

Timed Network Games*

Guy Avni¹, Shibashis Guha², and Orna Kupferman²

1 IST Austria

2 Hebrew University of Jerusalem

Abstract

Network games are widely used as a model for selfish resource-allocation problems. In the classical model, each player selects a path connecting her source and target vertex. The cost of traversing an edge depends on the number of players that traverse it. Thus, it abstracts the fact that different users may use a resource at different times and for different durations, which plays an important role in defining the costs of the users in reality. For example, when transmitting packets in a communication network, routing traffic in a road network, or processing a task in a production system, the traversal of the network involves an inherent delay, and so sharing and congestion of resources crucially depends on time.

We study *timed network games*, which add a time component to network games. Each vertex v in the network is associated with a cost function, mapping the load on v to the price that a player pays for staying in v for one time unit with this load. In addition, each edge has a guard, describing time intervals in which the edge can be traversed, forcing the players to spend time on vertices. Unlike earlier work that add a time component to network games, the time in our model is continuous and cannot be discretized. In particular, players have uncountably many strategies, and a game may have uncountably many pure Nash equilibria. We study properties of timed network games with cost-sharing or congestion cost functions: their stability, equilibrium inefficiency, and complexity. In particular, we show that the answer to the question whether we can restrict attention to boundary strategies, namely ones in which edges are traversed only at the boundaries of guards, is mixed.

Digital Object Identifier 10.4230/LIPIcs.MFCS.2017.37

1 Introduction

Network games (NGs, for short) [9, 38, 39] constitute a well studied model of non-cooperative games. The game is played among selfish players on a network, which is a directed graph. Each player has a source and a target vertex, and a strategy is a choice of a path that connects these two vertices. The cost of a player is the sum of costs of the edges her path traverses, where a cost of an edge depends on the *load* on it, namely the number of players using the edge. We distinguish between two types of costs. In *cost-sharing* games (a.k.a. network formation games), each edge has a cost and the players that use it split the cost among them, thus the load has a positive effect on cost. For example, when the costs correspond to prices, users that share a resource also share its price. Then, in *congestion* games, the load has a negative effect on cost: each edge has a non-decreasing *latency function* that maps the load on the edge to its cost given this load. For example, when the network models a road system and costs correspond to the traversal time, an increased load on an edge corresponds to a traffic jam, increasing the cost of the players that use it.

One limitation of NGs is that the cost of using a resource abstracts the fact that different users may use the resource at different times and for different durations. This is a real limitation, as time plays an important role in many real-life settings. For example, in a road or communication systems,

* The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013, ERC grant no 278410) and the Austrian Science Fund (FWF) under grants S11402-N23 (RiSE/SHiNE) and Z211-N23 (Wittgenstein Award).



© Avni, Guha, Kupferman;
licensed under Creative Commons License CC-BY

42nd International Symposium on Mathematical Foundations of Computer Science (MFCS 2017).

Editors: Kim G. Larsen, Hans L. Bodlaender, and Jean-Francois Raskin; Article No. 37; pp. 37:1–37:15

Leibniz International Proceedings in Informatics

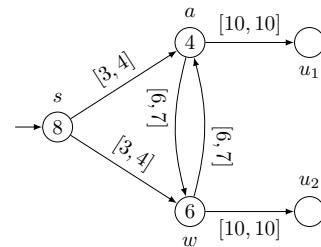


LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

congestion only affects cars or messages that use a road or a channel simultaneously. We are interested in settings in which congestion affects the QoS or the way a price is shared (rather than the travel time). For example, discomfort increases in a crowded train or price is shared by the passengers in a taxi without affecting the travel time. Similarly, in mobile networks, the call quality depends on the number of subscribers using the network simultaneously. As a third example, when processing a task in a production system, jobs move from one station to another. The way the cost of running the stations is shared by the jobs that use it depends on the time spent in the stations and on the synchronization among the jobs.

We introduce and study *timed network games* (TNGs, for short) – a new model that adds a time component on NGs. Similar to NGs, the game is played on a network and the players need to find a path from their source to target vertices. Rather than paying for the traversal of edges, in TNGs the players pay for spending time in vertices. Each edge in the network has a *guard*, which is a disjunction of time intervals that specifies when an edge can be traversed. Traversing an edge is done instantaneously. So, a strategy for a player is a *timed path*: a sequence of pairs $\langle v, t \rangle$ of a vertex v and the time t spent on v . When the path traverses an edge in the network, the guard of the edge must be satisfied. For an integer $k \in \mathbb{N}$, let $[k] = \{1, \dots, k\}$. Each vertex v has a cost function $r_v : [k] \rightarrow \mathbb{R}_{\geq 0}$ that assigns the cost of using v for one time unit, given the load in v . A *profile* in a TNG is then a vector of timed paths, namely the strategies of all players. Given a profile P , the cost of each player is induced by the cost functions of the vertices visited in her timed path, the time spent at each vertex, and the load on the vertices during these visits.

► **Example 1.** Consider an automobile service center with three stations: (s) tuning engine, (a) tire and air check, and (w) dry and wet wash. The costs for operating the stations per one time unit are 8, 4 and 6 respectively, and they are independent of the number of cars served. Accordingly, cost is shared by the users. There are two billing counters, u_1 and u_2 , for dropped-in and registered cars. The setting is modeled by the TNG below. As described in the TNG, after spending some time in s , the cars can alternate between stations w and a . Assume that there are two players, and consider the profile P in which the first player chooses the timed path $(s, 3), (a, 7), u_1$ and the second player chooses the timed path $(s, 3), (a, 4), (w, 3), u_2$. Player 1’s cost in P is $8/2 \cdot 3 + 4/2 \cdot 4 + 4 \cdot 3 = 32$ and Player 2’s cost is $8/2 \cdot 3 + 4/2 \cdot 4 + 6 \cdot 3 = 38$. Another possible profile in this game is P' , in which the strategies of the two players are $(s, 3), (w, 4), (a, 3), u_1$ and $(s, 3), (w, 7), u_2$. Now, the costs are 36 and 42, respectively. ◀



There has been reference to *time* already in early work on *flow networks* [27]. Research spans from pioneering and theoretical work on flow networks in which congestion leads to queues (c.f., [42, 43]) to nowadays practical research on traffic engineering in software defined networks [2]. These works, however, do not address the problem from a game-theoretic perspective. To the best of our knowledge, the first works to consider network games with a time component are [37] and [29]. In [37], the focus is still on flow networks, and it enriches [42, 43] by viewing infinitesimal flow particles as selfish agents (see also [16]). Closer to our work, network games with time components were studied in [29, 32, 36]. These models differ from our model in two main aspects. First, the cost a player pays in these models is the time it takes to reach its destination, and our cost represents the QoS. Second, time is discrete in these models so the set of strategies the players choose is finite, whereas the source of the difficulty of our model is the real-time and the fact that the players have uncountably many strategies. The closest to our model is a model studied in [29], which studies a QoS pricing but using discrete time.

Our model of TNGs is the first to add real-time considerations to the strategies of the players. Indeed, a strategy for a player is not just the path of edges she is going to traverse, but also the time

spent in vertices, which can be any number in $\mathbb{R}_{\geq 0}$. Thus, even if we restrict attention to simple paths, each player has uncountably many strategies. This continuous time and the richness of strategies that it brings with it is also a key difference between TNGs and NGs. Our model is inspired by *timed automata* [5]. There too, time is continuous, transitions between states are guarded by time constraints, and so is the time spent in a state. There are typically uncountably many runs of a timed automaton, corresponding to the uncountably many strategies a player typically has in our TNGs. The fact timed automata handle continuous time makes them the prominent formalism for specifying real-time on-going behaviors, and they are way more useful than formalisms in which time has been discretized (c.f., temporal logic with discrete clocks [24], or the fictitious-clock approach of [28]). We note that our TNGs correspond to a restricted class of timed automata, as our guards refer to the global time and cannot express, for example, a bound on the time spent in a vertex. In Section 7 we discuss the extension of our model to a richer one.

Note that, as in the *time-dependent cost* model of [29], load does not affect travel time and only affects the cost. Unlike [29], in TNGs time is continuous, which enables TNGs to model richer settings in practice. Note also that the cost function may model various applications. Consider, for example, a communication network with servers that encode or decode messages. A typical cost function for a server is the inverse of the quality of the signal, which is related to the number of bits needed to encode a message. Assuming that a server can handle a certain amount of data per unit time, this cost is the reciprocal of the number of bits used to encode a message. If the server allows a 16-bit encoding of a message when it serves less than 128 users simultaneously, and allows an 8-bit encoding when it serves between 128 and 256 users simultaneously, then the cost function maps x to $\frac{1}{16}$, for $x \leq 128$, and to $\frac{1}{8}$, for $129 \leq x \leq 256$, reflecting a better quality of the received message when load goes below 128 [34].

The first question that arises in the context of games is the existence of *stable outcomes* of the game. In the context of NGs, the most prominent stability concept is that of a (pure) *Nash equilibrium* (NE, for short) – a profile such that no player can decrease her cost by unilaterally deviating from her current strategy¹. Decentralized decision-making may lead to solutions that are sub-optimal from the point of view of society as a whole. The standard measures to quantify the inefficiency incurred due to selfish behavior is the *price of stability* (PoS) [9] and the *price of anarchy* (PoA) [31]. In both measures we compare against a *social optimum* (SO, for short), namely a profile that minimizes the sum of costs of all players. The PoS (PoA, respectively) is the best-case (worst-case) inefficiency of an NE; that is, the ratio between the cost of a best (worst) NE and an SO. In Example 1, profile P is an SO, and is also a (best) NE, while profile P' is a worst NE. Note that there can be uncountably many NEs in the TNG in Example 1. Indeed, for all $t \in [3, 4]$, the profile P_t with the strategies $(s, 3), (a, t), (w, 4 - t)(a, 3)u_1$ and $(s, 3), (a, t), (w, 7 - t)u_2$, is an NE with costs $8/2 \cdot 3 + 4/2 \cdot t + 6/2 \cdot (4 - t) + 4 \cdot 3 = 36 - t$ and $8/2 \cdot 3 + 4/2 \cdot t + 6/2 \cdot (4 - t) + 6 \cdot 3 = 42 - t$.

The picture of stability and equilibrium inefficiency for standard NGs is well understood. Every NG has an NE, and in fact these games are *potential games* [38], thus every sequence of *best response* moves, namely moves that the players perform in order to reduce their costs, converges to an NE. For k -player cost-sharing NGs, the PoS and PoA are $\log k$ and k , respectively [9]. For congestion games with affine cost functions, PoS ≈ 1.577 [22, 3] and PoA = $\frac{5}{2}$ [23].

The fact a TNG has uncountably many profiles makes the adoption of results known for NGs questionable. Let us elaborate on this point. Consider a TNG \mathcal{T} , and a finite set $T \subseteq \mathbb{R}_{\geq 0}$ of time points. Note that there are only finitely many T -profiles in \mathcal{T} (that is, profiles with T -strategies, in which all edges are taken at some time point in T). We show that once we restrict attention to T -profiles, we can construct an NG that is isomorphic to \mathcal{T} , in the sense that there is a cost-preserving

¹ Throughout this paper, we consider *pure* strategies, as is the case for the vast literature on cost-sharing games.

bijection between profiles in the NG and T -profiles in the TNG \mathcal{T} . While this enables us to reduce questions about T -profiles in the TNG to questions on NGs, it is not clear to which finite set T we can restrict attention. In the setting of timed automata, much work has been done on obtaining decidability by partitioning $\mathbb{R}_{\geq 0}$ into finitely many *regions*. Essentially, all time points within a region are bisimilar, in the sense that the actions the automaton may take inside all time points in a region, coincide [5]. Our challenge here is similar: searching for a finite set of time points that partitions $\mathbb{R}_{\geq 0}$ to finitely many intervals.

Recall that the source for delays in TNGs are time guards on the edges, where each guard is a disjunction of intervals $[a, b]$, for $a \leq b \in \mathbb{Q}_{\geq 0}$. We refer to the two end points of all guards as *boundaries*. One can suspect that we can restrict attention to *boundary strategies*, namely timed paths that traverse edges only at boundary time points, and *boundary profiles* in which all the players choose boundary strategies. We show that the situation is mixed. The good news follows from choosing T above to be the boundaries, thus we show that a boundary NE and SO exist and an NE can be found by performing best-response moves that use only boundary strategies. Unfortunately, however, one cannot restrict attention to boundary profiles, as the best and worst NEs need not be boundary. We show a best and worst NE is attained in TNGs, which is not a-priori guaranteed.

In terms of inefficiency, the reduction from TNGs to isomorphic NGs enables us to extend upper bounds on the PoS and PoA from NGs to TNGs. The adoption of lower bounds requires a reduction in the other direction – from NGs to TNGs, which we can show only for acyclic NGs. Consequently, we can apply only lower bounds known for acyclic NGs, which forces us to either prove direct bounds or to tighten lower bounds known for NGs to acyclic NGs. All in all, we are able to show that the PoS and PoA coincide for NGs and TNGs, except for the lower bound on the PoS of congestion TNGs, which we leave open. Finally, in terms of computational complexity, we prove that the problem of finding an NE is PLS-complete [30] for TNGs, which coincides with the complexity bounds for NGs [25, 41]. Proving membership in PLS follows easily from the reduction from TNGs to NGs. Proving hardness is more complex. For congestion TNGs, we are able to rely on known hardness results for congestion NGs, as they apply already for acyclic congestion NGs [1]. For cost-sharing TNGs we need a similar reduction from acyclic cost-sharing NGs, whose precise complexity is an open problem. Accordingly, we first settle the latter problem and prove that finding an NE in acyclic cost-sharing NG is PLS-hard, which allows us to prove the hardness result for cost-sharing TNGs.

Due to lack of space, some proofs appear in the full version, which can be found in the authors' homepages.

2 Preliminaries

We describe a (*closed*) *time interval* by $[m_1, m_2]$, for $m_1, m_2 \in \mathbb{R}_{\geq 0}$. We refer to m_1 and m_2 as the *start* and the *end interval boundaries*, respectively. A *guard* is the constant *true* or a disjunction of time intervals. A point in time $t \in \mathbb{R}_{\geq 0}$ *satisfies* a guard g if g is *true* or g includes a disjunct $[m_1, m_2]$ such that $m_1 \leq t \leq m_2$.

A *timed network* (TN) is a tuple $\langle V, E, \{g_e\}_{e \in E} \rangle$, where V is a set of vertices, $E \subseteq V \times V$ is a set of directed edges, and for each edge $e \in E$, the guard g_e specifies the time intervals during which e may be traversed. A *timed network game* (TNG) is $\mathcal{T} = \langle k, V, E, \{g_e\}_{e \in E}, \{r_v\}_{v \in V}, \langle s_i, u_i \rangle_{i \in [k]} \rangle$, where k is the number of players; $\langle V, E, \{g_e\}_{e \in E} \rangle$ is a timed network; for $v \in V$, the cost function $r_v : [k] \rightarrow \mathbb{R}_{\geq 0}$ maps the load on vertex v , namely the number of players that simultaneously visit vertex v , to the cost each of them pays for staying in v for one time unit with this load; and for $i \in [k]$, the pair $\langle s_i, u_i \rangle \in V \times V$ describes the objective of Player i : choosing a timed path from s_i to u_i . A timed network game is *symmetric* if all the players have the same objective, i.e. the same source and target pair. We use $B(\mathcal{T})$ to denote the set of interval boundaries appearing in the guards of \mathcal{T} .

In order to satisfy her objective, Player i has to choose a path in \mathcal{T} from s_i to u_i as well as the duration spent in each vertex in the path. Indeed, while edges are traversed instantaneously, the guards on the edges force the players to spend time on vertices. Each player then aims to minimize the cost of these stays. In order to formally define the strategies of the players and their costs, we first need some definitions.

A *timed path* in the TNG \mathcal{T} is a sequence $\pi = \langle v_0, t_0 \rangle, \dots, \langle v_{n-1}, t_{n-1} \rangle, v_n \in (V \times \mathbb{R}_{\geq 0})^* \cdot V$, such that for all $0 \leq i < n$, we have that $\langle v_j, v_{j+1} \rangle \in E$; that is, v_0, \dots, v_n is a path in the graph $\langle V, E \rangle$. Intuitively, for all $0 \leq i < n$, we have that t_j describes the time spent in the vertex v_j before the path continues to v_{j+1} . Let $\tau_0 = t_0$ and $\tau_j = \tau_{j-1} + t_j$, for $0 < j < n$. Note that $\tau_j = \sum_{l=0}^j t_l$. Thus, τ_j is the time that has elapsed since the traversal of π starts and until π leaves the vertex v_j . We sometimes refer to π also as the sequence $\langle \tau_0, e_1 \rangle, \dots, \langle \tau_{n-1}, e_n \rangle \in (\mathbb{R}_{\geq 0} \times E)^*$, where for all $1 \leq j \leq n$, we have that $e_j = \langle v_{j-1}, v_j \rangle$ is the j -th edge in π and is taken at time τ_{j-1} . We say that the timed path π is *legal* if for all $0 \leq j < n$, we have that τ_j satisfies the guard $g_{e_{j+1}}$.

A *strategy* for a player with an objective $\langle s, u \rangle$ is a legal timed path $\pi = \langle v_0, t_0 \rangle, \dots, \langle v_{n-1}, t_{n-1} \rangle, v_n$ such that $v_0 = s$ and $v_n = u$. Consider a finite set $T \subseteq \mathbb{R}_{\geq 0}$ of time points. We say that the strategy π is a T -*strategy* if all edges in π are taken at times in T . Formally, for all $0 \leq j < n$, we have that $\tau_j \in T$. A *profile* is a tuple $P = \langle \pi_1, \dots, \pi_k \rangle$ of strategies for the players. That is, for $1 \leq i \leq k$, we have that π_i is a strategy for Player i . A profile is a T -*profile* if all its strategies are T -strategies. Of special interest are *boundary* strategies and profiles, namely T -strategies and T -profiles for $T = B(\mathcal{T})$. Note that each profile P has a finite minimal set $T \subseteq \mathbb{R}_{\geq 0}$ such that P is a T -profile. We denote this set by T_P .

Given $T \subseteq \mathbb{R}_{\geq 0}$, let $t_{\max} = \max(T)$. Also, for $t \in T$ such that $t \neq t_{\max}$, let $next_T(t)$ be the time point $t' \in T$ such that $t < t'$ and there is no $t'' \in T$ such that $t < t'' < t'$. That is, $next_T(t)$ is the time point successor to t in T . We can partition the interval $[0, t_{\max}]$ to a set Υ of sub-intervals $[m_1, m_2]$ such that m_1 and m_2 are in $T \cup \{0\}$, and $m_2 = next_T(m_1)$. We refer to the sub-intervals in Υ as *periods*. When T is T_P for some profile P , then the set Υ is the coarsest partition of $[0, t_{\max}]$ into periods such that no player crosses an edge within each period. We denote this partition by Υ_P .

Consider a T -profile P . For a player $i \in [k]$ and a period $\gamma \in \Upsilon_P$, let $visits_P(i, \gamma)$ be the vertex that Player i visits during period γ . That is, if $\pi_i = \langle v_0^i, t_0^i \rangle, \dots, \langle v_{n_i-1}^i, t_{n_i-1}^i \rangle, v_{n_i}^i$ is the legal timed path that is the strategy for Player i and $\gamma = [m_1, m_2]$, then $visits_P(i, \gamma)$ is the vertex v_j^i for the index $1 \leq j < n_i$ such that $\tau_{j-1}^i \leq m_1 \leq m_2 \leq \tau_j^i$. Note that since P is a T -profile, then for each period $\gamma = [m_1, m_2] \in \Upsilon_P$, the number of players that stay in each vertex v during γ is fixed. Let $load_P(v, \gamma)$ denote this number. Formally $load_P(v, \gamma) = |\{i : visits_P(i, \gamma) = v\}|$. Finally, for a period $\gamma = [m_1, m_2]$, let $|\gamma| = m_2 - m_1$ be the duration of γ .

Recall that the cost function $r_v : [k] \rightarrow \mathbb{R}_{\geq 0}$ maps the load of v to the cost of v per time unit. Accordingly, if $visits_P(i, \gamma) = v$, then the cost of Player i in P over the period γ is $cost_{\gamma,i}(P) = r_v(load_P(v, \gamma)) \cdot |\gamma|$. We distinguish between two types of cost functions. We say that in *uniform cost-sharing games* (CS-TNGs, for short), the players that visit a vertex share its cost equally. Formally, each vertex v is associated with a rate $b_v \in \mathbb{R}_{\geq 0}$, and for all $l \geq 1$, we have $r_v(l) = \frac{b_v}{l}$. Note that increasing the load in uniform cost-sharing games decreases the cost of the players. On the other hand, in *congestion games* (CON-TNGs, for short), the cost functions are non-decreasing, thus increasing the load also increases the cost for each player. The total cost of Player i in profile P is then $cost_i(P) = \sum_{\gamma \in \Upsilon_P} cost_{\gamma,i}(P)$. The cost of the profile P , denoted $cost(P)$, is the total cost incurred by all the players, i.e., $cost(P) = \sum_{i=1}^k cost_i(P)$.

Consider a TNG \mathcal{T} . For a profile P and a strategy π of player $i \in [k]$, let $P[i \leftarrow \pi]$ denote the profile obtained from P by replacing the strategy for Player i by π . A profile P is said to be a (*pure*) *Nash equilibrium* (NE) if none of the players in $[k]$ can benefit from a unilateral deviation from her strategy in P to another strategy. In other words, for every player i and every strategy π that Player i

can deviate to from her current strategy in P , it holds that $cost_i(P[i \leftarrow \pi]) \geq cost_i(P)$. The set of NEs of the game \mathcal{T} is denoted by $NE(\mathcal{T})$.

A *social optimum* (SO) of a game \mathcal{T} is a profile that attains the infimum cost over all profiles. We denote by $SO(\mathcal{T})$ the cost of an SO profile; i.e., $SO(\mathcal{T}) = \inf_P cost(P)$. Note that since a TNG may have infinitely many profiles, we should indeed take the infimum (rather than minimum) over all profiles, and thus, an SO profile may not exist. As we shall show, however, all TNGs have boundary SO profiles. An SO profile may be achieved by a centralized authority and need not be an NE. The following parameters measure the inefficiency caused as a result of the selfish interests of the players. First, the *price of stability* (PoS) [8] of a timed network game \mathcal{T} is the ratio between the infimum cost of an NE and the cost of a social optimum of \mathcal{T} . That is, $PoS(\mathcal{T}) = \inf_{P \in NE(\mathcal{T})} cost(P) / SO(\mathcal{T})$. Then, the *price of anarchy* (PoA) [35] of \mathcal{T} is the ratio between the supremum cost of an NE and the cost of a social optimum of \mathcal{T} . That is, $PoA(\mathcal{T}) = \sup_{P \in NE(\mathcal{T})} cost(P) / SO(\mathcal{T})$. Note that here too, we have to use infimum and supremum rather than minimum and maximum, yet we are going to show that best and worst NEs are always attained. For a family \mathcal{F} of games, we say that the PoA of \mathcal{F} is *at most* x if for all games F in \mathcal{F} , we have $PoA(F) \leq x$ and is *at least* x , if there exists a game F in \mathcal{F} such that $PoA(F) = x$, and similarly for PoS.

3 Reduction to and from Network Games

A *network game* (NG) is $\mathcal{N} = \langle k, V, E, \{l_e\}_{e \in E}, \langle s_i, u_i \rangle_{i \in [k]} \rangle$, and has a similar structure to a TNG. A strategy of a player $i \in [k]$ is a path from s_i to u_i . The cost function $l_e : [k] \rightarrow \mathbb{R}_{\geq 0}$ maps the load on edge e to the cost each player pays for using e . As is the case with TNGs, one can consider both cost-sharing (CS-NGs) and congestion (CON-NGs) network games. Consider a profile $P = \langle \sigma_1, \sigma_2, \dots, \sigma_k \rangle$ in the game. Since all the costs are positive, we can restrict attention to strategies in which the paths chosen by the players are simple. Then, we can also ignore the order between the edges in the paths and assume that for all $i \in [k]$, we have that $\sigma_i \subseteq E$ is a set of edges that composes a path from s_i to u_i .² For an edge $e \in E$, we denote by $load_P(e)$, the number of players that use the edge e in P . Each player that uses e then pays $l_e(load_P(e))$, and the cost of Player i in P is $\sum_{e \in \sigma_i} l_e(load_P(e))$.

Given an NG \mathcal{N} , a TNG \mathcal{T} and a finite set $T \subseteq \mathbb{R}_{\geq 0}$, we say that \mathcal{N} and \mathcal{T} are *isomorphic with respect to* T if \mathcal{N} and \mathcal{T} have the same number of players and there exists a 1-to-1 cost-preserving correspondence between the profiles in \mathcal{N} and the T -profiles in \mathcal{T} . Formally, there exists a bijection f from the set of T -profiles in \mathcal{T} and the profiles in \mathcal{N} such that for every T -profile P in \mathcal{T} and $i \in [k]$, the costs of Player i in P and $f(P)$ coincide.

NGs have been extensively studied. In this section, we show that once we fix a set $T \subseteq \mathbb{R}_{\geq 0}$ of time points, we can reduce a TNG \mathcal{T} with edges taken only at time points in T to an NG. Formally, we have the following.

► **Theorem 2.** *Given a TNG \mathcal{T} and a finite set $T \subseteq \mathbb{R}_{\geq 0}$, we can construct an NG \mathcal{N} such that \mathcal{N} and \mathcal{T} are isomorphic with respect to T . The size of \mathcal{N} is polynomial in the size of \mathcal{T} and T , and it is constructed in polynomial time.*

Proof. In TNGs, cost is associated with vertices and the time is spent in them, whereas in NGs, cost is associated with the edges and there is no reference to time. Thus, the construction translates the cost of staying in vertices during time intervals induced by T to the cost of traversing edges.

² Note that the assumptions on each edge being visited at most once in strategies in NGs does not apply to TNGs. Indeed, there, a player may benefit from visiting a vertex multiple times (see Example 1).

Consider a TNG $\mathcal{T} = \langle k, V, E, \{r_v\}_{v \in V}, \{g_e\}_{e \in E}, (s_i, u_i)_{i \in [k]} \rangle$ and the given set T . We assume that $0 \in T$. We construct an NG $\mathcal{N} = \langle k, V', E', \{l_e\}_{e \in E'}, ((s_i, 0), u_i)_{i \in [k]} \rangle$, where $V' \subseteq (V \times T) \cup \{u_i\}_{i \in [k]}$ and $E' \subseteq V' \times V'$ is defined as follows (See an example in the full version. For every vertex $v \in V$, we have the following edges in E' . Let $\tau_{\max} = \max(T)$.

1. For every $\tau \neq \tau_{\max} \in T$, let $\tau' = \text{next}_T(\tau)$. Then, the edge $e = ((v, \tau), (v, \tau'))$ is in E' , corresponding to players staying in vertex v during the interval $[\tau, \tau']$. Accordingly, the cost of e is such that for every $m \in [k]$, we have $l_e(m) = r_v(m)(\tau' - \tau)$.
2. For every $v' \neq v$ with $(v, v') \in E$ and $\tau \in T$ such that τ satisfies $g_{\langle v, v' \rangle}$, we have an edge $e = ((v, \tau), (v', \tau))$ in E' . This edge corresponds to the edge (v, v') in E . Recall that the cost of crossing an edge in a TNG is 0. Accordingly, the cost of e is such that for every $m \in [k]$, we have $l_e(m) = 0$.
3. If $v = u_i$ for some $i \in [k]$, then for all $\tau \in T$, we have an edge $e = ((v, \tau), v)$ in E' , with $l_{e'}(m) = 0$ for every $m \geq 1$. In \mathcal{N} , the target vertex for Player i is u_i .

It is easy to see that the size of \mathcal{N} is polynomial in \mathcal{T} and T . In the full version, we prove that \mathcal{N} and \mathcal{T} are indeed isomorphic with respect to T . That is, we show a bijection f from the set of T -profiles in \mathcal{T} and the profiles in \mathcal{N} such that for every T -profile P in \mathcal{T} and $i \in [k]$, the costs of Player i in P and $f(P)$ coincide. ◀

A reduction in the other direction, namely of NGs to TNGs, is not obvious, as the dynamic of TNGs requires a synchronization among all the traversals in each of the edges. We illustrate this in the full version of the paper. When, however, the NG is acyclic, we can use a topological ordering on the edges and synchronize the traversals. Intuitively, each edge in the NG induces a vertex in the TNG, and the guards are defined so that the vertex associated with the j -th edge in the topological order is visited during the period $[j - 1, j]$. This can be easily forced by guarding the edges entering the vertex by $[j - 1, j - 1]$ and guarding these that leave it by $[j, j]$. See the full version for the proof.

► **Theorem 3.** *Given an acyclic NG \mathcal{N} , we can construct in polynomial time a TNG \mathcal{T} that is isomorphic to \mathcal{N} with respect to $B(\mathcal{T}) \cup \{0\}$. The size of \mathcal{T} is polynomial in the size of \mathcal{N} .*

4 On Boundary Strategies and Profiles

Since a strategy for a player in a TNG is a timed path with time points in $\mathbb{R}_{\geq 0}$, then each player has uncountably many possible strategies, and hence it is possible to have uncountably many profiles. In NGs, a strategy is a non-timed path from the source to the target. Even there, in the non-timed setting, there may be infinitely many paths from the source to the target. It is easy to see, however, that every strategy that is a non-simple path is dominated by the strategy obtained by removing cycles, and thus one can restrict attention to the finitely many profiles that consist of strategies that are simple paths. Our goal in this section is to examine whether some similar restriction can be made in TNGs. Indeed, being able to restrict attention to finitely many profiles would simplify our understanding of TNGs and their analysis. A natural candidate is a restriction to boundary strategies, namely these in which all edges are taken at interval boundaries. We show that while a boundary NE exists in all TNGs, and that all TNGs have a boundary SO, there may be uncountably many NEs that are not boundary. Moreover, there are TNGs in which the best and worst NEs are not boundary.

We first need the following lemma.

► **Lemma 4.** *Consider a TNG \mathcal{T} and a finite set $T \subset \mathbb{R}_{\geq 0}$ such that $B(\mathcal{T}) \subseteq T$. Let π_1, \dots, π_{k-1} be T -strategies of players $1, \dots, k - 1$ respectively. There exists a T -strategy π_k of Player k such that for every strategy π'_k of Player k that is not a T -strategy, we have $\text{cost}_k(\langle \pi_1, \dots, \pi_{k-1}, \pi_k \rangle) \leq \text{cost}_k(\langle \pi_1, \dots, \pi_{k-1}, \pi'_k \rangle)$.*

Intuitively, Lemma 4 states that if all players but one use boundary strategies, then a best strategy for the k -th player can also be a boundary one. It implies that when we want to prove that a certain boundary profile is an NE, we can restrict attention to deviations that use boundary strategies.

► **Theorem 5.** *All TNGs have a boundary NE. Moreover, from every profile P , there is a sequence of best-response moves that converges to an NE. When P is boundary, so is the obtained NE.*

Proof. Given a TNG \mathcal{T} , let \mathcal{N} be an NG that is isomorphic to \mathcal{T} with respect to $B(\mathcal{T})$. Let f be the bijection from the set of profiles in \mathcal{N} to the set of $B(\mathcal{T})$ -profiles in \mathcal{T} such that for every profile P in \mathcal{N} and $i \in [k]$, the costs of Player i in P and $f(P)$ coincide. By Theorem 2, such an NG and bijection f exist. By [8, 25, 38], all NGs have an NE. Consider an NE $P_{\mathcal{N}}$ in \mathcal{N} . In the full version we prove that $f(P_{\mathcal{N}})$ is an NE in \mathcal{T} .

For the second claim, the above considerations also imply that starting from a profile P , we can restrict attention to best-response moves in which edges are taken in time points in $T_P \cup B(\mathcal{T})$, and reach the desired NE. In particular, when P is boundary, so is the obtained NE. ◀

Recall that an SO profile attains the infimum cost over all profiles. We now show that an SO profile always exists, and in fact there always exists a boundary SO. We show that a boundary SO profile always exists. The idea is that if, in a profile P , an edge e is taken at a non-boundary time τ , then it is possible to obtain a profile P' in which e is taken at a boundary time and $cost(P') \leq cost(P)$. We formalize this intuition in the full version.

► **Theorem 6.** *All TNGs have a boundary SO.*

We proceed to show that there are non-trivial TNGs that have uncountably many NEs, which implies they also have uncountably many non-boundary NEs.

► **Theorem 7.** *There exist CS-TNGs and CON-TNGs that have uncountably many NEs.*

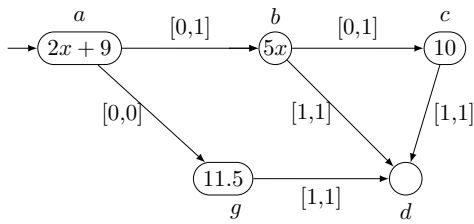
Proof. The CS-TNG from Example 1 has uncountably many NEs. In the full version, we present and analyze in detail a different CS-TNG with uncountably many NEs.

We continue to CON-TNGs. Consider the TNG appearing in Figure 1. The objectives of Players 1 and 2 is $\langle a, d \rangle$ and the objectives of Players 3 and 4 is $\langle b, d \rangle$. The cost functions are written in the vertices. For $y \in [0, 0.5]$, let P_y be the profile in which Players 1 and 2 traverse the edge $\langle a, b \rangle$, and Players 3 and 4 traverse the edge $\langle b, c \rangle$, all at time y . In the full version, we prove that for every $y \in [0, 0.5]$, the profile P_y is an NE. Since y can have any value in $[0, 0.5]$, we are done. ◀

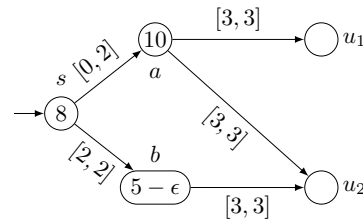
Theorem 7 suggests that the values of a best and worst NEs should be defined by means of infimum and supremum, and may not be attained. In the full version we prove that best and worst NEs do exist. Essentially, it follows from the fact that our guards are closed intervals, implying that the time points in an NE should satisfy a system of inequalities with no strict inequalities. As bad news, we now show that while a boundary NE always exists, the best and worst NEs need not be boundary.

► **Theorem 8.** *There exists a CS-TNG in which the best NE is not a boundary profile.*

Proof. Consider the two-player TNG \mathcal{N} that is played on the network depicted in Figure 2. The objective of Player i is $\langle s, u_i \rangle$. Player 1 has two boundary strategies: A , in which she traverses the edge $\langle s, a \rangle$ at time 0, and B , in which she takes it at time 2. Note that the suffixes of the strategies are fixed, as Player 1 must traverse the edge $\langle a, u_1 \rangle$ at time 3. Player 2 has three boundary strategies: Strategies A and B , in which she traverses edge $\langle s, a \rangle$ at time 0 and 2, respectively, and Strategy C , in which she traverses the edge $\langle s, b \rangle$ at time 2. Again, the suffixes of the strategies A and B are fixed. In the full version, we prove that $\langle A, A \rangle$ and $\langle B, C \rangle$ are the only boundary NEs with $cost(\langle A, A \rangle) = 30$ and $cost(\langle B, C \rangle) = 31 - \epsilon$.



■ **Figure 1** A CON-TNG in which the worst NE is not boundary.



■ **Figure 2** A CS-TNG in which the best NE is not boundary.

For $x \in (0, 2)$, let P_x be the profile in which both players traverse the edge $\langle s, a \rangle$ at time $2 - x$. In the full version, we show that we can define ϵ and x so that P_x is an NE with $cost(P_x) < \min\{cost(\langle A, A \rangle), cost(\langle B, C \rangle)\}$. For example, by taking $x = 0.25$ and $\epsilon = 0.5$ we get an NE with $cost(P_x) = 26.5$ ◀

► **Theorem 9.** *There exists a CS-TNG in which the worst NE is not a boundary profile.*

Proof. Consider the two-player TNG \mathcal{N} that is played on the network depicted in Figure 3. The objective of Player i is $\langle s, u_i \rangle$. Player 1 has three boundary strategies: A , in which she traverses the edge (v_1, v_2) at time 0; B , in which she takes it at time 2; and D , in which she traverses the edge (v_1, v_5) at time 2.

Player 2 has four boundary strategies: A , in which she traverses edge (v_1, v_2) at time 0; B , where she takes (v_1, v_2) at time 2; C , where she traverses the edge (v_1, v_4) at time 2; and E , where she traverses the edge (s, v_3) . Note that strategy E has a fixed cost of 13.2.

In the full version, we prove that the only boundary profile that is an NE is the profile $\langle D, C \rangle$, whose cost is 26.3, and that the non-boundary profile $P_{0.2}$ in which Players 1 and 2 traverse the edge (v_1, v_2) together at time 1.8 is an NE with cost 26.4, which is higher than $cost(\langle D, C \rangle)$. ◀

► **Theorem 10.** *There exists a CON-TNG in which the worst NE is not a boundary profile.*

Proof. Recall the CON-TNG presented in Figure 1. In the proof of Theorem 7, we proved that for all $0 \leq y \leq 0.5$, the profile P_y , in which Players 1 and 2 traverse the edge (a, b) and Players 3 and 4 traverse the edge (b, c) , all at time y , is an NE. We have $cost_1(P_y) = cost_2(P_y) = 13y + 10 \cdot (1 - y) = 3y + 10$, whereas $cost_3(P_y) = cost_4(P_y) = 10y + 10 \cdot (1 - y) = 10$. Thus $cost(P_y) = 6y + 40$.

Players 1 and 2 have three boundary strategies: A , in which they traverse the edge (a, b) at time 0; B , in which they traverse the edge (a, b) at time 1; and C , in which they traverse the edge (a, g) at time 0. Players 3 and 4 have three boundary strategies: D , in which they traverse the edge (b, c) at time 0, and E , in which they traverse the edge (b, c) at time 1, and F , in which they traverse the edge (b, d) at time 1.

In the full version, we show that the boundary NEs with the highest cost are $\langle C, B, E, E \rangle$ and $\langle C, B, F, F \rangle$ having a cost of 42.5. The cost of the profile $P_{0.5}$ is $6 \cdot 0.5 + 40 = 43$. This implies that the worst NE in the CON-TNG in Figure 1 is a non-boundary profile. ◀

We note that it might appear that whenever there exists a non-boundary NE in a TNG \mathcal{T} , there exist uncountably many NEs in \mathcal{T} . This, however, is not the case as can be seen in the TNG in Figure 3. As argued in the proof of Theorem 9, this TNG has only one non-boundary NE. We also note that while we showed that the best and worst NEs in a CS-TNG need not be boundary, for congestion games we only showed that the worst NE need not be boundary. Thus, the problem of whether there is a CON-TNG in which the best NE is not boundary is left open.

5 Equilibrium Inefficiency

As discussed in Section 1, decentralized decision-making often leads to solutions that are sub-optimal from the point of view of the society as a whole. Recall that the measures PoS and PoA measure the inefficiency caused by the selfish behavior of the players. It refers to the ratio between the best (PoS) and worst (PoA) NEs and the SO. In this section we discuss these measures for TNGs. For NGs, the PoS and PoA are well understood. In order to use Theorem 2 and apply the results known for NGs to TNGs, we need to find a set of time points with respect to which the models are isomorphic. As discussed in Section 4, the natural candidate for this is the set of interval boundaries. While, however, we can restrict attention to boundary strategies when we consider the SO, such a restriction is not sound when we consider the infimum and supremum values of NEs. We show that our results in Theorem 2 and Section 4 do imply the required upper bounds, and that the lower bounds known for NGs can be extended to TNGs by carefully revising the examples known there.

► **Theorem 11.** *The PoS and PoA for TNGs are upper-bounded by these for NGs. Thus, for CS-TNGs with k players, the PoS is at most $\log k$ and the PoA is at most k . For CON-TNGs with affine cost functions, the PoS is at most 1.577 and the PoA is at most $\frac{5}{2}$.*

Proof. Consider a TNG \mathcal{T} . Let P be an NE in \mathcal{T} and let \mathcal{N}_P be the NG isomorphic to \mathcal{T} with respect to $B(\mathcal{T}) \cup T_P$. Let f be a cost preserving bijection from the $(B(\mathcal{T}) \cup T_P)$ -profiles of \mathcal{T} and these of \mathcal{N}_P . As argued in the proof of Theorem 5, the profile $f(P)$ is an NE in \mathcal{N}_P . It follows that the cost of an NE in \mathcal{T} is upper and lower bounded by the cost of an NE in an NG. Also, by Theorem 6, there exists a boundary SO in \mathcal{T} , which, by Theorem 2, corresponds to an SO in \mathcal{N} . Thus, the ratio between an NE in \mathcal{T} and the cost of its SO is upper and lower bounded by this ratio in an NG. Since the above holds for all TNGs, we are done. ◀

Adopting the lower bounds on PoS and PoA from NGs to TNGs is more difficult, as the reduction from NGs to TNGs can be applied only to acyclic NGs. Fortunately, for CS-NGs, matching lower bounds have been proven for acyclic networks. Hence, using considerations that are similar to these in the proof of Theorem 11 (in fact, simpler ones, as there is no need to refer to T_P), we can use the reduction described in Theorem 3 in order to conclude the following.

► **Theorem 12.** *The PoS and PoA for TNGs are lower-bounded by these for acyclic NGs. Thus, for CS-TNGs with k players, the PoS is at least $\log k$ and the PoA is at least k .*

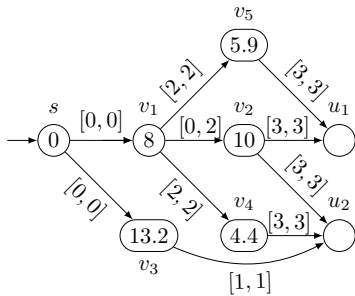
For CON-TNG, the adoption of results from CON-NGs is more challenging, as known lower bounds use cyclic network. We are still able to prove a lower bound for the PoA. A bound for CON-NGs with linear cost function has been shown in [22]. In our case, we show that the upper bound is matched asymptotically as we increase the number of players k , where $k \geq 3$.

► **Theorem 13.** *There are CON-TNGs with linear cost functions such that for $k=3$ or more players, the PoA is $\frac{5 \cdot k}{2(k-2)+3+5}$. Hence as $k \rightarrow \infty$, the PoA approaches $\frac{5}{2}$.*

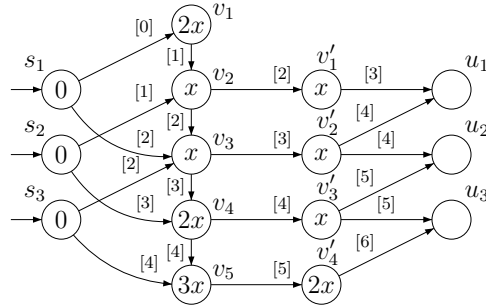
Proof. Consider the three-player CON-TNG appearing in Figure 4. The sources and targets of the three players are s_1, s_2, s_3 and u_1, u_2, u_3 , respectively. The cost of staying in the source vertices is 0. For the rest of the vertices, the cost functions are as follows: $r_{v_1}(x) = r_{v_4}(x) = 2x$, $r_{v_2}(x) = r_{v_3}(x) = x$, $r_{v_5}(x) = 3x$, $r_{v'_1}(x) = r_{v'_2}(x) = r_{v'_3}(x) = x$, and $r_{v'_4}(x) = 2x$.

Consider the profile P in which Player i , for all $i \in \{1, 2, 3\}$, visits the vertices v_i, v_{i+1}, v'_i . The profile P is an NE in which each player pays 5, so $\text{cost}(P) = 15$. However, an SO is obtained when each Player i moves from her source to target through vertices v_{i+2}, v'_{i+1} . In this profile, the costs of the players are 2, 3, and 5. Thus $\text{PoA} = \frac{15}{2+3+5} = \frac{3}{2}$.

If there are k players, we consider the game with vertices v_1, \dots, v_{k+2} and vertices v'_1, \dots, v'_{k+1} . The cost functions are $r_{v_1}(x) = r_{v_{k+1}}(x) = r_{v'_{k+1}}(x) = 2x$, $r_{v_{k+2}}(x) = 3x$, while for the remaining vertices v apart from the source and the target vertices, $r_v(x) = x$. The PoA is $\frac{5 \cdot k}{2(k-2)+3+5}$. Hence, PoA asymptotically reaches its upper bound as k tends to ∞ . ◀



■ **Figure 3** A CS-TNG in which the worst NE is not boundary.



■ **Figure 4** A lower bound of PoA = $\frac{5 \cdot k}{2(k-2)+3+5}$ for CON-TNGs.

6 The Complexity of Finding an NE

The complexity class PLS contains local search problems with polynomial time searchable neighborhoods [30]. Essentially, a problem is in PLS if there is a set of feasible solutions for it such that it is possible to find, in polynomial time, an initial feasible solution and then iteratively improve it, with each improvement being performed in polynomial time, until a local optimum is reached. See the full version for the formal definition.

In this section we prove that the problem of finding an NE is PLS-complete for TNGs, which coincides with the complexity bounds for NGs [25, 41]. Proving membership in PLS would follow easily from the reduction to NGs. Proving hardness is more involved: While for CON-TNGs we are able to rely on previous results, corresponding to CS-TNGs, we first solve the problem for acyclic CS-NGs. We start with the upper bound.

▶ **Theorem 14.** *The problem of finding an NE in CS-TNGs and CON-TNGs is in PLS. For symmetric TNGs, the problem can be solved in polynomial time.*

Proof. For membership in PLS, we describe an algorithm to find an NE. Consider a TNG \mathcal{T} , and let \mathcal{N} be the isomorphic NG with respect to $B(\mathcal{T})$. Recall that the size of \mathcal{N} is polynomial in the size of \mathcal{T} . We run the PLS algorithm for finding an NE P in \mathcal{N} . As in Theorem 5, the profile $f^{-1}(P)$ is an NE in \mathcal{T} , thus we are done. When \mathcal{T} is symmetric, so is \mathcal{N} . Since finding an NE in a symmetric NG can be done in polynomial time [25], the claim follows. ◀

For PLS-hardness, we describe a reduction from the problem of finding a local MAX CUT in a weighted network (LMC, for short) which is known to be PLS-complete [40]. In [1], a polynomial-time reduction is shown from the LMC problem to the problem of finding an NE in CON-NGs. The reduction involves two steps: from the LMC problem to the problem of finding an NE in a class of games called *quadratic threshold games*, which in turn is reduced to the problem of finding an NE in a CON-NG. The reduction in [1] always produces an acyclic CON-NG. By Theorem 3, the latter can be reduced to an isomorphic CON-TNG. In order to use a similar technique for CS-TNGs, we first establish PLS-hardness for acyclic CS-NGs, which is an open problem. The proof uses a non-trivial reduction from the LMC problem and can be found in the full version.

► **Theorem 15.** *The problem of finding an NE in acyclic CS-NGs is PLS-hard.*

We thus have a matching lower bound also for CS-TNGs leading to the following theorem.

► **Theorem 16.** *The problems of finding an NE in CS-TNGs and CON-TNGs are PLS-complete.*

7 Discussion and Directions for Future Research

We introduced and studied timed network games, which are an extension of network games with real-time considerations. TNGs are inspired by *timed automata* [5], which are automata extended by a finite set of *clocks*. A clock is a variable that takes values in $\mathbb{R}_{\geq 0}$ and whose values increase as time passes. In the full version we study *TNGs with clocks*, in which, as in timed automata, transitions are labeled by constraints on the clocks and clocks may be *reset* when traversing a transition. For example, if we reset a clock x when we enter a vertex v , then a guard $x \leq 5$ in a transition that leaves v , bounds the time spent in v to be at most 5 time units. The TNGs we study here are equivalent to a model with clocks that are never reset. Indeed, then, all clocks maintain the time that has passed since the start of the game, and guards impose bounds on this time. TNGs with clocks are already interesting in the degenerate case when there is only one player, a.k.a. *priced timed automata* (PTA, for short) [7, 15].

We describe here briefly our results for TNGs with clocks. Clearly, the negative results we obtain here for TNGs without clocks follow to the general setting. Recall that a main tool for obtaining positive results is a reduction between TNGs and NGs. The key to such a reduction is a partition of $\mathbb{R}_{\geq 0}$ into finitely many intervals, which involves two questions: about the granularity to which we have to partition $\mathbb{R}_{\geq 0}$, and about the maximal point in time that is of interest. While the answer to the first question is not difficult also for TNGs with clocks, the answer to the second question is difficult and interesting in its own right. Our positive results are not obtained using such a reduction. In order to prove the existence of an NE in every TNG with clocks, we show that such games are potential games and we also find a lower bound on the decrease in potential in a best response. Note that only showing that TNGs with clocks are potential games does not suffice to prove existence of an NE as there are infinitely many profiles. We then turn to study computational-complexity problems and show that the best-response problem is PSPACE-complete, which matches the complexity of *cost optimal reachability* in PTAs [?]. Finally, we address the question above; namely, we find bounds on the minimal time at which the players reach their destinations in an NE and an SO.

This work belongs to a line of works that transfer concepts and ideas between the areas of formal verification and algorithmic game theory: logics for specifying multi-agent systems [6, 20], studies of equilibria in games related to synthesis and repair problems [19, 18, 26, 4], and of non-zero-sum games in formal verification [21, 17]. This line of work also includes an extension of NGs to objectives that are richer than reachability [14], NGs in which the players select their paths dynamically [11], and efficient reasoning about NGs with huge networks [33, 10].

Additional extensions of TNGs that we plan to study are the following: (1) Richer objectives, where the vertices of the TNG are labeled by letters from an alphabet, allowing objectives that describe on-going behaviors [14]. For example, an objective may require each visit to vertex labeled *send* to be preceded by a vertex labeled *encode*. (2) A *dynamic* choice of paths, where strategies do not specify the full path but rather map prefixes of paths of all players to the next move [11]. For example, when the network models a network of roads and the players are drivers, it makes sense to allow drivers to observe the congestion in the network when reaching a junction (vertex) before choosing the next road (edge) in their path. (3) A *global-cost* mechanism, in which the load on a resource refers to the total time for which it is used, rather than to particular time instants.

References

- 1 H. Ackermann, H. Röglin, and B. Vöcking. On the impact of combinatorial structure on congestion games. *J. ACM*, 55(6):25:1–25:22, December 2008.
- 2 S. Agarwal, M. S. Kodialam, and T. V. Lakshman. Traffic engineering in software defined networks. In *Proc. 32nd IEEE International Conference on Computer Communications*, pages 2211–2219, 2013.
- 3 S. Aland, D. Dumrauf, M. Gairing, B. Monien, and F. Schoppmann. Exact price of anarchy for polynomial congestion games. *SIAM J. Comput.*, 40(5):1211–1233, 2011.
- 4 S. Almagor, G. Avni, and O. Kupferman. Repairing multi-player games. In *Proc. 26th Int. Conf. on Concurrency Theory*, volume 42 of *LIPICs*, pages 325–339, 2015.
- 5 R. Alur and D. Dill. A theory of timed automata. *Theoretical Computer Science*, 126(2):183–236, 1994.
- 6 R. Alur, T.A. Henzinger, and O. Kupferman. Alternating-time temporal logic. *Journal of the ACM*, 49(5):672–713, 2002.
- 7 R. Alur, S. La Torre, and G. J. Pappas. Optimal paths in weighted timed automata. *Theoretical Computer Science*, 318(3):297–322, June 2004.
- 8 E. Anshelevich, A. Dasgupta, J. Kleinberg, E. Tardos, T. Wexler, and T. Roughgarden. The price of stability for network design with fair cost allocation. In *Proc. 45th IEEE Symp. on Foundations of Computer Science*, pages 295–304. IEEE Computer Society, 2004.
- 9 E. Anshelevich, A. Dasgupta, J. Kleinberg, E. Tardos, T. Wexler, and T. Roughgarden. The price of stability for network design with fair cost allocation. *SIAM J. Comput.*, 38(4):1602–1623, 2008.
- 10 G. Avni, S. Guha, and O. Kupferman. An abstraction-refinement methodology for reasoning about network games. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, 2017*.
- 11 G. Avni, T.A. Henzinger, and O. Kupferman. Dynamic resource allocation games. In *Proc. 9th International Symposium on Algorithmic Game Theory*, volume 9928 of *Lecture Notes in Computer Science*, pages 153–166, 2016.
- 12 G. Avni, O. Kupferman, and T. Tamir. Network-formation games with regular objectives. *Information and Computation*, 251:165–178, 2016.
- 13 G. Behrmann, A. A. Fehnker, T. Hune, K.G. Larsen, P. Pettersson, J. Romijn, and F. W. Vaandrager. Minimum-cost reachability for priced timed automata. pages 147–161, London, UK, 2001. Springer-Verlag.
- 14 U. Bhaskar, L. Fleischer, and E. Anshelevich. A stackelberg strategy for routing flow over time. In *Proc. of SODA*, pages 192–201, 2011.
- 15 P. Bouyer, T. Brihaye, V. Bruyère, and J-F. Raskin. On the optimal reachability problem of weighted timed automata. *Formal Methods in System Design*, 31(2):135–175, October 2007.
- 16 T. Brihaye, V. Bruyère, J. De Pril, and H. Gimbert. On subgame perfection in quantitative reachability games. *Logical Methods in Computer Science*, 9(1), 2012.
- 17 K. Chatterjee. Nash equilibrium for upward-closed objectives. In *Proc. 15th Annual Conf. of the European Association for Computer Science Logic*, volume 4207 of *Lecture Notes in Computer Science*, pages 271–286. Springer, 2006.
- 18 K. Chatterjee, T. A. Henzinger, and M. Jurdzinski. Games with secure equilibria. *Theoretical Computer Science*, 365(1-2):67–82, 2006.
- 19 K. Chatterjee, T. A. Henzinger, and N. Piterman. Strategy logic. In *Proc. 18th Int. Conf. on Concurrency Theory*, pages 59–73, 2007.
- 20 K. Chatterjee, R. Majumdar, and M. Jurdzinski. On Nash equilibria in stochastic games. In *Proc. 13th Annual Conf. of the European Association for Computer Science Logic*, volume 3210 of *Lecture Notes in Computer Science*, pages 26–40. Springer, 2004.
- 21 G. Christodoulou and E. Koutsoupias. On the price of anarchy and stability of correlated equilibria of linear congestion games. In *ESA*, pages 59–70, 2005.

- 22 G. Christodoulou and E. Koutsoupias. The price of anarchy of finite congestion games. In *Proc. 37th ACM Symp. on Theory of Computing*, pages 67–73, 2005.
- 23 E.A. Emerson, A.K. Mok, A.P. Sistla, and J. Srinivasan. Quantitative temporal reasoning. In *Proc. 2nd Int. Conf. on Computer Aided Verification*, volume 531 of *Lecture Notes in Computer Science*, pages 136–145. Springer, 1990.
- 24 A. Fabrikant, C. Papadimitriou, and K. Talwar. The complexity of pure nash equilibria. In *Proc. 36th ACM Symp. on Theory of Computing*, pages 604–612, 2004.
- 25 D. Fisman, O. Kupferman, and Y. Lustig. Rational synthesis. In *Proc. 16th Int. Conf. on Tools and Algorithms for the Construction and Analysis of Systems*, volume 6015 of *Lecture Notes in Computer Science*, pages 190–204. Springer, 2010.
- 26 L.R. Ford and D.R. Fulkerson. *Flows in networks*. Princeton Univ. Press, Princeton, 1962.
- 27 T.A. Henzinger, Z. Manna, and A. Pnueli. What good are digital clocks? In *Proc. 19th Int. Colloq. on Automata, Languages, and Programming*, volume 623 of *Lecture Notes in Computer Science*, pages 545–558. Springer, 1992.
- 28 M. Hoefer, V. S. Mirrokni, H. Röglin, and S. Teng. Competitive routing over time. *Theor. Comput. Sci.*, 412(39):5420–5432, 2011. doi:10.1016/j.tcs.2011.05.055.
- 29 D. S. Johnson, C. H. Papadimitriou, and M. Yannakakis. How easy is local search? *Journal of Computer and Systems Science*, 37(1):79–100, August 1988.
- 30 E. Koutsoupias and C. Papadimitriou. Worst-case equilibria. *Computer Science Review*, 3(2):65–69, 2009.
- 31 E. Koutsoupias and K. Papakonstantinou. Contention issues in congestion games. In *Proceedings of the 39th International Colloquium Conference on Automata, Languages, and Programming - Volume Part II, ICALP' 12*, pages 623–635. Springer-Verlag, 2012.
- 32 O. Kupferman and T. Tamir. Hierarchical network formation games. In *Proc. 23rd Int. Conf. on Tools and Algorithms for the Construction and Analysis of Systems*, Lecture Notes in Computer Science. Springer, 2017.
- 33 B. P. Lathi and R. Green. *Essentials of digital signal processing*. Cambridge University Press, 2014.
- 34 C. H. Papadimitriou. Algorithms, games, and the internet. In *Proc. 33rd ACM Symp. on Theory of Computing*, pages 749–753, 2001.
- 35 M. Penn, M. Polukarov, and M. Tennenholtz. Random order congestion games. *Mathematics of Operations Research*, 34(3):706–725, 2009.
- 36 K. Ronald and S. Martin. Nash equilibria and the price of anarchy for flows over time. *Theoretical Computer Science*, 49(1):71–97, 2011.
- 37 R. W. Rosenthal. A class of games possessing pure-strategy Nash equilibria. *International Journal of Game Theory*, 2:65–67, 1973.
- 38 T. Roughgarden and E. Tardos. How bad is selfish routing? *Journal of the ACM*, 49(2):236–259, 2002.
- 39 A. A. Schäffer and M. Yannakakis. Simple local search problems that are hard to solve. *SIAM J. Comput.*, 20(1):56–87, 1991.
- 40 V. Syrgkanis. The complexity of equilibria in cost sharing games. In *WINE*, volume 6484 of *Lecture Notes in Computer Science*, pages 366–377. Springer, 2010.
- 41 W. S. Vickrey. Congestion theory and transport investment. *The American Economic Review*, 59(2):251–260, 1969.
- 42 S. Yagar. Dynamic traffic assignment by individual path minimization and queuing. *Transportation Research*, 5(3):179–196, 1971.