_i}(q_i,x_i^0)\leq\gamma$. Furthermore, using the Taylor expansion together with Proposition 2, we know that for any $x,x'\in\mathcal{K}$,

$$\Phi(x')\leq\Phi(x)+\langle\nabla\Phi(x),x'-x\rangle+\alpha\|x'-x\|_2^2/2,$$

with $\alpha=mb_1d$. Since

$$\|x'-x\|_2^2/2=\sum_i\|x_i'-x_i\|_2^2/2=\sum_i\mathcal{B}^{R_i}(x_i',x_i),$$

we can choose $\lambda=\alpha$ to guarantee that Φ is λ smooth.

Next, let us consider the case that each player plays the multiplicative update algorithm. Note that this corresponds to choosing $R_i(u_i)=\sum_s(u_{i,s}\ln u_{i,s}-u_{i,s})$ for each i, and one can show that $\mathcal{B}^{R_i}(u_i,v_i)=\sum_s u_{i,s}\ln(u_{i,s}/v_{i,s})$, for $u_i,v_i\in\mathcal{K}_i$. Then, we have

$$\mathcal{B}^{R_i}(q_i,x_i^0)\leq\sum_s q_{i,s}\ln(|\mathcal{S}_i|n)\leq\ln(dn).$$

Therefore, we can choose $\gamma=\ln(dn)$ to have $\mathcal{B}^{R_i}(q_i,x_i^0)\leq\gamma$. Furthermore, we know that

$$\|x_i'-x_i\|_2^2/2\leq\|x_i'-x_i\|_1^2/2\leq\mathcal{B}^{R_i}(x_i',x_i),$$

by Pinsker's inequality. Therefore, we can again choose $\lambda=\alpha$ to guarantee that Φ is λ smooth.

Substituting these bounds of γ and λ into Theorem 3, Corollary 4 then follows.

4. EQUILIBRIUM, SOCIAL EFFICIENCY, AND MAKESPAN

According to Theorem 3, the flow x^t at step $t\geq T_\varepsilon$ enjoys the nice property that $\Phi(x^t)\leq\Phi(q)+\varepsilon$. In this section, we show the implication of this property.

4.1 Approximate Equilibrium

We say that a flow $x\in\mathcal{K}$ is an δ-equilibrium if for any player $i\in N$ and any paths $s,s'\in\mathcal{S}_i$ with $x_{i,s}>0$, $c_s(x)\leq c_{s'}(x)+\delta$. Note that with $\delta=0$, we recover the standard definition of equilibrium for nonatomic games. The following shows that after the convergence time, the system playing our algorithm will stay in an δ-equilibrium for a small δ.

THEOREM 9. *Any $x\in\mathcal{K}$ such that $\Phi(x)\leq\Phi(q)+\varepsilon$ must be a δ-equilibrium for some $\delta\leq\sqrt{8b_1m\varepsilon}$.*

PROOF. Consider any $x\in\mathcal{K}$ such that $\Phi(x)\leq\Phi(q)+\varepsilon$ and any $i\in N$. Let s_0 be the path in $\mathcal{S}_i$ which minimizes $c_s(x)$ among $s\in\mathcal{S}_i$, and let s_1 be the path which maximizes $c_s(x)$ among $s\in\mathcal{S}_i$ with $x_{i,s}>0$. Let $\delta=c_{s_1}(x)-c_{s_0}(x)$ and our goal is to show that δ is small. The idea is that if δ were large, we could move a significant amount of load from s_1 to s_0 and decrease the Φ value substantially, which is impossible as $\Phi(x)$ is close to the minimum value $\Phi(q)$. Formally, let us move some Δ amount of load from s_1 to s_0, and let z denote the new flow. Note that the cost increase on s_0 and the cost decrease on s_1 are both at most $mb_1\Delta$, since $c_e'(y)\leq b_1$ for any y according to the condition (2). Thus, with $\Delta=\delta/(4b_1m)$, we can have $c_{s_1}(z)\geq c_{s_1}(x)-\delta/4$ and $c_{s_0}(z)\leq c_{s_0}(x)+\delta/4$, so that

$$c_{s_1}(z)-c_{s_0}(z)\geq c_{s_1}(x)-c_{s_0}(x)-\delta/2=\delta/2. \qquad (11)$$

On the other hand, moving the load decreases the Φ value by the amount

$$\begin{aligned}&\Phi(x)-\Phi(z)\\ &=\sum_{e\in s_1\setminus s_0}\int_{\ell_e(x)-\Delta}^{\ell_e(x)}c_e(y)dy-\sum_{e\in s_0\setminus s_1}\int_{\ell_e(x)}^{\ell_e(x)+\Delta}c_e(y)dy\\ &\geq\sum_{e\in s_1\setminus s_0}\Delta c_e(\ell_e(x)-\Delta)-\sum_{e\in s_0\setminus s_1}\Delta c_e(\ell_e(x)+\Delta)\\ &=\Delta\sum_{e\in s_1}c_e(\ell_e(z))-\Delta\sum_{e\in s_0}c_e(\ell_e(z))\\ &=\Delta\left(c_{s_1}(z)-c_{s_0}(z)\right)\\ &\geq\Delta\delta/2,\end{aligned}$$

where the first inequality holds as the function c_e is non-decreasing and the last inequality holds by (11). Since z is still a feasible flow in $\mathcal{K}$, its Φ value cannot be smaller than that of q and we must have $\Phi(x)-\Phi(z)\leq\Phi(x)-\Phi(q)\leq\varepsilon$, which implies that $\Delta\delta/2\leq\varepsilon$. With $\Delta=\delta/(4b_1m)$, we have $\delta\leq\sqrt{8b_1m\varepsilon}$. Since this holds for any $i\in N$, we have the theorem. ☐

4.2 Average Individual Cost

We show that after the convergence time, the average individual cost achieved by our algorithm is only within a constant factor from the optimum one.

THEOREM 10. *Any $x\in\mathcal{K}$ such that $\Phi(x)\leq\Phi(q)+\varepsilon$ must have $\frac{C_A(x)}{C_A(x^*)}\leq\frac{b_1}{b_0}\left(1+\frac{2m\varepsilon}{b_0}\right)$.*

PROOF. For any $z\in\mathcal{K}$, we can rewrite $C_A(z)$ as

$$\begin{aligned}C_A(z)&=\sum_e\ell_e(z)c_e(\ell_e(z))\\ &=\sum_e\int_0^{\ell_e(z)}(yc_e(y))'\,dy\\ &=\sum_e\int_0^{\ell_e(z)}\left(c_e(y)+yc_e'(y)\right)dy.\end{aligned}$$

Under the condition (2), we have $yc_e'(y)\leq yb_1=\frac{b_1}{b_0}b_0y\leq$

$\frac{b_1}{b_0} c_e(y)$ and thus

$$C_A(z) \leq \sum_e \int_0^{\ell_e(z)} \left(1 + \frac{b_1}{b_0}\right) c_e(y) dy = \frac{b_0 + b_1}{b_0} \Phi(z). \quad (12)$$

On the other hand, we also have $y c'_e(y) \geq y b_0 = \frac{b_0}{b_1} b_1 y \geq \frac{b_0}{b_1} c_e(y)$ and thus

$$C_A(z) \geq \sum_e \int_0^{\ell_e(z)} \left(1 + \frac{b_0}{b_1}\right) c_e(y) dy = \frac{b_0 + b_1}{b_1} \Phi(z). \quad (13)$$

Replacing z in (12) by x with $\Phi(x) \leq \Phi(q) + \varepsilon$, and replacing z in (13) by x^*, we obtain

$$\frac{C_A(x)}{C_A(x^*)} \leq \frac{b_1}{b_0} \frac{\Phi(x)}{\Phi(x^*)} \leq \frac{b_1}{b_0} \frac{\Phi(q) + \varepsilon}{\Phi(q)},$$

as $\Phi(x^*) \geq \Phi(q)$, which gives us

$$\frac{C_A(x)}{C_A(x^*)} \leq \frac{b_1}{b_0} \left(1 + \frac{\varepsilon}{\Phi(q)}\right). \quad (14)$$

Finally, using the condition (2), we have for any $z \in \mathcal{K}$ that

$$\Phi(z) \geq \sum_e \int_0^{\ell_e(z)} b_0 y dy = \frac{b_0}{2} \sum_e (\ell_e(z))^2 \geq \frac{b_0}{2m} \left(\sum_e \ell_e(z)\right)^2 \geq \frac{b_0}{2m}, \quad (15)$$

where the second inequality is by Cauchy-Schwarz and the last inequality holds as the total load of players is 1. Substituting the bound of (15) into (14) with $z = q$, we have the theorem. □

Remark 2. *We can make* $\frac{C_A(x)}{C_A(x^*)} \leq \frac{b_1}{b_0}(1 + \sigma)$ *for any* σ *we want, by choosing* $\varepsilon = b_0 \sigma / (2m)$*. Then by Remark 1, one can compute the corresponding convergence time* T_ε*, which is proportional to* $1/\sigma$*.*

4.3 Maximum Individual Cost in Symmetric Games

In a symmetric game, $\mathcal{S}_i = \mathcal{S}$ for every $i \in N$. Taking advantage of this property, we show that after the convergence time the maximum individual cost achieved by our algorithm is again within a constant factor from the optimum one.

Theorem 11. *Any* $x \in \mathcal{K}$ *such that* $\Phi(x) \leq \Phi(q) + \varepsilon$ *must have* $\frac{C_M(x)}{C_M(\hat{x})} \leq \frac{b_1}{b_0}\left(1 + \frac{2m\varepsilon}{b_0} + \frac{\delta m}{b_1}\right)$*, where* $\delta \leq \sqrt{8 b_1 m \varepsilon}$*.*

Proof. Consider any $x \in \mathcal{K}$ with $\Phi(x) \leq \Phi(q) + \varepsilon$. Let $s_0 = \arg\min_{s \in \mathcal{S}} c_s(x)$ and $s_1 = \arg\max_{s \in \mathcal{S}} c_s(x)$. To apply Theorem 9, let us choose a player i with $x_{i,s_1} > 0$; such a player must exist because otherwise there would be no load on s_1 and $c_{s_1}(x) = 0$ could not be the highest path cost. Since $\mathcal{S}_i = \mathcal{S}$ in a symmetric game, s_0 is also the path of player i with the lowest path cost. Therefore, we can apply Theorem 9 and have $\delta = c_{s_1}(x) - c_{s_0}(x) \leq \sqrt{8 b_1 m \varepsilon}$. Note that $C_M(x) = c_{s_1}(x)$ by definition. Thus, we have

$$\frac{C_M(x)}{C_M(\hat{x})} \leq \frac{c_{s_1}(x)}{C_A(\hat{x})} = \frac{c_{s_0}(x) + \delta}{C_A(\hat{x})} \leq \frac{C_A(x) + \delta}{C_A(x^*)},$$

where the first inequality is by the definitions of C_M and C_A, and the second inequality follows from the fact that $c_{s_0}(x) \leq C_A(x)$ and x^* minimizes C_A. Furthermore,

$$\frac{C_A(x) + \delta}{C_A(x^*)} = \frac{C_A(x)}{C_A(x^*)} + \frac{\delta}{C_A(x^*)} \leq \frac{b_1}{b_0}\left(1 + \frac{2\varepsilon m}{b_0}\right) + \frac{\delta}{C_A(x^*)}$$

by Theorem 10. Finally, using a similar analysis as in the proof of Theorem 10, one can show that

$$C_A(x^*) \geq \sum_e \int_0^{\ell_e(x^*)} (b_0 y + b_0 y) dy = b_0 \sum_e (\ell_e(x^*))^2 \geq \frac{b_0}{m}.$$

Combining all the bounds together, we have the theorem. □

Remark 3. *We can make* $\frac{C_M(x)}{C_M(\hat{x})} \leq \frac{b_1}{b_0}(1 + \sigma)$ *for any* σ *we want, by choosing* $\varepsilon = b_0 \sigma^2 / (32m)$*. Then according to Remark 1, one can compute the corresponding convergence time* T_ε*, which is now proportional to* $1/\sigma^2$*.*

5. CONCLUSIONS AND FUTURE WORK

We show that the mirror-descent dynamics converges to an approximate equilibrium in nonatomic congestion games. We do this by observing that the dynamics corresponds to a mirror-descent process on a convex potential function of such a game and then proving that the process converges to the minimum of the function. Moreover, we provide bounds on the outcome quality achieved by our dynamics in terms of two social costs: the average individual cost and the maximum individual cost.

A possible immediate extension is to consider the bandit setting [2, 12, 1], an even more stringent partial information model, in which a player in each round only gets to observe one single value: the cost of her strategy she just played. We are looking for good estimates for the gradient vectors to adapt mirror descent to work also in the bandit setting. However, it is not clear if any bandit mirror-descent dynamics would be able to converge, let alone the convergence time, since one can only have an estimator of the true gradient vector, and the estimators used by previous works all differ from the gradient significantly with high probability, although its expectation equals the gradient.

Finally, there may be other no-regret or even other learning algorithms which could guarantee nice convergence properties or simply good quality of outcomes. For example, convergence may not lead to any meaningful notions of equilibria, but may result in good efficiency in terms of some objectives [4]; natural learning processes have the potential to significantly outperform equilibrium-based analysis in some games [13]. There are more learning algorithms and dynamics to be explored in repeated games, while classes of games are even more numerous. Beyond learning, there is still a variety of different dynamics in repeated games. For instance, Auletta et al. [3] presented general bounds on the mixing time of “logit” dynamics for classes of strategic games, in which individual participants act selfishly and keep responding according to some partial noisy knowledge. In particular, they proved nearly tight bounds for potential games and games with dominant strategies. Different classes of games could have different choices of learning algorithms for better fine-tuned results.

6. REFERENCES

[1] J. Abernethy, E. Hazan, and A. Rakhlin. Competing in the dark: An efficient algorithm for bandit linear optimization. In *Proc. 21st Annual Conference on Learning Theory*, 2008.

[2] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1), 2003.

[3] V. Auletta, D. Ferraioli, F. Pasquale, and P. Penna. Convergence to equilibrium of logit dynamics for strategy games. In *Proc. 23rd ACM symposium on Parallelism in algorithms and architectures*, 2011.

[4] M.-F. Balcan, A. Blum, and Y. Mansour. Circumventing the price of anarchy: Leading dynamics to good behavior. *SIAM Journal on Computing*, 42(1), 2013.

[5] Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

[6] B. Birnbaum, N. Devanur, and L. Xiao. Distributed algorithms via gradient descent for fisher markets. In *Proc. 12th ACM Conference on Electronic Commerce*, pages 127–136, 2011.

[7] A. Blum, E. Even-Dar, and K. Ligett. Routing without regret: On convergence to Nash equilibria of regret-minimizing algorithms in routing games. In *Proc. 25th ACM Symposium on Principles of Distributed Computing*, 2006.

[8] A. Blum, M. T. Hajiaghayi, K. Ligett, and A. Roth. Regret minimization and the price of total anarchy. In *Proc. 40th Annual ACM Symposium on Theory of Computing*, 2008.

[9] N. Cesa-Bianchi and G. Lugosi, editors. *Prediction, Learning, and Games.* Cambridge University Press, 2006.

[10] Y K. Cheung, R. Cole, and N. Devanur. Tatonnement beyond gross substitutes?: gradient descent to the rescue. In *Proc. 45th ACM Symposium on theory of computing*, pages 191–200, 2013.

[11] E. Even-dar, Y. Mansour, and U. Nadav. On the convergence of regret minimization dynamics in concave games. In *Proc. 41st Annual ACM Symposium on Theory of Computing*, 2009.

[12] A. Flaxman, A. T. Kalai, and H. B. McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. In *Proc. 16th ACM-SIAM Symposium on Discrete Algorithms*, 2005.

[13] R. Kleinberg, K. Ligett, G. Piliouras, and E. Tardos. Beyond the nash equilibrium barrier. In *Proc. 2nd Symposium on Innovations in Computer Science*, 2011.

[14] R. Kleinberg, G. Piliouras, and E. Tardos. Load balancing without regret in the bulletin board model. In *Proc. 28th ACM Symposium on Principles of Distributed Computing*, 2009.

[15] R. Kleinberg, G. Piliouras, and E. Tardos. Multiplicative updates outperform generic no-regret learning in congestion games. In *Proc. 40th ACM Symposium on Theory of Computing*, 2009.

[16] T. Roughgarden. Intrinsic robustness of the price of anarchy. In *Proc. 41st Annual ACM Symposium on Theory of Computing*, 2009.

[17] T. Roughgarden and F. Schoppmann. Local smoothness and the price of anarchy in atomic splittable congestion games. In *Proc. 22nd ACM-SIAM Symposium on Discrete Algorithms*, 2011.

APPENDIX

A. PROOF OF PROPOSITION 1

Recall that

$$\Phi(x) = \sum_{e \in E} \int_0^{\ell_e(x)} y dy,$$

where $\ell_e(x) = \sum_{i \in N} \sum_{s: e \in s} x_{i,s}$. Let

$$\psi_e(v) = \int_0^v c_e(y) dy$$

so that $\Phi(x) = \sum_{e \in E} \psi_e(\ell_e(x))$. Observe that ℓ_e is a linear function of $x \in \mathcal{K}$, while ψ_e is a convex function of $v \in \mathbb{R}$ as c_e is assumed to be nondecreasing. Then for any $\delta \in [0,1]$ and any $x, x' \in \mathcal{K}$,

$$\begin{aligned} &(1-\delta)\Phi(x) + \delta\Phi(x') \\ &\quad = \sum_{e \in E} \big((1-\delta)\psi_e(\ell_e(x)) + \delta\psi_e(\ell_e(x'))\big) \\ &\quad \geq \sum_{e \in E} \psi_e((1-\delta)\ell_e(x) + \delta\ell_e(x')) \\ &\quad = \sum_{e \in E} \psi_e(\ell_e((1-\delta)x + \delta x')) \\ &\quad = \Phi((1-\delta)x + \delta x'). \end{aligned}$$

This proves that Φ is convex.

B. PROOF OF PROPOSITION 2

Let $\xi \in \mathcal{K}$. Consider any $i, j \in N$, $s \in \mathcal{S}_i$, and $r \in \mathcal{S}_j$. First, we have

$$\frac{\partial \Phi(\xi)}{\partial x_{i,s}} = \sum_{e \in E} c_e(\ell_e(\xi)) \frac{\partial \ell_e(\xi)}{\partial x_{i,s}} = \sum_{e \in s} c_e(\ell_e(\xi)).$$

Next, note that if $i \neq j$, we have

$$\frac{\partial^2 \Phi(\xi)}{\partial x_{i,s} \partial x_{j,r}} = 0,$$

and if $i = j$, we have

$$\frac{\partial^2 \Phi(\xi)}{\partial x_{i,s} \partial x_{i,r}} = \sum_{e \in s} \frac{\partial c_e(\ell_e(\xi))}{\partial x_{i,r}} = \sum_{e \in s \cap r} c'_e(\ell_e(\xi)).$$

This means that each entry of the Hessian matrix $\nabla^2 \Phi(\xi)$ is at most mb_1. Then for any $z \in \mathbb{R}^d$, we have

$$\begin{aligned} z^\top (\nabla^2 \Phi(\xi)) z &\leq mb_1 \sum_{(i,s),(j,r)} |z_{i,s}||z_{j,r}| \\ &= mb_1 \left(\sum_{i,s} |z_{i,s}| \right)^2 \\ &\leq mb_1 \left(\sqrt{d} \|z\|_2 \right)^2, \end{aligned}$$

by Cauchy-Schwarz inequality. This implies that $\nabla^2 \Phi(\xi) \preceq \alpha I$ with $\alpha = mb_1 d$.