Dynamic Network Congestion Games

Nathalie Bertrand
Univ Rennes, Inria, CNRS, IRISA, Rennes, France
nathalie.bertrand@inria.fr

Nicolas Markey
Univ Rennes, Inria, CNRS, IRISA, Rennes, France
nicolas.markey@irisa.fr

Suman Sadhukhan
Univ Rennes, Inria, CNRS, IRISA, Rennes, France
suman.sadhukhan@inria.fr

Ocan Sankur
Univ Rennes, Inria, CNRS, IRISA, Rennes, France
ocan.sankur@irisa.fr

Abstract

Congestion games are a classical type of games studied in game theory, in which n players choose a resource, and their individual cost increases with the number of other players choosing the same resource. In network congestion games (NCGs), the resources correspond to simple paths in a graph, *e.g.* representing routing options from a source to a target. In this paper, we introduce a variant of NCGs, referred to as *dynamic NCGs*: in this setting, players take transitions synchronously, they select their next transitions dynamically, and they are charged a cost that depends on the number of players simultaneously using the same transition.

We study, from a complexity perspective, standard concepts of game theory in dynamic NCGs: social optima, Nash equilibria, and subgame perfect equilibria. Our contributions are the following: the existence of a strategy profile with social cost bounded by a constant is in PSPACE and NP-hard. (Pure) Nash equilibria always exist in dynamic NCGs; the existence of a Nash equilibrium with bounded cost can be decided in EXPSPACE, and computing a witnessing strategy profile can be done in doubly-exponential time. The existence of a subgame perfect equilibrium with bounded cost can be decided in 2EXPSPACE, and a witnessing strategy profile can be computed in triply-exponential time.

2012 ACM Subject Classification Theory of computation → Algorithmic game theory; Theory of computation → Formal languages and automata theory; Theory of computation → Solution concepts in game theory; Theory of computation → Verification by model checking

Keywords and phrases Congestion games, Nash equilibria, Subgame perfect equilibria, Complexity

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2020.40

Related Version `https://arxiv.org/abs/2009.13632`.

1 Introduction

Congestion games model selfish resource sharing among several players [19]. A special case is the one of network congestion games, in which players aim at routing traffic through a congested network. Their popularity is certainly due to the fact that they have important practical applications, whether in transportation networks, or in large communication networks [18]. In network congestion games, each player chooses a set of transitions, forming a simple path from a source state to a target state, and the cost of a transition increases with its load, that is, with the number of players using it.

Network congestion games can be classified into atomic and non-atomic variants. Non-atomic semantics is appropriate for large populations of players, thus seen as a continuum. One then considers portions of the population that apply predefined strategies, and there

40th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2020).
Editors: Nitin Saxena and Sunil Simon; Article No. 40; pp. 40:1–40:16
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

is no such thing as the effect of an individual player on the cost of others. In contrast, in atomic games, the number of players is fixed, and each player may significantly impact the cost other players incur. We only focus on atomic games in this paper.

Network congestion games. Network congestion games, also called atomic selfish routing games in the literature, were first considered by Rosenthal [19]. These games are defined by a directed graph, a number of pairs of source-target vertices, and non-decreasing cost functions for each edge in the graph. For each source-target pair, a player must choose a route from the source to the target vertex. Given their choice of simple paths, the cost incurred by a player depends on the number of other players that choose paths sharing edges with their path, and on the cost functions of these edges. In this setting, a Nash equilibrium maps each player to a path in such a way that no player has an incentive to deviate: they cannot decrease their cost by choosing a different path.

Rosenthal proved that they are potential games, so that Nash equilibria always exist. Monderer and Shapley [16] studied in a more general way potential games, and explained how to iteratively use best-response strategies to converge to an equilibrium. Interestingly, under reasonable assumptions on the cost functions, Bertsekas and Tsitsiklis established that there is a direct correspondence between equilibria in selfish routing and distributed shortest-path routing, which is used in practice for packet routing in computer networks [7]. We refer the interested reader to [20] for an introduction and many basic results on general routing games.

A natural question is whether selfish routing is very different from a routing strategy decided by a centralized authority. In other words, how far can a selfish optimum be from the social optimum, in which players would cooperate. The notion of price of anarchy, first proposed by Koutsoupias and Papadimitriou [13], is the ratio of the worst cost of a Nash equilibrium and the cost of the social optimum. This measures how bad Nash equilibria can be. In the context of network congestion games, the price of anarchy was first studied by Suri *et al.* [21], establishing an upper bound of $\frac{5}{2}$ when all cost functions are affine. A refined upper bound was provided by Awerbuch *et al.* [5]. Bounds on the dual notion of price of stability, which is the ratio of the cost of a best Nash equilibrium and the cost of the social optimum was also studied for routing games [1].

Timing aspects. Several works investigated refinements of this setting. In [10], the authors study network congestion games in which each edge is traversed with a fixed duration independent of its load, while the cost of each edge depends on the load. The model is thus said to have *time-dependent* costs since the load depends on the times at which players traverse a given edge. The authors prove the existence of Nash equilibria by reduction to the setting of [19]. An extension of this setting with timed constraints was studied in [2, 3].

The setting of fixed durations with time-dependent costs is interesting in applications where the players sharing a resource (an edge) see their quality of service decrease, while the time to use the resource is unaffected [3]. This might be the case, for instance, in some telecommunication and multimedia streaming applications. Timing also appears, for instance, in [17, 14] where the load affects travel times and players' objective is to minimize the total travel time. Other works focus on flow models with a timing aspect [12, 8].

Dynamic network congestion games. In classical network congestion games, including those mentioned above, players choose their strategies (*i.e.,* their *simple* paths) in one shot. However, it may be interesting to let agents choose their paths *dynamically*, that is, step by step, by observing other players' previous choices. In this paper, we study network congestion games with time-dependent costs as in [10], but with unit delays, and in a dynamic setting. More precisely, at each step, each of the players simultaneously selects the edge they want to take; each player is then charged a cost that depends on the load of the edge they selected, and traverses that edge in one step. We name these games *dynamic network congestion games* (dynamic NCGs in short); the behaviour of the players in such games is formalized by means of *strategies*, telling the players what to play depending of the current configuration. Notice that, because congestion effect applies to edges used *simultaneously* by several players, taking cycles can be interesting in dynamic NCGs, which makes our setting more complex than most NCG models [4, 10, 19, 20].

Such a dynamic setting was studied in [4] for resource allocation games, which extends [19] with dynamic choices. A more detailed related work appears at the end of this section.

Standard solution concepts. We study classical solution concepts on dynamic network congestion games. A strategy profile (*i.e.*, a function assigning a strategy to each player) is a *Nash Equilibrium* (NE) when each single strategy is an optimal response to the strategies of the other players; in other terms, under such a strategy profile, no player may lower their costs by unilaterally changing their strategies. Notice that NEs need not exist in general, and when they exist, they may not be unique. In the setting of dynamic games, Nash Equilibria are usually enforced using *punishing strategies*, by which any deviating player will be punished by all other players once the deviation has been detected. However, such punishing strategies may also increase the cost incurred to the punishing players, and hence do not form a credible threat; *Subgame-Perfect Equilibria* (SPEs) refine NEs and address this issue by requiring that the strategy profile is an NE along any play.

NEs and SPEs aim at minimizing the individual cost of each player (without caring of the others' costs); in a collaborative setting, the players may instead try to lower the social cost, i.e., the sum of the costs incurred to all the players. Strategy profiles achieving this are called *social optima* (SO). Obviously, the social cost of NEs and SPEs cannot be less than that of the social optimum; the *price of anarchy* measures how bad selfish behaviours may be compared to collaborative ones.

Our contributions. We take a computational-complexity viewpoint to study dynamic network congestion games. We first establish the complexity of computing the social optimum, which we show is in PSPACE and NP-hard. We then prove that best-response strategies can be computed in polynomial time, and that dynamic NCGs are potential games, thereby showing the existence of Nash equilibria in any dynamic NCG; this also shows that some Nash equilibrium can be computed in pseudo-polynomial time. We then give an EXPSPACE (resp. 2EXPSPACE) algorithm to decide the existence of Nash Equilibria (resp. Subgame-Perfect Equilibria) whose costs satisfy given bounds. This allows us to compute best and worst such equilibria, and then the price of anarchy and the price of stability.

Note that some of the high complexities follow from the binary encoding of the number of players, which is the main input parameter. For instance, the exponential-space complexity drops to pseudo-polynomial time for a fixed number of players. This parameter becomes important since we advocate the study of computational problems, such as computing Nash equilibria with a given cost bound. We also believe that computing precise values for price of anarchy and the price of stability is interesting, rather than providing bounds on the set of all instances as in *e.g.* [21].

Omitted proofs can be found in the corresponding arXiv article [6].

Comparison with related work. The works closest to our setting are [10, 4, 2, 3]. As in [10, 3], we establish the existence of Nash equilibria using potential games. Unlike [10], we cannot obtain this result immediately by reducing our games to congestion games [19] since the lengths of the strategies cannot be bounded *a priori*. Moreover, the best-response problem has a polynomial-time solution in our setting while the problem is NP-hard both in [10, 3]. In [10], this is due to the possibility of having arbitrary durations, while the source of complexity in [2, 3] is due to the use of timed automata. Thus, our setting offers a simpler way of expressing timings, and avoids their high complexity for this problem.

Dynamic choices were studied in [4] but with a different cost model. Moreover, network congestion games can only be reduced to such a setting given an a priori bound on the length of the paths. So we cannot directly transfer any of their results to our setting. Dynamic choices were also studied in [10] in the setting of coordination mechanisms which are local policies that allow one to sequentialize traffic on the edges.

2 Preliminaries

2.1 Dynamic network congestion games

Let $\mathcal{F}$ be the family of non-decreasing functions from $\mathbb{N}$ to $\mathbb{N}$ that are piecewise-affine, with finitely many pieces. We assume that each $\mathsf{f} \in \mathcal{F}$ is represented by the endpoints of intervals, and the coefficients, all encoded in binary. An arena for dynamic network congestion games is a weighted graph $\mathcal{A} = \langle V, E, \mathsf{src}, \mathsf{tgt}\rangle$, where V is a finite set of states, $E\colon V \times V \to \mathcal{F}$ is a partial function defining the cost of edges, and src and tgt are a source- and a target state in V. It is assumed throughout this paper that tgt has only a single outgoing transition, which is a self-loop with constant cost function $x \mapsto 0$. We also assume that tgt is reachable from all other states.

A dynamic network congestion game (dynamic NCG for short) is a pair $\mathcal{G} = \langle \mathcal{A}, n\rangle$ where $\mathcal{A}$ is an arena as above and $n \in \mathbb{N}$ is the (binary-encoded) number of players. In a dynamic network congestion game, all players start from src and simultaneously select the edges they want to traverse, with the aim of reaching the target state with minimal individual accumulated cost. Taking an edge $e = (v, \mathsf{f}, v')$ has a cost $\mathsf{f}(l)$, where l is the number of players simultaneously using edge e. The cost function of edge e is denoted by f_e. We let $\kappa = \max_{e\in E} \mathsf{f}_e(n)$, which is the maximal cost that a player may endure along one edge.

Our setting differs from classical network congestion games [20] mainly in two respects:

- first, the game is played in rounds, during which all players take exactly one transition; the number of players using an edge e is measured *dynamically*, at each round;
- second, during the play, players may adapt their choices to what the other players have been doing in the previous rounds.

▶ Remark 1. In this work, we mainly focus on the *symmetric* case, where all players have the same source and target. This is because we take a parametric-verification point of view, with

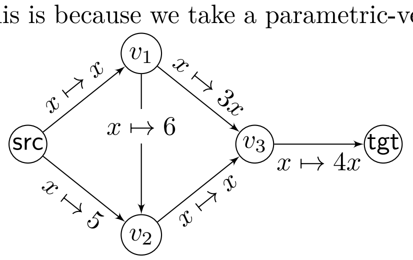

Figure 1 Representation of an arena for a dynamic NCG (loop omitted on tgt).

the (long-term) aim of checking properties of dynamic NCGs for arbitrarily many players. An important consequence of this choice is that the number of players now is encoded in binary, which results in an exponential blow-up in the number of configurations of the game (compared to the asymmetric setting).

Semantics as a concurrent game. For any $k \in \mathbb{N}$, we write $[\![k]\!] = \{i \in \mathbb{N} \mid 1 \leq i \leq k\}$. A *configuration* of a dynamic network congestion game $\langle \mathcal{A}, n\rangle$ is a mapping $c\colon [\![n]\!] \to V$, indicating the position of each player in the arena. We define $c_{\mathsf{src}}\colon i \in [\![n]\!] \mapsto \mathsf{src}$ and $c_{\mathsf{tgt}}\colon i \in [\![n]\!] \mapsto \mathsf{tgt}$ as the initial and target configurations, respectively.

With $\langle \mathcal{A}, n\rangle$, we first associate a multi-weighted graph $\mathcal{M} = \langle C, T\rangle$, where $C = V^{[\![n]\!]}$ is the set of all configurations and $T \subseteq C \times \mathbb{N}^{[\![n]\!]} \times C$ is a set of edges, defined as follows: there is an edge (c, w, c') in T if, and only if, there exists a collection $\mathbf{e} = (e_i)_{i\in[\![n]\!]}$ of edges of E such that for all $i \in [\![n]\!]$, writing $e_i = (v_i, \mathsf{f}_i, v'_i)$ and $u_i = \#\{j \in [\![n]\!] \mid e_j = e_i\}$, we have $c(i) = v_i$, $c'(i) = v'_i$, and $w(i) = \mathsf{f}_i(u_i)$. We denote this edge with $c \stackrel{\mathbf{e}}{\Rightarrow} c'$. We may omit to mention $\mathbf{e}$ since it can be obtained from c and c'; similarly, we write $\mathsf{cost}_i(c, c')$ for $w(i)$.

Two edges (c, w, c') and (d, x, d'), in that order, are said to be *consecutive* whenever $d = c'$. Given a configuration c, a *path* from c in a dynamic network congestion game is either the single configuration c (we call this a trivial path) or a non-empty, finite or infinite sequence of consecutive edges $\rho = (t_j)_{1\leq j<|\rho|}$ in $\mathcal{M}$, where t_1 is a transition from c; the size of a path ρ is one for trivial paths, and $|\rho| \in \mathbb{N} \cup \{+\infty\}$ otherwise. We write $\mathsf{Paths}(\langle \mathcal{A}, n\rangle, c)$ and $\mathsf{Paths}^\omega(\langle \mathcal{A}, n\rangle, c)$ for the set of finite and infinite paths from c in $\langle \mathcal{A}, n\rangle$, respectively.

With each path $\rho = (c_j, w_j, c'_j)_j$, and each player $i \in [\![n]\!]$, we associate a *cost*, written $\mathsf{cost}_i(\rho)$, which is zero for trivial paths, $+\infty$ for infinite paths along which $c_j(i) \neq \mathsf{tgt}$ for all j, and $\sum_{j=1}^{|\rho|-1} w_j(i)$ otherwise. We define the *social cost* of ρ, denoted by $\mathsf{soccost}(\rho)$, as $\sum_{i\in[\![n]\!]} \mathsf{cost}_i(\rho)$.

Given a path ρ, an index $1 \leq j < |\rho| + 1$ and a player $i \in [\![n]\!]$, we write $\rho(j)$ for the j-th configuration of ρ, and $\rho(j)(i)$ for the state of Player i in that configuration. For $j \geq 2$, we define $\rho_{\leq j}$ as the prefix of ρ that ends in the j-th configuration; we let $\rho_{\leq 1} = \rho(1)$. Similarly, for $1 \leq j \leq |\rho| - 1$, we let $\rho_{\geq j}$ denote the suffix that starts at the j-th configuration. Finally, if $|\rho|$ is finite, we let $\rho_{\geq |\rho|} = \rho(|\rho|)$.

▶ **Example 2.** Consider the arena $\mathcal{A}$ displayed at Fig. 1 and the dynamic NCG $\langle \mathcal{A}, 2\rangle$ with two players. Assume that Player 1 follows the path $\pi_1\colon \mathsf{src} \to v_1 \to v_3 \to \mathsf{tgt}$ and Player 2 goes via $\pi_2\colon \mathsf{src} \to v_1 \to v_2 \to v_3 \to \mathsf{tgt}$. This gives rise to the following path:

$$\begin{pmatrix} 1 \mapsto \mathsf{src} \\ 2 \mapsto \mathsf{src} \end{pmatrix} \xrightarrow[2\mapsto 2]{1\mapsto 2} \begin{pmatrix} 1 \mapsto v_1 \\ 2 \mapsto v_1 \end{pmatrix} \xrightarrow[2\mapsto 6]{1\mapsto 3} \begin{pmatrix} 1 \mapsto v_3 \\ 2 \mapsto v_2 \end{pmatrix} \xrightarrow[2\mapsto 1]{1\mapsto 4} \begin{pmatrix} 1 \mapsto \mathsf{tgt} \\ 2 \mapsto v_3 \end{pmatrix} \xrightarrow[2\mapsto 4]{1\mapsto 0} \begin{pmatrix} 1 \mapsto \mathsf{tgt} \\ 2 \mapsto \mathsf{tgt} \end{pmatrix}$$

Notice how edge $v_3 \to \mathsf{tgt}$ of $\mathcal{A}$ is used by both players, but not simultaneously, so that the cost of using that edge is 4 for each of them, while it would be 8 in classical NCGs. ◀

We now extend this graph to a concurrent game structure. A *move* for Player $i \in [\![n]\!]$ from configuration c is an edge $e = (v, \mathsf{f}, v') \in E$ such that $v = c(i)$. A *move vector* from c is a sequence $\mathbf{e} = (e_i)_{i\in[\![n]\!]}$ such that for all $i \in [\![n]\!]$, e_i is a move for Player i from c.

A network congestion game $\langle \mathcal{A}, n\rangle$ then gives rise to a concurrent game structure $\mathcal{S} = \langle C, T, M, U\rangle$ where $\langle C, T\rangle$ is the graph defined above, $M\colon C \times [\![n]\!] \to 2^E$ lists the set of possible moves for each player in each configuration, and $U\colon C \times E^{[\![n]\!]} \to T$ is the transition function, such that for every configuration c and every move vector $\mathbf{e} = (e_i)_{i\in[\![n]\!]}$ with $e_i \in M(c, i)$ for all $i \in [\![n]\!]$, $U(c, e) = (c \stackrel{\mathbf{e}}{\Rightarrow} c')$.

A *strategy* for Player i in $\mathcal{S}$ from configuration c is a function $\sigma_i\colon \mathsf{Paths}(\langle \mathcal{A}, n\rangle, c) \to E$ that associates, with any finite path ρ from c in $\mathcal{S}$, a move for this player from the last configuration of ρ. A *strategy profile* is a family $\sigma = (\sigma_i)_{i\in[\![n]\!]}$ of strategies, one for each player. We write $\mathfrak{S}$ for the set of strategies, and $\mathfrak{S}^n$ for the set of strategy profiles.

Let c be a configuration, h be a finite path from c and a strategy profile $\sigma = (\sigma_i)_{i\in[\![n]\!]}$ from c. The *residual strategy profile* of σ after h is the strategy profile $\sigma^h = (\sigma_i^h)_{i\in[\![n]\!]}$ from the last configuration of h defined by $\sigma_i^h(h') = \sigma_i(h \cdot h')$, where $h \cdot h'$ is the concatenation of paths h and h'.

The *outcome* of a strategy profile σ from c is the infinite path $\rho = (c_i, w_i, c_{i+1})_{i\geq 1}$, hereafter denoted with $\mathsf{outcome}(\sigma)$, obtained by running the strategy profile; it is formally defined as the only infinite path such that $(c_1, w_1, c_2) = U(c, \sigma(c))$, and such that for any $j \geq 2$, $(c_j, w_j, c_{j+1}) = U(c_j, \sigma(h'))$, where $h' = (c_1, w_1, c_2) \cdots (c_{j-1}, w_{j-1}, c_j)$.

Pick a strategy profile $\sigma = (\sigma_i)_{i\in[\![n]\!]}$, and let $\rho = (t_j)_{j\geq 1}$ be its outcome, writing $t_j = (c_j, (w_j^i)_{i\in[\![n]\!]}, c'_j)$ for all $j \geq 1$. Let $k \in [\![n]\!]$. If $c'_l(k) = \mathsf{tgt}$ for some $l \in \mathbb{N}$, then σ_k is said to be winning for Player k. In that case, we define $\mathsf{cost}_k(\sigma)$ as $\mathsf{cost}_k(\mathsf{outcome}(\sigma))$. If $c'_l(i) = \mathsf{tgt}$ for all $i \in [\![n]\!]$, we define the *social cost* of σ as $\mathsf{soccost}(\sigma) = \mathsf{soccost}(\rho)$.

A strategy σ_i for Player i is said *blind* whenever for any two finite paths ρ and ρ' having same length k, if for any position $0 \leq j < k$ we have $\rho(j)(i) = \rho'(j)(i)$, then $\sigma_i(\rho) = \sigma_i(\rho')$. Intuitively, this means that strategy σ_i follows a path in $\mathcal{A}$, independently of what the other players do. A blind strategy can thus be represented as a path and we write $|\sigma_i|$ for the length of that path (until its first visit to tgt, if any). We write $\mathfrak{B}$ for the set of blind strategies.

▶ **Example 3.** Consider again the arena $\mathcal{A}$ of Fig. 1. The paths π_1 and π_2 from Example 2 are two blind strategies in that dynamic NCG. In a 2-player setting, an example of a non-blind strategy σ consists in first taking the transition $\mathsf{src} \to v_1$, and then either taking $v_1 \to v_3$ if the other player took the same initial transition, or taking $v_1 \to v_2$ otherwise. ◀

Representation as a weighted graph. Another way of representing configurations is to consider their Parikh images. With a configuration $c \in V^{[\![n]\!]}$, we associate an abstract configuration $\bar{c} \in [\![n]\!]^V$ defined as $\bar{c}(v) = \#\{i \in [\![n]\!] \mid c(i) = v\}$.

The *abstract weighted graph* associated with a dynamic network congestion game $\langle \mathcal{A}, n\rangle$ is the weighted graph $\mathcal{P} = \langle A, B\rangle$, where A contains all abstract configurations, and there is an edge (a, w, a') in $B \subseteq A \times \mathbb{N} \times A$ if, and only if, there is a mapping $b\colon E \to [\![n]\!]$ such that $\sum_{e\in E} b(e) = n$ and for all $v \in V$,

$$a(v) = \sum_{e=(v,\mathsf{f},v')} b(e) \qquad w = \sum_{e=(v,\mathsf{f},v')} b(e) \times \mathsf{f}(b(e)) \qquad a'(v) = \sum_{e=(v',\mathsf{f},v)} b(e).$$

Similarly to the representation as multi-weighted graphs, an *abstract path* of a network congestion game is either a single configuration or a non-empty, finite or infinite sequence of consecutive edges in the abstract weighted graph. The *cost* of an abstract path is the sum of the weights of its edges (if any). Then:

▶ **Lemma 4.** *For any $w \in \mathbb{N} \cup \{+\infty\}$, there is an abstract path in $\mathcal{M}$ with social cost w if, and only if, there is an abstract path in $\mathcal{P}$ with cost w.*

2.2 Social optima and equilibria

Consider a dynamic network congestion game $\mathcal{G} = \langle \mathcal{A}, n\rangle$. We recall standard ways of optimizing the strategies of the players, depending on the situation.

In a collaborative situation, all players want to collectively minimize the total cost for having all of them reach the target state of the arena. Formally, a strategy profile $\sigma = (\sigma_i)_{i\in[\![n]\!]}$ realizes the *social optimum* if $\mathsf{soccost}(\sigma) = \inf_{\tau\in\mathfrak{S}^n} \mathsf{soccost}(\tau)$.

In a selfish situation, each player aims at optimizing their response to the others' strategies. Given a strategy profile $\sigma = (\sigma_i)_{i\in[\![n]\!]}$, a player $k \in [\![n]\!]$, and a strategy $\sigma'_k \in \mathfrak{S}$, we denote by $\langle\sigma_{-k}, \sigma'_k\rangle$ the strategy profile $(\tau_i)_{i\in[\![n]\!]}$ such that $\tau_k = \sigma'_k$ and $\tau_i = \sigma_i$ for all $i \in [\![n]\!] \setminus \{k\}$. The strategy σ_k is a *best response* to $(\sigma_i)_{i\in[\![n]\!]\setminus\{k\}}$ if $\mathsf{cost}_k(\sigma) = \inf_{\sigma'_k\in\mathfrak{S}} \mathsf{cost}_k(\langle\sigma_{-k}, \sigma'_k\rangle)$. A strategy profile $\sigma = (\sigma_i)_{i\in[\![n]\!]}$ is a *Nash equilibrium* if for each $k \in [\![n]\!]$, the strategy σ_k is a best response to $(\sigma_i)_{i\in[\![n]\!]\setminus\{k\}}$. In such a case, no player has profitable unilateral deviations, *i.e.*, no player alone can decrease their cost by switching to a different strategy.

Nash equilibria can be defined for subclasses of strategy profiles. In particular, a *blind Nash equilibrium* is a blind strategy profile σ that is a Nash equilibrium *for blind-strategy deviations*: for all $k \in [\![n]\!]$, $\mathsf{cost}_k(\sigma) = \inf_{\sigma'_k\in\mathfrak{B}} \mathsf{cost}_k(\langle\sigma_{-k}, \sigma'_k\rangle)$. *A priori*, a blind Nash equilibrium need not be a Nash equilibrium for general strategies.

In an NE, once a player deviated from their original strategy in the strategy profile, the other players can punish the deviating player, even if this results in increasing their own costs. Indeed, the condition for being an NE only requires that the deviation should not be profitable to the deviating player. Subgame-Perfect Equilibria (SPE) refine NEs and rule out such non-credible threats by requiring that, for any path h in the configuration graph, the residual strategy profile after h is an NE.

▶ **Example 5.** Consider again the dynamic NCG $\langle\mathcal{A}, 2\rangle$, with the arena $\mathcal{A}$ of Fig. 1. Assume that Player 1 plays the blind strategy corresponding to $\pi_3\colon \mathsf{src} \to v_2 \to v_3 \to \mathsf{tgt}$, while Player 2 plays the non-blind strategy σ of Example 3. The cost for Player 1 then is 10, while that of Player 2 is 12.

This strategy profile is an NE: Player 2 could be tempted to play π_1, but they would then synchronize with Player 1 on edge $v_3 \to \mathsf{tgt}$, and get cost 12 again. Similarly, Player 1 could be tempted to play π_1 instead of π_3, but in that case strategy σ would tell Player 2 to follow the same path, and the cost for Player 1 (and 2) would be 16. Notice in particular that this is not an SPE, but that the blind strategy profile $\langle\pi_1, \pi_2\rangle$ (extended to the whole configuration tree in the only possible way) is an SPE in $\langle\mathcal{A}, 2\rangle$. ◀

In Sections 4 and 5, we focus on NEs and SPEs, developing EXPSPACE and 2EXPSPACE-algorithms for deciding the existence of NEs and SPEs respectively of social cost less than or equal to a given bound. Actually, our approach extends to the $\vec{\gamma}$-weighted social cost, where $\vec{\gamma} \in \mathbb{Z}^{[\![n]\!]}$ are coefficients applied to the costs of the respective players when computing the social cost. As a consequence, we can compute best and worst NEs and SPEs, hence also the price of anarchy and price of stability [13]. Before that, in Section 3, we extend classical techniques using blind strategies to compute the social optimum and prove that NEs always exist.

3 Socially-optimal strategy profiles

To compute a socially-optimal strategy profile, it suffices to find a path in the concurrent game structure of the given network congestion game with minimal total cost since one can define a strategy profile that induces any given path. Rather than finding such a path in the concurrent game structure, and in view of Lemma 4, one can look for one in the abstract weighted graph, thereby reducing in complexity. The socially-optimal cost in a dynamic NCG $\langle\mathcal{A}, n\rangle$ is thus the cost of a shortest path in the associated weighted abstract graph $\mathcal{P}$ from $\bar{c}_{\mathsf{src}}$ to $\bar{c}_{\mathsf{tgt}}$.

Since $\mathcal{P}$ has exponential size, we derive complexity upper bounds for computing a socially-optimal strategy and deciding the associated decision problem. Moreover, adapting [15, Theorem 4.1] which proves NP-hardness in classical NCGs, we provide a reduction from the Partition problem to establish an NP lower-bound.

▶ **Theorem 6.** *A socially-optimal strategy profile can be computed in exponential time. The constrained social-optimum problem is in PSPACE and NP-hard.*

Note, that while $\mathcal{P}$ has size $(n+1)^{|V|}$, it is sufficient to consider paths with a smaller number of transitions when looking for a shortest path:

▶ **Lemma 7.** *There is a shortest path (w.r.t. cost) in $\mathcal{P}$ with size (in terms of its number of transitions) at most $n \cdot |V|$.*

▶ Remark 8. A consequence of Lemma 7 is that deciding the constrained social-optimum problem is in NP for asymmetric games, since in that setting the lists of sources and targets of each player is part of the input, so that n is polynomial in the size of the input. However, our NP-hardness proof only works in the symmetric case.

4 Nash equilibria

In this section, we study the existence of Nash equilibria and give algorithms to compute them under given constraints.

4.1 Existence and computation of (blind) Nash equilibria

To prove that blind Nash equilibria always exist, we establish that dynamic NCGs with blind strategies are potential games [19, 16] which are known to have Nash equilibria.

Consider a dynamic NCG $\langle \mathcal{A}, n \rangle$, a blind strategy profile π, and let N_π denote the maximum length of the paths prescribed by π. We define the following potential function, which is an adaptation of that used in [19]:

$$\psi(\pi) = \sum_{j=1}^{N_\pi} \sum_{e \in E} \sum_{i=1}^{\mathsf{load}_e(\pi,j)} \mathsf{f}_e(i),$$

where $\mathsf{load}_e(\pi, j)$ denotes the number of players that take edge e in the j-step under π, and f_e is the cost function on edge e.

Using the above-defined potential function, one can derive an algorithm to find a Nash equilibrium, by a classical *best-response* iteration. Starting with an arbitrary blind strategy profile, at each step we replace some player's strategy with their best-response, and we continue as long as some player's cost can be decreased. When this procedure terminates, the profile at hand is a blind Nash equilibrium. In dynamic NCGs, best responses exist and can be computed in polynomial time. Indeed, one can construct a game in which all players but Player i follow their fixed strategies given by profile π, using N_π copies of the game in order to distinguish the steps. After the N_π-step, all players in $[\![n]\!] \setminus \{i\}$ have reached their targets. Since it is the only remaining player, the remaining path for Player i should not be longer than $|V|$. Altogether, we obtain the following complexity upper-bound:

▶ **Theorem 9.** *In dynamic NCGs, blind Nash equilibria always exist, and we can compute one in pseudo-polynomial time.*

▶ Remark 10. As an alternative proof to existence of blind NEs, we could have bounded the length of outcomes of blind NEs as follows: all players have a strategy realizing cost at most $|V| \cdot \kappa$, where $\kappa = \max_{e \in E} \mathsf{f}_e(n)$, since the shortest path from src to tgt has length at most $|V|$, and the cost for a player at each step along edge e is at most κ. It follows that no path along which the cost for some player is larger than $|V| \cdot \kappa$ can be the outcome of a blind NE. As a consequence, if a dynamic NCG has a blind NE, then it has one of length at most $|V| \cdot \kappa \cdot |V|^n$ (by removing zero-cycles). Using this bound, we can transform dynamic NCGs into classical congestion games, in which blind NEs always exist [10, 19].

We now show that blind Nash equilibria are in fact Nash equilibria. This is proved using the observation that given a blind strategy profile, the most profitable deviation for any player can be assumed to be a blind strategy.

▶ **Lemma 11.** *In dynamic NCGs, blind Nash equilibria are Nash equilibria.*

Computing some (blind) Nash equilibrium may not be satisfactory for two reasons: one might want to compute the best (or the worst) Nash equilibrium in terms of the social cost; and as Lemma 12 claims, blind Nash equilibria are suboptimal, *i.e.*, a lower social cost can be achieved by Nash equilibria with general strategies. This justifies the study of more complex strategy profiles in the next subsection.

▶ **Lemma 12.** *There exists a dynamic NCG with a Nash equilibrium π such that for all blind Nash equilibria π', we have $\mathsf{cost}(\pi) < \mathsf{cost}(\pi')$.*

The proof is based on the dynamic NCG depicted on Fig. 2, for which we prove there is a Nash equilibrium with total cost 36, while any *blind* Nash equilibrium has higher social cost.

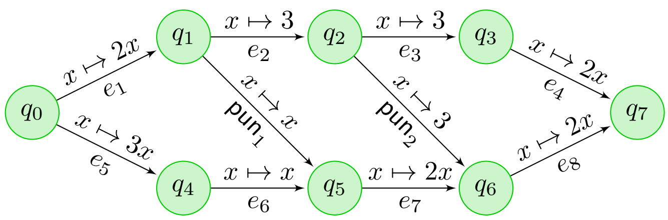

Figure 2 An arena on which blind Nash equilibria are sub-optimal.

4.2 Computation of general Nash equilibria

Characterization of outcomes of Nash Equilibria. Let us consider a dynamic NCG $\langle \mathcal{A}, n \rangle$, and the corresponding game structure $\mathcal{S} = \langle C, T, M, U \rangle$. Given two configurations c, c' with $c \Rightarrow c'$, we let $\mathsf{cost}_i(c, c')$ denote the cost of Player i on this transition from $c(i)$ to $c'(i)$. We define $\mathsf{dev}_i(c, c')$ as the set of all configurations reachable when all players but Player i choose moves prescribed by the given transition $c \Rightarrow c'$:

$$\mathsf{dev}_i(c, c') = \{c'' \in C \mid c \Rightarrow c'' \text{ and } \forall j \in [\![n]\!] \setminus \{i\}.\ c''(j) = c'(j)\}.$$

The *value* of configuration c for Player i is $\mathsf{val}_{i,c} = \sup_{\sigma_{-i} \in \mathfrak{S}^{n-1}} \inf_{\sigma_i \in \mathfrak{S}} \mathsf{cost}_i((\sigma_{-i}, \sigma_i), c)$. Note that the value corresponds to the value of the zero-sum game where Player i plays against the opposing coalition, starting at c. By [11], those values can be computed in polynomial time in the size of the game. Here the game is a 2-player game with state space $|V| \times [\![n-1]\!]^{|V|}$, keeping track of the position of Player i and the abstract position of the coalition. It follows that each $\mathsf{val}_{i,c}$ can be computed in exponential time in the size of the input $\langle \mathcal{A}, n\rangle$. Moreover, memoryless optimal strategies exist (in $\mathcal{S}$), that is, the opposing coalition has a memoryless strategy σ_{-i} to ensure a cost of at least $\mathsf{val}_{i,c}$ from c.

The characterization of Nash equilibria outcomes is given in the following lemma.

▶ **Lemma 13.** *A path ρ in $\langle \mathcal{A}, n\rangle$ is the outcome of a Nash equilibrium if, and only if,*

$$\forall i \in [\![n]\!].\ \forall 1 \leq l < |\rho|.\ \forall c \in \mathit{dev}_i(\rho(l), \rho(l+1)). \quad \mathit{cost}_i(\rho_{\geq l}) \leq \mathit{val}_{i,c} + \mathit{cost}_i(\rho(l), c).$$

The intuition is that if the suffix $\mathsf{cost}_i(\rho_{\geq l})$ of ρ has cost more than $\mathsf{val}_{i,c} + \mathsf{cost}_i(\rho(l), c)$, then Player i has a profitable deviation regardless of the strategy of the opposing coalition, since $\mathsf{val}_{i,c}$ is the maximum cost that the coalition can inflict to Player i at configuration c where the deviation is observed. The lemma shows that the absence of such a suffix means that a Nash equilibrium with given outcome exists, which the proof constructs.

Proof. Consider a Nash equilibrium $\sigma = (\sigma_i)_{i\in[\![n]\!]}$ with outcome ρ. Consider any player i, and any strategy σ'_i for this player. Let ρ' denote the outcome of $\sigma[i \to \sigma'_i]$. Let l denote the index of the last configuration where ρ and ρ' are identical. Since σ is a Nash equilibrium, we have $\mathsf{cost}_i(\rho) \leq \mathsf{cost}_i(\rho')$, that is,

$$\mathsf{cost}_i(\rho_{\geq l}) \leq \mathsf{cost}_i(\rho(l), \rho'(l+1)) + \mathsf{cost}_i(\sigma[i \to \sigma'_i], \rho'_{\leq l+1})$$

where $\mathsf{cost}_i(\sigma[i \to \sigma'_i], \rho'_{\leq l+1})$ is the cost for Player i of the outcome of the residual strategy $(\sigma[i \to \sigma'_i])^{\rho'_{\leq l+1}}$. Since the choice of σ'_i is arbitrary here, we have,

$$\mathsf{cost}_i(\rho_{\geq l}) \leq \mathsf{cost}_i(\rho(l), \rho'(l+1)) + \inf_{\sigma'_i\in\mathfrak{S}} \mathsf{cost}_i(\sigma[i \to \sigma'_i], \rho'_{\leq l+1}).$$

Moreover, we have $\inf_{\sigma'_i\in\mathfrak{S}} \mathsf{cost}_i(\pi[i \to \sigma'_i], \rho'_{\leq l+1}) = \inf_{\sigma'_i\in\mathfrak{S}} \mathsf{cost}_i(\pi[i \to \sigma'_i], \rho'(l+1))$ since memoryless strategies suffice to minimize the cost [11]. We then have

$$\inf_{\sigma'_i\in\mathfrak{S}} \mathsf{cost}_i(\pi[i \to \sigma'_i], \rho'(l+1)) \leq \sup_{\sigma_{-i}\in\mathfrak{S}^{n-1}} \inf_{\sigma_i\in\mathfrak{S}} \mathsf{cost}_i((\sigma_{-i}, \sigma_i), \rho'(l+1)).$$

We obtain the required inequality

$$\begin{aligned}\mathsf{cost}_i(\rho_{\geq l}) &\leq \mathsf{cost}_i(\rho(l), \rho'(l+1)) + \sup_{\sigma_{-i}\in\mathfrak{S}^{n-1}} \inf_{\sigma_i\in\mathfrak{S}} \mathsf{cost}_i((\sigma_{-i}, \sigma_i), \rho'(l+1)) \\ &\leq \mathsf{cost}_i(\rho(l), c) + \mathsf{val}_{i,c}.\end{aligned}$$

Conversely, consider a path ρ that satisfies the condition. We are going to construct a Nash equilibrium having outcome ρ. The idea is that players will follow ρ, and if some player i deviates, then the coalition $-i$ will apply a joint strategy to maximize the cost of Player i, thus achieving at least $\mathsf{val}_{i,c}$, where c is the first configuration where deviation is detected.

Let us define the punishment function $\mathcal{P}_\rho \colon \mathsf{Paths}(\langle \mathcal{A}, n\rangle) \to [\![n]\!] \cup \{\perp\}$ which keeps track of the deviating players and the step where such a player has deviated. For path $h' = h(c, w, c')$, we write

$$\mathcal{P}_\rho(h') = \begin{cases} \perp & \text{if } h' \leq_{\mathsf{pref}} \rho, \\ i & \text{if } h \leq_{\mathsf{pref}} \rho, h(c, w, c') \not\leq_{\mathsf{pref}} \rho, \text{ and } i \in [\![n]\!] \text{ min. s.t. } c'(i) \neq \rho(|h|+1)(i), \\ \mathcal{P}_\rho(h) & \text{otherwise.} \end{cases}$$

Intuitively, $\perp$ means that no players have deviated from ρ in the current path. If $\mathcal{P}_\pi(h) = j$, then Player j was among the first players to deviate from ρ in the path h; so for some l,

$h(l)(j) = \rho(l)(j)$ but $h(l+1)(j) \neq \rho(l+1)(j)$. Notice that if several players deviate at the same step, there are no conditions to be checked, and the strategy can be chosen arbitrarily. For each configuration c and coalition $-i$, let $\sigma_{-i,c}$ be the strategy of coalition $-i$ maximizing the cost of Player i from configuration c; thus achieving at least $\mathsf{val}_{i,c}$. Player j's strategy in this coalition, for $j \neq i$, is denoted $\sigma_{-i,c,j}$. For path $h' = h(c, w, c')$, define

$$\tau_i(h') = \begin{cases} (c'(i), m(i), c''(i)) & \text{if } \mathcal{P}_\rho(h') = \perp, \rho(|h'|+1) = (c', w', c''), \\ & \text{ and } m \in E^n \text{ is such that } T(c', m) = (w', c''), \\ \text{arbitrary} & \text{if } \mathcal{P}_\rho(h') = i, \\ \sigma_{-j,c,i}(h') & \text{if } \mathcal{P}_\rho(h') = j \text{ for some } j \neq i. \end{cases}$$

The first case ensures that the outcome of the profile $(\tau_i)_{i\in[\![n]\!]}$ is ρ. The third case means that Player i follows the coalition strategy $\sigma_{-j,c}$ after Player j has deviated to configuration c. The second case corresponds to the case where Player i has deviated: the precise definition of this part of the strategy is irrelevant.

Let us show that this profile is indeed a Nash equilibrium. Consider any player $j \in [\![n]\!]$ and any strategy τ'_j. Let ρ' denote the outcome of (τ_{-j}, τ'_j), and l the index of the last configuration where ρ and ρ' are identical. We have

$$\begin{aligned} \mathsf{cost}_j((\tau_{-j}, \tau'_j)) &= \mathsf{cost}_j(\rho_{\leq l}) + \mathsf{cost}_j(\rho(l), \rho'(l+1)) + \mathsf{cost}_j((\tau_{-j}, \tau_j), \rho'_{\leq l+1}) \\ &\geq \mathsf{cost}_j(\rho_{\leq l}) + \mathsf{cost}_j(\rho(l), \rho'(l+1)) + \mathsf{val}_{j,\rho'(l+1)(j)} \\ &\geq \mathsf{cost}_j((\tau_i)_{i\in[\![n]\!]}), \end{aligned}$$

where the second line follows from the fact that the coalition switches to a strategies ensuring a cost of at least $\mathsf{val}_{j,\rho'(l)(j)}$ at step l; and the third line is obtained by assumption. This shows that $(\tau_i)_{i\in[\![n]\!]}$ is indeed a Nash equilibrium and concludes the proof. ◀

Algorithm. We define a graph that describes the set of outcomes of Nash equilibria by augmenting the n-weighted configuration graph $\mathcal{M} = \langle C, T\rangle$. For any real vector $\vec{\gamma} = (\gamma_i)_{i\in[\![n]\!]}$, we define the weighted graph $\mathcal{G}_{\langle \mathcal{A},n\rangle,\vec{\gamma}} = \langle C', T'\rangle$ with $C' = C \times ([\![Y]\!] \cup \{0, \infty\})^n$ where $Y = |V| \cdot \kappa$, and $T' \subseteq C' \times \mathbb{N} \times C'$; remember that all players have a strategy realizing cost at most Y in $\langle \mathcal{A}, n\rangle$. The initial state is $(c_{\mathsf{src}}, \infty^n)$. The set of transitions T' is defined as follows: $((c, b), z, (c', b')) \in T'$ if, and only if, there exists $(c, w, c') \in T$, $z = \vec{\gamma} \cdot w$ (where $\cdot$ is dot product), and for all $i \in [\![n]\!]$,

$$b'_i = \min(b_i - w_i, \min_{c''\in \mathsf{dev}_i(c(i),c'(i))} \mathsf{cost}_i(c, c'') + \mathsf{val}_{i,c''} - w_i). \tag{1}$$

Notice that by definition of C', b'_i must be nonnegative for all $i \in [\![n]\!]$, so there are no transitions $((c, b), z, (c', b'))$ if the above expression is negative for some i. Notice also that the size of $\mathcal{G}_{\langle \mathcal{A},n\rangle,\vec{\gamma}}$ is doubly-exponential in that of the input $\langle \mathcal{A}, n\rangle$, since this is already the case for C, while Y is singly-exponential.

Intuitively, for any path ρ that visits some state (c, b) in this graph, in order for ρ to be compatible with a Nash equilibrium, each player i must have cost no more than b_i in the rest of the path. In fact, the second term of the minimum in (1) is the least cost Player i could guarantee by not following (c, w, c') but going to some other configuration $c'' \in \mathsf{dev}_i(c, c')$, so the bound b_i is used to guarantee that these deviations are not profitable. The definition of b'_i in (1) is the minimum of $b_i - w_i$ and the aforementioned quantity since we check both the previous bound b_i, updated with the current cost w_i (which gives the left term), and the non-profitability of a deviation at the previous state (which is the right term). If this minimum becomes negative, this precisely means that at an earlier point in the current path, there was a strategy for Player i which was more profitable than the current path regardless of the strategies of other players; so the current path cannot be the outcome of a Nash equilibrium. This is why the definition of $\mathcal{G}_{\langle \mathcal{A},n\rangle,\vec{\gamma}}$ restricts the state space to nonnegative values for the b_i.

We prove that computing the cost of a Nash equilibrium minimizing the $\vec{\gamma}$-weighted social cost reduces to computing a shortest path in $\mathcal{G}_{\langle \mathcal{A},n\rangle,\vec{\gamma}}$. In particular, letting $\gamma_i = 1$ for all $i \in [\![n]\!]$, a $\vec{\gamma}$-minimal Nash equilibrium is a best Nash equilibrium (minimizing the social cost), while taking $\gamma_i = -1$ for all $i \in [\![n]\!]$, we get a worst Nash equilibrium (maximizing the social cost).

▶ **Theorem 14.** *For any dynamic NCG $\langle \mathcal{A}, n\rangle$ and vector $\vec{\gamma}$, the cost of the shortest path from $(c_{\mathsf{src}}, \infty^n)$ to some (c_{tgt}, b) in $\mathcal{G}_{\langle \mathcal{A},n\rangle,\vec{\gamma}}$ is the cost of a $\vec{\gamma}$-minimal Nash equilibrium.*

Proof. We show that for each path of $\langle \mathcal{A}, n\rangle$ from c_{src} to c_{tgt}, there is a path in $\mathcal{G}_{\langle \mathcal{A},n\rangle,\vec{\gamma}}$ from $(c_{\mathsf{src}}, \infty^n)$ to some (c_{tgt}, b) with the same cost, and vice versa.

Consider a Nash equilibrium $\pi = (\sigma_j)_{j\in[\![n]\!]}$ with outcome $\rho = (c_j, w_j, c_{j+1})_{1\le j<l}$. We build a sequence $b_1, b_2, \ldots$ such that $\rho' = ((c_j, b_j), \vec{\gamma}\cdot w_j, (c_{j+1}, b_{j+1}))_{1\le j<l}$ is a path of $\mathcal{G}_{\langle \mathcal{A},n\rangle,\vec{\gamma}}$. We set $b_1(j) = \infty$ for all $j \in [\![n]\!]$. For $j \ge 1$, define

$$b_{j+1}(i) = \min\left(b_j(i) - w_j(i), \min_{c''\in\mathsf{dev}_j(c_j(i),c_{j+1}(i))} \mathsf{cost}_i(c_j, c'') + \mathsf{val}_{i,c''} - w_j(i)\right).$$

We are going to show that for all $1 \le j \le l$, $\mathsf{cost}_i(\rho_{\ge j}) \le b_j$, which shows that $b_j \ge 0$, and thus ρ' is a path of $\mathcal{G}_{\langle \mathcal{A},n\rangle,\vec{\gamma}}$.

We show this by induction on j. This is clear for $j = 1$. Assume this holds up to $j \ge 1$. We have, by induction that $\mathsf{cost}_i(\rho_{\ge j}) \le b_j(i)$ for all $i \in [\![n]\!]$. Moreover, since π is a Nash equilibrium, by Lemma 13,

$$\forall i \in [\![n]\!], \mathsf{cost}_i(\rho_{\ge j}) \le \min_{c''\in\mathsf{dev}_i(\rho(j),\rho(j+1))} \mathsf{val}_{i,c''} + \mathsf{cost}_i(\rho(j), c'').$$

Therefore,

$$\begin{aligned}\mathsf{cost}_i(\rho_{\ge j+1}) &= \mathsf{cost}_i(\rho_{\ge j}) - w_j(i)\\ &\le \min(b_j(i) - w_j(i), \min_{c''\in\mathsf{dev}_i(\rho(j),\rho(j+1))} \mathsf{val}_{i,c''} + \mathsf{cost}_i(\rho(j), c'') - w_j(i))\end{aligned}$$

as required, and both paths have the same $\vec{\gamma}$-weighted cost.

Consider now a path $((c_i, b_i), z_i, (c_{i+1}, b_{i+1}))_{1\le i<l}$ in $\mathcal{G}_{\langle \mathcal{A},n\rangle,\vec{\gamma}}$. By the definition of $\mathcal{G}_{\langle \mathcal{A},n\rangle,\vec{\gamma}}$, there exists $w_1, w_2, \ldots$ such that $\rho = (c_j, w_j, c_{j+1})_{1\le j<l}$ is a path of $\langle \mathcal{A}, n\rangle$, and $z_j = \vec{\gamma}\cdot w_j$. So it only remains to show that that ρ is the outcome of a Nash equilibrium. We will show that ρ satisfies the criterion of Lemma 13. We show by backwards induction on $1 \le j \le l$ that for all $i \in [\![n]\!]$,

1. $\mathsf{cost}_i(\rho_{\ge j}) \le b_j(i)$,
2. $\mathsf{cost}_i(\rho_{\ge j}) \le \min_{c''\in\mathsf{dev}_i(\rho(j),c'')} \mathsf{cost}_i(\rho(j), c'') + \mathsf{val}_{i,c''}$.

For $j = l$, we have $\mathsf{cost}_i(\rho_{\ge l}) = 0$ so this is trivial. Assume the property holds down to $j + 1$ for some $1 \le j < l$. By induction hypothesis, we have

$$\mathsf{cost}_i(\rho_{\ge j+1}) \le b_{j+1}(i) = \min\left(b_j(i) - w_j(i), \min_{c''\in\mathsf{dev}_i(\rho(j),c'')} \mathsf{cost}_i(\rho(j), c'') + \mathsf{val}_{i,c''} - w_j(i)\right).$$

Therefore,

$$\mathsf{cost}_i(\rho_{\ge j}) = \mathsf{cost}_i(\rho_{\ge j+1}) + w_j(i) \le \min\left(b_j(i), \min_{c''\in\mathsf{dev}_i(\rho(j),c'')} \mathsf{cost}_i(\rho(j), c'') + \mathsf{val}_{i,c''}\right),$$

as required. By Lemma 13, ρ is the outcome of a Nash equilibrium. ◀

Thanks to Theorem 14, we can compute the costs of the best and worst NEs of $\langle\mathcal{A}, n\rangle$ in exponential space. We can also decide the existence of an NE with constraints on the costs (both social and individual), by non-deterministically guessing an outcome and checking in $\mathcal{G}_{\langle\mathcal{A},n\rangle,\vec{\gamma}}$ that it is indeed an NE. We obtain the following conclusion:

▶ **Corollary 15.** *In dynamic NCGs, the constrained Nash-equilibrium problem is in EXPSPACE.*

Proof. As noted earlier, the number of vertices in $\mathcal{G}_{\langle\mathcal{A},n\rangle,\vec{\gamma}}$ is doubly exponential since $|C| = |V|^n$ is doubly exponential. Storing a configuration and computing its successors can be performed in exponential space. One can thus guess a path of size at most the size of the graph and check whether its cost is less than the given bound. This can be done using exponential-space counters, and provides us with an EXPSPACE algorithm. ◀

Note that one can effectively compute a Nash-equilibrium strategy profile satisfying the constraints in doubly-exponential time by finding the shortest path of $\mathcal{G}_{\langle\mathcal{A},n\rangle,\vec{\gamma}}$, and applying the construction of (the proof of) Lemma 13.

▶ Remark 16. The exponential complexity is due to the encoding of the number of players in binary. If we consider asymmetric NCGs, in which the source-target pairs would be given explicitly for all players, the size of $\mathcal{G}_{\langle\mathcal{A},n\rangle,\vec{\gamma}}$ would be singly-exponential, and the constrained Nash-equilibrium problem would be in PSPACE.

5 Subgame-perfect equilibria

In this section, we characterize the outcomes of SPEs and decide the existence of SPEs with constraints on the social cost. We follow the approach of [9], extending it to the setting of concurrent weighted games, which we need to handle dynamic NCGs.

Characterization of outcomes of SPE. Consider a dynamic NCG $\langle\mathcal{A}, n\rangle$, and the associated configuration graph $\mathcal{M} = \langle C, T\rangle$. We partition the set C of configurations into $(X_j)_{0\leq j\leq n}$ such that a configuration c is in X_j if, and only if, $j = \#\{i \in [\![n]\!] \mid c(i) = \mathsf{tgt}\}$. Since tgt is a sink state in $\mathcal{A}$, if there is a transition from some configuration in X_j to some configuration in X_k, then $k \geq j$. We define $X_{\geq j} = \bigcup_{i\geq j} X_i$, $Z_j = \{(c, w, c') \in T \mid c \in X_j\}$ and $Z_{\geq j} = \{(c, w, c') \in T \mid c \in X_{\geq j}\}$.

Following [9], we inductively define a sequence $(\lambda^{j^*})_{0\leq j\leq n}$, where each $\lambda^{j^*} = \langle\lambda_i^{j^*}\rangle_{i\in[\![n]\!]}$ is a n-tuple of labeling functions $\lambda_i^{j^*} : Z_{\geq j} \to \mathbb{N} \cup \{-\infty, +\infty\}$. This sequence will be used to characterize outcomes of SPEs through the notion of λ-*consistency*:

▶ **Definition 17.** *Let $j \leq n$, and $\lambda = (\lambda_i)_{i\in[\![n]\!]}$ be a family of functions such that $\lambda\colon Z_{\geq j} \to \mathbb{N} \cup \{-\infty, +\infty\}$ Let $c \in X_{\geq j}$. A finite path $\rho = (t_k)_{1\leq k<|\rho|}$ from c ending in c_{tgt} is said to be λ-consistent whenever for any $i \in [\![n]\!]$ and any $1 \leq k < |\rho|$, it holds $\mathsf{cost}_i(\rho_{\geq k}) \leq \lambda_i(t_k)$. We write $\Gamma_\lambda(c)$ for the set of all λ-consistent paths from c.*

We now define λ^{j^*} for all $0 \leq j \leq n$ in such a way that, for all $c \in X_{\geq j}$, $\Gamma_{\lambda^{j^*}}(c)$ is the set of all outcomes of SPEs in the subgame rooted at c. The case where $j = n$ is simple: we have $X_{\geq n} = \{c_{\mathsf{tgt}}\}$ and $Z_{\geq n} = \{(c_{\mathsf{tgt}}, 0^n, c_{\mathsf{tgt}})\}$; there is a single path, which obviously is the outcome of an SPE since no deviations are possible. For all $i \in [\![n]\!]$, we let $\lambda_i^{n^*}(c_{\mathsf{tgt}}, 0^n, c_{\mathsf{tgt}}) = 0$.

Now, fix $j < n$, assuming that $\lambda^{(j+1)^*}$ has been defined. In order to define λ^{j^*}, we introduce an intermediary sequence $(\mu_i^k)_{k\geq 0, i\in[\![n]\!]}$, with $\mu_i^k\colon Z_{\geq j} \to \mathbb{N} \cup \{-\infty, +\infty\}$, of which $(\lambda_i^{j^*})_{i\in[\![n]\!]}$ will be the limit.

Functions μ_i^k mainly operate on $Z_j = Z_{\geq j} \setminus Z_{\geq j+1}$: for any $\mathbf{e} \in Z_{\geq j+1}$, we let $\mu_i^k(\mathbf{e}) = \lambda_i^{(j+1)^*}(\mathbf{e})$. Now, for $\mathbf{e} = (c, w, c') \in Z_j$, $\mu_i^k(e)$ is defined inductively as follows:

- $\mu_i^0(\mathbf{e}) = 0$ if $c(i) = \mathsf{tgt}$, and $\mu_i^0(\mathbf{e}) = +\infty$ otherwise;
- for $k > 0$, μ^k is defined from μ^{k-1} following three cases: if $c(i) = \mathsf{tgt}$, then $\mu_i^k(\mathbf{e}) = 0$; if $\Gamma_{\mu^{k-1}}(c') = \emptyset$ for some $(c, w', c') \in T$, then $\mu_i^k(\mathbf{e}) = -\infty$; otherwise,

$$\mu_i^k(\mathbf{e}) = \min_{c'' \in \mathsf{dev}_i(c,c')} \sup_{\rho \in \Gamma_{\mu^{l-1}}(c'')} (\mathsf{cost}_i(c, c'') + \mathsf{cost}_i(\rho))$$

We can then prove that for any $e \in Z_{\geq j}$ and any $k > 0$, $\mu_i^k(e) \geq \mu_i^{k-1}(e)$. It follows that the sequence $(\mu^k)_{k \geq 0}$ stabilizes, and we can define λ^{j^*} as its limit. Let $\Gamma^* = \Gamma_{\lambda^{0^*}}$. Then:

▶ **Theorem 18.** *A path ρ in $\mathcal{G} = \langle \mathcal{A}, n \rangle$ is the outcome of an SPE if, and only if, $\rho \in \Gamma^*(c_{\mathsf{src}})$.*

Algorithm. It remains to compute the sequence $(\mu^k)_{k \geq 0}$ (which will include checking non-emptiness of the corresponding Γ-sets), and to bound the stabilization time. To this aim, with any family $\mu = (\mu_i)_{i \in [\![n]\!]}$ of functions as above and any configuration c, we associate an infinite-state *counter graph* $\mathbb{C}[\mu, c] = \langle C', T' \rangle$ to capture all μ-consistent paths from c:

- the set of vertices is $C' = C \times (\mathbb{N} \cup \{+\infty\})^{[\![n]\!]}$;
- T' contains all edges $((d, b), w, (d', b'))$ for which (d, w, d') is an edge of $\mathcal{M}$ and for all $i \in [\![n]\!]$, $b'(i) = 0$ if $d(i) = \mathsf{tgt}$, and $b'_i = \min\{b_i - w_i, \mu_i(d, w, d') - w_i\}$ otherwise (provided that $b'_i \geq 0$ for all i, in order for (d', b') to be an edge of $\mathbb{C}[\mu, c]$).

With the initial configuration c, we associate b^c such that $b_i^c = 0$ if $c(i) = \mathsf{tgt}$ and $b_i^c = +\infty$ otherwise: this configuration imposes no constraint, since no edges has been taken yet. Intuitively, in configuration (d, b), b is used to enforce μ-consistency: each edge taken along a path imposes a constraint on the cost of the players for the rest of the path; this constraint is added to the constraints of the earlier edges, and propagated along the path. We can prove that the number of reachable states from (c, b^c) in $\mathbb{C}[\mu, c]$, which we denote with $|C'|_r$, is bounded by $|C| \cdot (n \cdot |V|^n \cdot \kappa)^{|V|}$.

Computing λ^{j^*} from $\lambda^{(j+1)^*}$ amounts to inductively computing $(\mu_i^{k+1})_{i \in [\![n]\!]}$ from μ^k for edges $e = (c, w, c') \in Z_j$, until stabilization. Since $\mathbb{C}[\mu^k, d]$ can be proved to capture μ^k-consistent paths from d, the computation mainly amounts to checking the existence of paths in such counter graphs, which can be performed in doubly-exponential space. Stabilization can be shown to occur within $|V|(1 + n \cdot \kappa \cdot |E|^n)$ steps. In the end:

▶ **Theorem 19.** *The existence of SPEs in a dynamic NCG can be decided in 2*EXPSPACE*.*

▶ Remark 20. Again, our algorithm is not specific to the symmetric setting of our dynamic NCGs; in an asymmetric context, where the number of players would be given in unary, our algorithm would run in EXPSPACE.

Existence of constrained SPEs. The algorithm above can be extended to compute the cost of the best and worst SPEs, and to include constraints on the costs (both social and individual) of the SPEs we are looking for.

First, for any vector $\vec{\gamma} = (\gamma_i)_{i \in [\![n]\!]}$, we define the *$\vec{\gamma}$-counter graph* $\mathbb{C}[\lambda^*, c_{\mathsf{src}}, \vec{\gamma}]$, which is obtained from $\mathbb{C}[\lambda^*, c_{\mathsf{src}}]$ by replacing the cost vector w on the edges with $\vec{\gamma} \cdot w$.

We can then compute the cost of a $\vec{\gamma}$-minimal SPE by checking existence of a path from $(c_{\mathsf{src}}, b^{c_{\mathsf{src}}})$ to (c_{tgt}, b) in $\mathbb{C}[\lambda^*, c_{\mathsf{src}}, \vec{\gamma}]$, which minimizes the $\vec{\gamma}$-weighted social cost. Again, letting $\gamma_i = 1$ for all $i \in [\![n]\!]$, a $\vec{\gamma}$-minimal SPE is a best SPE, while taking $\gamma_i = -1$ for all $i \in [\![n]\!]$, we get a worst SPE (maximizing social cost).

We can also solve the constrained-SPE-existence problem by non-deterministically guessing an outcome and checking that it is a path in $\mathbb{C}[\lambda^{0^*}, c_{\mathsf{src}}]$ and that it satisfies the constraints. In each case, we can inductively build a strategy profile witnessing the fact that the selected path is the outcome of an SPE.

6 Conclusion and future works

In this paper, we introduced dynamic network congestion games, and studied the complexity of various decision and computation problems concerning social optima, Nash equilibria and subgame perfect equilibria. Our algorithms allow us to compute the price of anarchy and price of stability for those games.

There are couple of areas that are left open in our discussion: possibly the foremost one being the complexity gaps of the decision problems we talked about. As of yet, we do not have interesting lower bounds for constraint NE or constraint SPE problem, so definitely one direction is there for completing the picture. Another aspect of what we do not address in this paper is to obtain *bounds* on PoA/PoSs of our model. Even though we are specifically interested in the measure(s) for a given instance, nonetheless obtaining such bounds could be interesting.

What we are mostly interested in as future work, is to compute how the price of anarchy and the price of stability (and costs of equilibria and social optimum) evolve when the number of players, seen as a parameter, grows.

References

1. Elliot Anshelevich, Anirban Dasgupta, Jon Kleinberg, Éva Tardos, Tom Wexler, and Tim Roughgarden. The price of stability for network design with fair cost allocation. *SIAM Journal on Computing*, 38(4):1602–1623, 2008. doi:10.1137/070680096.
2. Guy Avni, Shibashis Guha, and Orna Kupferman. Timed network games. In Kim Guldstrand Larsen, Hans L. Bodlaender, and Jean-François Raskin, editors, *Proceedings of the 42nd International Symposium on Mathematical Foundations of Computer Science (MFCS'17)*, volume 84 of *Leibniz International Proceedings in Informatics*, pages 37:1–37:16. Leibniz-Zentrum für Informatik, August 2017. doi:10.4230/LIPIcs.MFCS.2017.37.
3. Guy Avni, Shibashis Guha, and Orna Kupferman. Timed network games with clocks. In Igor Potapov, Paul G. Spirakis, and James Worrell, editors, *Proceedings of the 43rd International Symposium on Mathematical Foundations of Computer Science (MFCS'18)*, volume 117 of *Leibniz International Proceedings in Informatics*, pages 23:1–23:18. Leibniz-Zentrum für Informatik, August 2018. doi:10.4230/LIPIcs.MFCS.2018.23.
4. Guy Avni, Thomas A. Henzinger, and Orna Kupferman. Dynamic resource allocation games. *Theoretical Computer Science*, 807:42–55, February 2020. doi:10.1016/j.tcs.2019.06.031.
5. Baruch Awerbuch, Yossi Azar, and Amir Epstein. Large the price of routing unsplittable flow. In Harold N. Gabow and Ronald Fagin, editors, *Proceedings of the 37th Annual ACM Symposium on the Theory of Computing (STOC'05)*, pages 57–66. ACM Press, May 2005. doi:10.1145/1060590.1060599.
6. Nathalie Bertrand, Nicolas Markey, Suman Sadhukhan, and Ocan Sankur. Dynamic network congestion games. *CoRR*, abs/2009.13632, 2020. arXiv:2009.13632.
7. Dimitri P. Bertsekas and John N. Tsitsiklis. *Parallel and distributed computation: numerical methods*. Prentice Hall, 1989.
8. Umang Bhaskar, Lisa Fleischer, and Elliot Anshelevich. A Stackelberg strategy for routing flow over time. *Games and Economic Behavior*, 92:232–247, July 2015. doi:10.1016/j.geb.2013.09.004.

9 Thomas Brihaye, Véronique Bruyère, Aline Goeminne, Jean-François Raskin, and Marie Van den Bogaard. The complexity of subgame perfect equilibria in quantitative reachability games. In Wan J. Fokkink and Rob van Glabbeek, editors, *Proceedings of the 30th International Conference on Concurrency Theory (CONCUR'19)*, volume 140 of *Leibniz International Proceedings in Informatics*, pages 13:1–13:16. Leibniz-Zentrum für Informatik, August 2019. doi:10.4230/LIPIcs.CONCUR.2019.13.

10 Martin Hoefer, Vahab S. Mirrokni, Heiko Röglin, and Shang-Hua Teng. Competitive routing over time. *Theoretical Computer Science*, 412(39):5420–5432, September 2011. doi:10.1016/j.tcs.2011.05.055.

11 Leonid Khachiyan, Endre Boros, Konrad Borys, Khaled Elbassioni, Vladimir Gurvich, Gabor Rudolf, and Jihui Zhao. On short paths interdiction problems: Total and node-wise limited interdiction. *Theory of Computing Systems*, 43(2):204–233, August 2008. doi:10.1007/s00224-007-9025-6.

12 Ronald Koch and Martin Skutella. Nash equilibria and the price of anarchy for flows over time. *Theory of Computing Systems*, 49(1):71–97, July 2011. doi:10.1007/s00224-010-9299-y.

13 Elias Koutsoupias and Christos H. Papadimitriou. Worst-case equilibria. *Computer Science Review*, 3(2):65–69, May 2009. doi:10.1016/j.cosrev.2009.04.003.

14 Elias Koutsoupias and Katia Papakonstantinopoulou. Contention issues in congestion games. In Artur Czumaj, Kurt Mehlhorn, Andrew Pitts, and Roger Wattenhofer, editors, *Proceedings of the 39th International Colloquium on Automata, Languages and Programming (ICALP'12) – Part II*, volume 7392 of *Lecture Notes in Computer Science*, pages 623–635. Springer-Verlag, July 2012. doi:10.1007/978-3-642-31585-5_55.

15 Carol A. Meyers and Andreas S. Schulz. The complexity of welfare maximization in congestion games. *Networks*, 59(2):252–260, March 2012. doi:10.1002/net.20439.

16 Dov Monderer and Lloyd S. Shapley. Potential games. *Games and Economic Behavior*, 14(1):124–143, May 1996. doi:10.1006/game.1996.0044.

17 Michal Penn, Maria Polukarov, and Moshe Tennenholtz. Random order congestion games. *Mathematics of Operations Research*, 34(3):706–725, August 2009. doi:10.1287/moor.1090.0394.

18 Lili Qiu, Richard Yang, Yin Zhang, and Scott Shenker. On selfish routing in internet-like environments. *IEEE Transactions on Computers*, 14(4):725–738, August 2006. doi:10.1109/TNET.2006.880179.

19 Robert W. Rosenthal. A class of games possessing pure-strategy Nash equilibria. *International Journal of Game Theory*, 2(1):65–67, December 1973. doi:10.1007/BF01737559.

20 Tim Roughgarden. Routing games. In Noam Nisan, Tim Roughgarden, Éva Tardos, and Vijay V. Vazirani, editors, *Algorithmic Game Theory*, chapter 18, page 461–486. Cambridge University Press, 2007.

21 Subhash Suri, Csaba D. Tóth, and Yunhong Zhou. Selfish load balancing and atomic congestion games. *Algorithmica*, 47(1):79–96, 2007. doi:10.1007/s00453-006-1211-4.