$\mathcal{NP}$-hardness of pure Nash equilibrium in Scheduling and Connection Games

Nguyen Kim Thang

LIX, Ecole Polytechnique, France.
thang@lix.polytechnique.fr.

Abstract. We prove $\mathcal{NP}$-hardness of pure Nash equilibrium for some problems of scheduling games and connection games. The technique is standard: first, we construct a gadget without the desired property and then embed it to a larger game which encodes a $\mathcal{NP}$-hard problem in order to prove the complexity of the desired property in a game. This technique is very efficient in proving $\mathcal{NP}$-hardness for deciding the existence of Nash equilibria. In the paper, we illustrate the efficiency of the technique in proving the $\mathcal{NP}$-hardness of deciding the existence of pure Nash equilibria of Matrix Scheduling Games and Weighted Connection Games. Moreover, using the technique, we can settle the complexity not only of the existence of equilibrium but also of the existence of good cost-sharing protocol.

Key words: Nash equilibrium, $\mathcal{NP}$-hardness.

1 Introduction

Considering the growth of Internet as a motivation, computer scientists are interested in studying Algorithmic Game Theory, in which one of the most well-studied objects is the stable state, called Nash equilibrium. Given a game with strategy sets for players, a *pure Nash equilibrium* is a strategy profile in which each player deterministically plays her chosen strategy and no one has an incentive to unilaterally change her strategy. A *mixed Nash equilibrium* is similar to the pure one except that now players can pick a randomized strategy – a probability distribution over their strategy sets. In 1951, Nash proved that every game with a finite number of players, each having a finite set of strategies, always possesses a mixed Nash equilibrium. However, no similar result exists for pure Nash equilibrium.

Rosenthal [14], Monderer and Shapley [12] introduced potential games which always possess a pure Nash equilibrium, for example: Congestion Games [12], Connection Games [13, chapter 19]. In these games, the existence of pure Nash equilibrium is proved by using a potential-function argument. The complexity of finding a pure equilibrium of Congestion Games is settled in [10]. Besides, it is proved that finding a pure Nash equilibrium is $\mathcal{NP}$-hard in some other games, for example games with imperfect information, perfect recall [3].

We are interested in the complexity of some properties of pure Nash equilibria. Until now there are two methods, in general, to prove the $\mathcal{NP}$-hardness of a problem: using gadgets or using the PCP Theorem. In the paper, we use a technique based on the former, specifically on the gadgets called *negated* and polynomial-time reductions. The technique is the following. First, find a *negated* gadget which does not have the desired property. (In fact, a negated gadget is a counter-example of the property.) Next, construct a family of games which encode a $\mathcal{NP}$-hard problem, and embed the gadget into. We argue that the game has the desired property if and only if there is a solution for the instance of $\mathcal{NP}$-hard problem by using the gadget to enforce players' behaviors in such a way that the game possesses the desired property.

This standard technique is successfully applied in settling the complexity on the existence of pure Nash equilibrium in games [8, 7]. In this paper, we present the technique as a framework and illustrate its application in different contexts. Interestingly, this technique is not only applied to the existence of equilibrium but also to other properties such as good cost-sharing protocol in Connection Games.

Note that, from now we use the term Nash equilibrium instead of pure Nash equilibrium. The paper is organized as follows. In Section 2, we introduce the Matrix Scheduling Games and prove the complexity of the existence of pure Nash equilibrium in this game. In Section 3, we prove the complexity of the existence of pure equilibrium in Weighted Connection Games – that answers a question in [4]. Moreover, in Connection Games we show the intractability of finding a fair cost-sharing protocol which induces an equilibrium that is not too far from the optimum.

2 Matrix Scheduling Games

In Matrix Scheduling Games, there are m machines, n players and a load matrix $(A_{ij})_{n\times m}$ where $A_{ij} \geq 0\ \forall i, j$. Each player has a set of jobs that need to be executed. Players can complete their jobs by choosing a subset of machines on which their jobs will be executed, i.e., the strategy set $\mathcal{S}_i$ of player p_i is a collection of subsets of machines. A_{ij} represents the load contribution of player p_i to machine m_j if she uses this machine. Given a strategy profile $S = (S_1, S_2, \ldots, S_n) \in (\mathcal{S}_1 \times \mathcal{S}_2 \times \ldots \times \mathcal{S}_n)$, the *load* ℓ_j of a machine m_j depends on the set of jobs executed on the machine and is defined as $\sum A_{ij}$ over all players p_i choosing a strategy (a subset of machines) which contains machine m_j. The *cost* of a player is the sum of all loads of machines that the player uses, i.e., $c_i = \sum_{m_j \in S_i} \ell_j$. Players are selfish and they choose a strategy which induces the cost as small as possible. Remark that, without loss of generality, the strategy set of a player is *inclusion-free*, i.e., no player possesses two strategies S and S' such that $S \subset S'$ since otherwise the player always prefer use S to S' in order to get a smaller cost.

If players' strategies are restricted to singleton machines then the game becomes the well-studied Load Balancing Games. There always exists Nash equilibrium on that game [9] (by using an argument of lexicographical potential function). However, without assumption on players' strategy sets, the game does not necessarily possess an equilibrium.

Lemma 1. *There exists a matrix scheduling game in which there is no Nash equilibrium.*

Proof: We describe the game with no Nash equilibrium. There are 4 machines and 3 players, each one has two strategies. The strategy sets of players p_1, p_2 and p_3 are $\mathcal{S}_1 = \{s_1^1 = (m_1, m_3); s_1^2 = (m_4)\}, \mathcal{S}_2 = \{s_2^1 = (m_1); s_2^2 = (m_2)\}$ and $\mathcal{S}_3 = \{s_3^1 = (m_2); s_3^2 = (m_3)\}$, respectively. The load matrix is given as in Figure 1. ($A_{ij} = \infty$ if it is not explicitly given.)

	m_1	m_2	m_3	m_4
p_1	2		10	15
p_2	5	4		
p_3		4	1	

Fig. 1. A matrix scheduling game with no Nash equilibrium.

p_1	p_2	p_3	Best reponse
s_1^1	s_2^1	s_3^1	$p_1 : s_1^1 \to s_1^2$ $(17 \to 15)$
s_1^1	s_2^1	s_3^2	$p_3 : s_3^2 \to s_3^1$ $(11 \to 4)$
s_1^1	s_2^2	s_3^1	$p_2 : s_2^2 \to s_2^1$ $(8 \to 7)$
s_1^1	s_2^2	s_3^2	$p_3 : s_3^2 \to s_3^1$ $(11 \to 8)$
s_1^2	s_2^1	s_3^1	$p_3 : s_3^1 \to s_3^2$ $(4 \to 1)$
s_1^2	s_2^1	s_3^2	$p_2 : s_2^1 \to s_2^2$ $(5 \to 4)$
s_1^2	s_2^2	s_3^1	$p_3 : s_3^1 \to s_3^2$ $(8 \to 1)$
s_1^2	s_2^2	s_3^2	$p_1 : s_1^2 \to s_1^1$ $(15 \to 13)$

Fig. 2. There is no stable strategy profile.

We prove that there is no Nash equilibrium in the game by verifying all 2^3 strategy profils. In Figure 2, the first three columns represent the strategies chosen by the players. The last column shows which player is unhappy and how she can decrease her cost. For example, the first row represents a strategy profile in which all players choose their first-strategy, player p_1 has cost 17 and she has an incentive to change to her second-strategy, which induces a cost 15. $\square$

Using the game from previous proof as a gadget, we prove the following theorem.

Theorem 1. *Deciding whether there exists a Nash equilibrium for a given matrix scheduling game is strongly $\mathcal{NP}$-complete. The problem remains $\mathcal{NP}$-complete even for a constant number of machines.*

Proof: The strong $\mathcal{NP}$-completeness is obtained by a reduction from 3-Partition [11]. Here we present a proof of $\mathcal{NP}$-completeness for the game with a constant number of machines and the proof of strong $\mathcal{NP}$-hardness is similar with slight modifications. Given a strategy profile, verifying whether it is an equilibrium can be done in $O(nm)$, so the problem is in $\mathcal{NP}$. In the following, we prove the $\mathcal{NP}$-hardness by a reduction from Partition [11].

In Partition, we are given n integer numbers $a_1, \ldots, a_n$ and the question is whether there exists a partition of these n numbers into two subsets (P_1, P_2) such that the sums of elements in these subsets are equal.

We construct a matrix scheduling game in which the existence of a Nash equilibrium is equivalent to the existence of a solution for an instance of Partition. Given n integers $a_1, \ldots, a_n$, let B be an integer such that $\sum_{i=1}^{n} a_i = 2B$. The reduction game consists of $n+6$ players. The first n players encode the Partition problem, the last three players encode the gadget of Lemma 1 and the remaining three players acts as connection between these groups. Each of the first n players has two strategies $\{m_1\}$ and $\{m_2\}$, the loads of player p_i $(1 \leq i \leq n)$ on machines m_1, m_2 are the same and equal to a_i. Player p_{n+1} has two strategies $\{m_2\}$ and $\{m_3\}$ with loads 0 and $B/2 + \epsilon$, respectively; player p_{n+2} has two strategies $\{m_1\}$ and $\{m_3\}$ with loads 0 and $B/2 + \epsilon$, respectively. Player p_{n+3} has two strategies $\{(m_3, m_4)\}$ and $\{m_8\}$ with loads as shown in Table 2. The last three players represent the gadget of Lemma 1. Note that the load contribution of a player to a machine is infinity if it is not explicitly given.

	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8
p_1	a_1	a_1						
p_2	a_2	a_2						
$\vdots$	$\vdots$	$\vdots$						
p_{n-1}	a_{n-1}	a_{n-1}						
p_n	a_n	a_n						
p_{n+1}		0	$B/2+\epsilon$					
p_{n+2}	0		$B/2+\epsilon$					
p_{n+3}			$B/2$	3				$B/2+4$
p_{n+4}				2		10	15	
p_{n+5}				5	4			
p_{n+6}					4	1		

Table 1. The load matrix of the reduction game.

First we show that if there is a solution $P = (P_1, P_2)$ for the Partition instance then there is a Nash equilibrium for this game. Consider a strategy profile in which player p_i $(1 \leq i \leq n)$ uses machine m_j for $j \in \{1, 2\}$ such that $i \in P_j$, player p_{n+1} uses machine m_2, p_{n+2} uses machine m_1, p_{n+3} uses machines (m_3, m_4) and three others play their second-strategy (defined in Lemma 1). It is straightforward that to verify that no one has an incentive to change her current strategy, hence this strategy profile is a Nash equilibrium. In this equilibrium, all n first players' cost are B.

For the converse, suppose that there is a Nash equilibrium for this game. In this equilibrium, player p_{n+3} must use machines (m_3, m_4) since otherwise, by Lemma 1, the strategy profile is not an equilibrium as it contains an unstable sub-game by the last three players. Hence, both players p_{n+1} and p_{n+2} play their first strategy since otherwise player p_{n+3} will move. Therefore the costs of p_{n+1} and p_{n+2} are at most B. Moreover, $\sum_{i=1}^{n} a_i = 2B$ thus their costs are exactly B. In other words, all n first players form a partition such that the sum in two subsets are the same and therefore we obtain a solution for the PARTITION instance. □

3 Connection Games

In Connection Games, we are given a directed graph $G = (V, E)$ with nonnegative edge costs c_e for all edges $e \in E$. There are n players, each player i has a specified source node s_i and sink t_i. Player i's goal is to build a network together with other players in order to connect her terminals s_i and t_i while paying as little as possible to do so. A strategy of player i is a path P_i from s_i to t_i in G. Given a strategy P_i for player i, the constructed network is $\cup_i P_i$, which induces the *social cost* $\sum_{e \in \cup_i P_i} c_e$ that is fully paid by players in the game.

Consider the *Shapley cost-sharing protocol* that splits the cost of an edge evenly among all players using this edge. Formally, given a strategy profile S, if n_e denotes the number of players whose path contains edge e then e assigns a cost c_e/n_e to each player using e. Thus, the total cost of player i in strategy profile S is $c_i(S) := \sum_{e \in P_i} c_e/n_e$. A *Nash equilibrium* is a strategy profile that is resilient to unilateral deviations. The Connection Games using the Shapley cost-sharing protocol is well studied in [1]. This game always possesses a Nash equilibrium. The inefficiency of the constructed network is measured by the *price of anarchy (PoA)* and the *price of stability (PoS)*. The PoA is defined as the ratio between the costs of the worst Nash equilibrium and the optimum, the PoS is defined as the ratio between the costs of the best Nash equilibrium and the optimum.

3.1 Weighted Connection Games

The Weighted Connection Games is similar to the Connection Games except that in the former, each player i has additionally a weight w_i and she needs to carry the weight from her source to her sink. Consider the *weighted Shapley cost-sharing protocol* that splits the cost of an edge proportionally to the players' weight. Formally, given a strategy profile S, let W_e be the total weight of players whose path contains e, i.e., $W_e = \sum_{i: e \in P_i} w_i$, the cost of player i on edge e is $c_e \cdot w_i/W_e$. The total cost of player i in strategy profile S is $c_i(S) = \sum_{e: e \in P_i} c_e \cdot w_i/W_e$.

As showed in [5], there does not necessarily exist an equilibrium for Weighted Connection Games. We use the counterexample in [5] as the gadget in proving the complexity on the existence of Nash equilibrium in the game. For completeness, we present here the gadget.

Lemma 2 ([5]). *There is a 3-player weighted connection game using weighted Shapley cost-sharing protocol that admits no Nash equilibrium.*

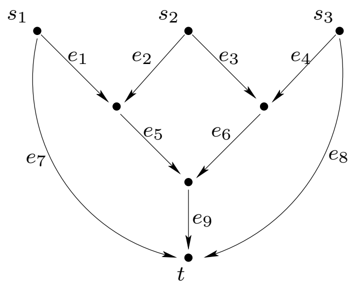

Fig. 3. A 3-players weighted Shapley connection game with no Nash equilibrium.

Edge	Cost
e_1	0
e_2	3ϵ
e_3	0
e_4	0
e_5	$w^3/(w^2+w+1)-\epsilon$
e_6	$w^3/(w^2+w+1)+\epsilon$
e_7	$(w^3+w^2)/(w^2+w+1)-$ $-\ \epsilon(2w^2+1)/(2w^2+2)$
e_8	$(w^3+w^2)/(w^2+w+1)-$ $-\ \epsilon(2w+1)/(2w+2)$
e_9	1

Fig. 4. The cost of edges in Lemma 2

Proof: Let $w > 1$ be an arbitrary number and $\epsilon > 0$ be much smaller than $1/w^3$. Consider the network in Figure 3 with players 1,2 and 3 which have weight $w^2, 1$ and w, respectively. The costs of edges are given in Figure 4 and are chosen in such a way that they satisfy the following inequalities.

$$c_5 \cdot \frac{w^2}{w^2+1} + c_9 \cdot \frac{w^2}{w^2+1} > c_7 > c_5 + c_9 \cdot \frac{w^2}{w^2+w+1} \tag{1}$$

$$c_6 + c_9 \cdot \frac{w}{w^2+w+1} > c_8 > c_6 \cdot \frac{w}{w+1} + c_9 \cdot \frac{w}{w+1} \tag{2}$$

If the second player uses path $e_2 \to e_5 \to e_9$ then the third player will use path e_8 (by the first half of the inequality (2)) and the first player strategy will be e_7 (by the first half of the inequality (1)). Hence, the second player will switch to the path $e_3 \to e_6 \to e_9$ to decrease her cost.

If the second player uses path $e_3 \to e_6 \to e_9$ then the third player will use path $e_4 \to e_6 \to e_9$ (by the second half of the inequality (2)) and so the first player strategy will be $e_1 \to e_5 \to e_9$ (by the second half of the inequality (1)). But in this case, the second player will switch to the path $e_2 \to e_5 \to e_9$ and get better off.

The two cases conclude the lemma.

$\square$

We use this lemma as the gadget of framework to prove the $\mathcal{NP}$-hardness of Nash equilibrium. We construct a larger weighted connection game based on a $\mathcal{NP}$-hard problem and embed the gadget into the larger game to relate the existence of a solution for an instance of the $\mathcal{NP}$-hard problem to the existence of Nash equilibrium in the game. Part of our constructed network is inspired by the contruction in [7].

Theorem 2. *It is $\mathcal{NP}$-complete to decide whether a given weighted Shapley connection game admits a Nash equilibrium.*

Proof: It is straightforward that verifying whether a strategy profile is an equilibrium of the game can be done in polynomial time. In the following, we prove the $\mathcal{NP}$-hardness by a reduction from MONOTONE3SAT[1] [11]. Consider an instance of MONOTONE3SAT with set of variables $X = \{x_1, \ldots, x_n\}$ and set of clauses $C = (c_1, \ldots, c_m)$. Each clause contains at most three literals and either all literals in a clause are negated or all are unnegated. Deciding whether there is a satisfying assignment for this instance is $\mathcal{NP}$-hard.

We construct a game such that for each literal $x \in X$, there is a *literal player* p_x with weight 1, source x and sink $\bar{x}$. Moreover, each clause $c \in C$ gives rise to a *clause player* p_c with weight 1, source c and sink $\bar{c}$. Besides, we have three additional players p_1, p_2, p_3 of weight $w^2, 1, w$ and source/sink $(s_1, t), (s_2, t), (s_3, t)$, respectively. These three players will play the role of the gadget from Lemma 2. One additional player p_4 has weight 1 and source/sink (s_4, t_4). Remark that in the reduction network, all players have weight 1 except players p_1 and p_3. In our construction, ϵ is positive and arbitrarily small.

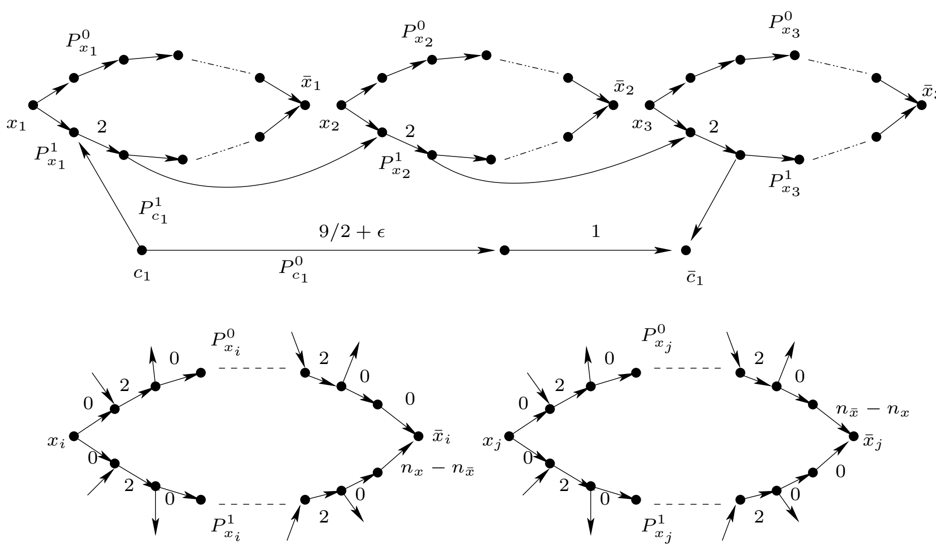

Fig. 5. Network of players p_x and p_c. Two paths of player p_{c_1}, for $c_1 = x_1 \vee x_2 \vee x_3$, are illustrated.

We first describe the sub-network for all players p_x and p_c, $x \in X, c \in C$. Part of this sub-network is described in Figure 5. For player p_x, there are two paths P^0_x, P^1_x from x to $\bar{x}$. Let $n_x := |\{c \in C | x \in c\}|$ and $n_{\bar{x}} := |\{c \in C | \bar{x} \in c\}|$. Path P^1_x consists of $(2n_x + 2)$ edges and path P^1_x consists of $(2n_{\bar{x}} + 2)$ edges. On each path, the cost of all oddth edges is 0 and that of all eventh edges is 2 except

[1] The choice of MONOTONE3SAT is driven by the simplicity in drawing the network. We can make reduction from 3SAT with exactly the same arguments

the last edge. If $n_x > n_{\bar{x}}$ then the cost of the last edge on path P_x^0 is $(n_x - n_{\bar{x}})$ and that on path P_x^1 is 0. Otherwise, the cost of the last edge on path P_x^0 is 0 and that on path P_x^1 is $(n_{\bar{x}} - n_x)$. Each player p_c also has two paths P_c^0, P_c^1 from c to $\bar{c}$. Path P_c^0 consists of two edges with cost $9/2 + \epsilon$ and 1. Path P_c^1 consists of seven edges and is constructed for $c = c_j$ in the order $j = 1, \ldots, m$ as follows. For a positive clause $c = c_j = (x_{j_1} \vee x_{j_2} \vee x_{j_3})$ with $j_1 < j_2 < j_3$, path P_c^1 starts with the edge connecting source c to the first inner node v_1 on path $P_{x_{j_1}}^1$ that has only two incident edges so far. The second edge is the unique edge (v_1, v_2) of path $P_{x_{j_1}}^1$ that has v_1 as its start vertex. The third edge connects v_2 to the first inner node v_3 on path $P_{x_{j_2}}^1$ that has only two incident edges so far. The fourth edge is the only edge (v_3, v_4) on $P_{x_{j_2}}^1$ with start vertex v_3. Similarly, the fifth edge is the edge connecting v_4 to the first inner node v_5 of $P_{x_{j_3}}^1$ which has only two incident edges so far, followed by (v_5, v_6). The last edge of P_c^1 connects v_6 to sink $\bar{c}$. For a negative clause $c_\ell = (\bar{x}_{\ell_1} \vee \bar{x}_{\ell_2} \vee \bar{x}_{\ell_3})$, the construction is similar. The difference with a positive clause is that a positive clause concerns only paths (of literal players in the clause) of superscript 1 while a negative clause concerns only with those of superscript 0.

The second part of the network consists of four players p_1, p_2, p_3 and p_4. First three players have a network (with edge costs) defined in Lemma 2. The only difference is that this network has an additional edge e_{10} of cost 0. Player p_4 has two paths P_4^0, P_4^1 connecting her source s_4 and her sink t_4. Path P_4^1 consists of edge e_8 and an additional edge (t, t_4) of cost $m - c_8/(w+1) - \epsilon$. Path P_4^0 shares edges of cost 1 with all paths $P_c^0, \forall c \in C$ and contains some additional edges (of cost 0) connecting those. The network with its edges' cost is shown in Figure 6. Note that each player in our network possesses two strategies. We call a *0-path* (*1-path*, resp) of a player the path with superscript 0 (1, resp).

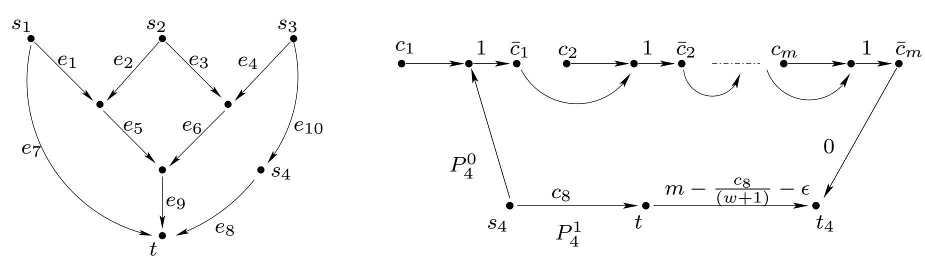

Fig. 6. Network of players p_1, p_2, p_3 and p_4 (an edge has cost 0 if it is not given).

Given a satisfying assignment, we argue that the strategy profile in which players p_x ($x \in X$) use their j-path ($j \in \{0, 1\}$) corresponding to the value j of x in the solution, players p_c ($c \in C$) use their 1-path, player p_4 uses her 1-path and players p_1, p_2, p_3 use paths $(e_7), (e_3 \to e_6 \to e_9), (e_{10} \to e_8)$ respectively is a Nash equilibrium. Player p_x has no incentive to switch her strategy because her cost stays the same (it equals $\max\{n_x, n_{\bar{x}}\}$) even if she changes the strategy (due to the trick that we add a new edge of cost $|n_x - n_{\bar{x}}|$ to some paths). Player p_c's cost is at most 5 since in a satisfying assignment, she shares at least an edge of cost 2 with a player $p_x, x \in X$. Observing that if using 0-path, player p_c may share only one edge of cost 1 with player p_4 so if she switches to 0-path, in the best case, her cost would be $9/2 + \epsilon + 1/2 > 5$. Hence, player p_c is happy on the 1-path. The cost of player p_4 is $m - \epsilon$ and she is happy on her current strategy. It is easy to verify that all three players p_1, p_2 and p_3 are also happy.

Suppose there is a Nash equilibrium in this game. Hence, in this equilibrium, player p_4 must use her 1-path since otherwise the strategy profile is not an equilibrium (by Lemma 2). The cost of p_4 is at least $m - \epsilon$ and this happens in case p_4 shares edge e_8 with p_3. Therefore all players p_c use her 1-path because if there is a player p_c uses her 0-path, player p_4 has an incentive to change her strategy and get a cost at most $m - 1/2$. The fact that players p_c ($\forall c \in C$) play their 1-path means that, for each $c \in C$, there is at least one player p_x shared an edge with p_c (otherwise, p_c will change her strategy). Hence, the assignment in which $x_i = 1$ if p_x uses 1-path and $x_i = 0$ otherwise gives a satisfying assignment for the MONOTONE3SAT instance. □

3.2 Cost-Sharing protocol for Good Equilibrium

The inefficiency of equilibria is measured by the price of anarchy (PoA) and the price of stability (PoS). In previous subsection, we consider the Shapley cost-sharing protocol as the allocation of cost to players in the constructed network. Under this cost-sharing protocol, the PoA may be as large as n which is the number of players in the game and the PoS may be as large as $\log n$ [1]. A natural question is whether there exists a cost-sharing protocol which guarantees a small inefficiency of the game. Chen, Roughgarden and Valiant [6] have studied the inefficiency of the Connection Games while considering the set of *admissible* cost-sharing protocols. A cost-sharing protocol is *admissible* if it satisfies: (i)*Budget-balance*: the cost of each edge in the constructed network is fully passed on to its users; (ii)*Separability*: the cost shares of an edge are completely determined by the set of players that use it; (iii)*Stability*: for every network using the cost-sharing protocol, there is at least one Nash equilibrium. In that paper, they leave a challenging open question in designing an admissible cost-sharing protocol such that for all networks, the PoS induced from this protocol is small, say constant.

Consider the Shapley cost-sharing protocol. This protocol has a good property, namely fairness (it divides evenly the cost of an edge to players using this edge). We are interested in a question similar to the one above but restricted to admissible cost-sharing protocols which possess a fairness property. First, we define the property ϵ-fairness which is desired in a cost-sharing protocol. Intuitively, in these protocols, if an edge is shared by several players, no one pays a too large fraction of the cost.

Definition 1. *Given $0 < \epsilon < 1/2$, an ϵ-*fair* cost-sharing protocol is a cost-sharing protocol in which if there are at least two players sharing an edge then no one pays more than $1 - \epsilon$ fraction of this edge cost.*

In Connection Games where all players have the same sink, there is a cost-sharing protocol which induces PoS $= 1$ for every network [2]. Nevertheless, the best Nash equilibrium is not very efficient compared to the optimal outcome in general case. Precisely, [6] obtained that for any admissible cost-sharing protocol, the PoS is at least $3/2$. Here, we present our proof for this lower bound.

Lemma 3 ([6]). *There is a directed network whose PoS is at least 3/2 for any admissible cost-sharing protocol.*

Proof: Consider the network in Figure 7, player p_i has source/sink (s_i, t_i) for $1 \leq i \leq n$. The *backbone* path consists of edges $e_1, e_2, \ldots, e_n$ of cost $1 + 1/n$ altering with edges of cost 0. Players can use the backbone path or other paths to connect her terminals. Player p_i $(2 \leq i \leq n)$ has a path containing three edges of cost $1/2$ that we call *one-hop-source edge* and *one-hop-sink edge* the first and the last among these three edges, respectively. The optimum, where all players use the backbone path, is of cost $n + 1$. We will prove that for any admissible cost-sharing protocol, an equilibrium of n players on this network has cost at least $3(n-1)/2 + 1$.

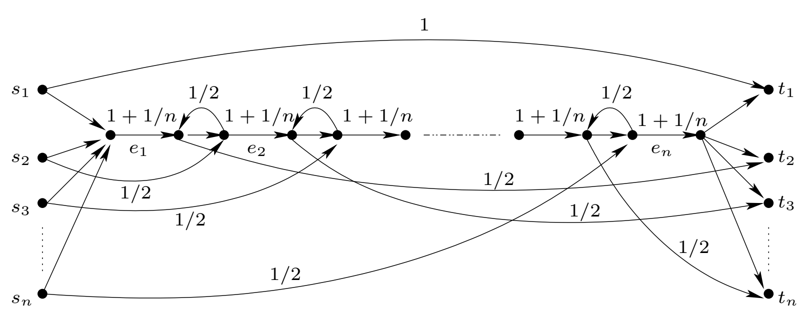

Fig. 7. Network of n players whose PoS $= 3/2$ (an edge has cost 0 if it is not explicitly given).

Fix an admissible cost-sharing protocol. We claim that in an equilibrium induced by the protocol, the first player does not use the backbone path. Assume that in an equilibrium S, p_1 uses the backbone path. Player p_i's cost $(2 \leq i \leq n)$ restricted to the backbone path is at most $1/2$ before edge e_i and at most $1/2$ after (including) edge e_2 since otherwise p_i has incentive to use her one-hop-source edge or one-hop-sink edge. Hence, the total cost of all players p_i $(2 \leq i \leq n)$ restricted on the backbone path is at most $n - 1$, so p_1 pays at least 2 on the backbone path. In this case, p_1 would switch to the one-hop path of cost 1, that contradicts to the assumption of equilibrium. Moreover, we claim that in an equilibrium, no player entirely uses the backbone path. Suppose in an equilibrium there are $k - 1$ players (different to the first player) entirely using the backbone path, i.e., $(n - k)$ other players use at least one of their one-hop edges in order to connect their terminals. The cost restricted on the backbone path of each one in these $(n-k)$ players is at most 1 (otherwise a player, whose restricted cost is larger than 1, would change her path restricted on the backbone by an one-hop edge with or without a backward edge of cost 1/2). Therefore, the total cost shared by $k-1$ players entirely using the backbone path is at least k. In other words, there is a player p_i who strictly pays more than 1 on the backbone path, it means that she strictly pays more than 1/2 on her path before e_i or after (including) e_i. Hence, this player can decrease her cost by switching her strategy.

Consider an equilibrium S and let $R = \{j : e_j \in S\}$. If $R = \emptyset$ then in S each player p_i $(2 \leq i \leq n)$ pays 3/2 and player p_1 pays 1. The total cost is $3/2 \cdot (n-1) + 1$. If $R \neq \emptyset$, let i be the smallest element in R. If $i > 1$ then p_i uses her one-hop-source edge and no player p_j uses an edge e_j with $j \leq i$ on the backbone, since otherwise that would contradict the minimality of i. Thus, player p_i fully pays edge e_i. In this case, the total cost of p_i is $3/2 + 1/n$ and she has an incentive to switch her strategy. Hence $i = 1$. Let T be the set of players using e_1 in S and $k := \min\{j : p_j \in T\}$. Remark that no one uses the full backbone path. Since S is an equilibrium, edge e_k is shared by p_k and some other player of index $\ell < k$ (otherwise, p_k is the only one who pays edge e_k and she is not happy). Since $\ell < k$ and p_ℓ uses an edge e_k, player p_ℓ must use a path containing $\{e_k, e_{k+1}, \ldots, e_n\}$ (she can not use her one-hop-sink edge). That means the backbone path is entirely bought. Moreover, no one uses the backbone path, i.e., each player uses at least an edge of cost 1/2. Hence, the total cost in S is $n + 1/2 \cdot (n-1)$. □

The following theorem shows that it is intractable to find an ϵ-fair admissible cost-sharing protocol which ensures small inefficiency of equilibria of the game. It also highlights an intuition in designing a cost-sharing protocol for the Connection Games with good PoS: this kind of cost-sharing protocol may not be fair.

Theorem 3. *Given a network $G(V,E)$ with edge costs $c : E \to \mathbb{Q}$ and a set of players, deciding whether there exists an ϵ-fair cost-sharing protocol such that the PoS $\leq 3/2 - \delta$ is $\mathcal{NP}$-hard, where $\delta > 0$ can be chosen arbitrarily small.*

Acknowledgments. We thank Christoph Dürr and Nguyen Hai Linh for helpful and interesting discussions.

References

1. Elliot Anshelevich, Anirban Dasgupta, Jon Kleinberg, Eva Tardos, Tom Wexler, and Tim Roughgarden. The price of stability for network design with fair cost allocation. In *Proceeding of the 45th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 295–304, 2004.
2. Elliot Anshelevich, Anirban Dasgupta, Éva Tardos, and Tom Wexler. Near-optimal network design with selfish agents. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing (STOC)*, pages 511–520, 2003.

3. Jean R. S. Blair, David Mutchler, and Michael Van Lent. Perfect recall and pruning in games with imperfect information. *Computational Intelligence*, 12:131–154, 1996.
4. Chandra Chekuri, Julia Chuzhoy, Liane Lewin-Eytan, Joseph Naor, and Ariel Orda. Non-Cooperative Multicast and Facility Location Games. In *Proceedings of the 7th ACM Conference on Electronic Commerce (EC)*, pages 72–81, 2006.
5. Ho-Lin Chen and Tim Roughgarden. Network Design with Weighted Players. In *Proceedings of the 18th Annual ACM Symposium on Parallel Algorithms and Architectures (SPAA)*, pages 29–38, 2006.
6. Ho-Lin Chen, Tim Roughgarden, and Gregory Valiant. Designing Networks with Good Equilibria. In *Proceedings of the 19th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2008.
7. Juliane Dunkel and Andreas S. Schulz. On the Complexity of Pure-Strategy Nash Equilibria in Congestion and Local-Effect Games. In *Internet and Network Economics, Second International Workshop (WINE)*, pages 62–73, 2006.
8. Christoph Dürr and Nguyen Kim Thang. Nash Equilibria in Voronoi Games on Graphs. In *Proceedings of the 15th Annual European Symposium on Algorithms (ESA)*, pages 17–28, 2007.
9. Eyal Even-Dar, Alexander Kesselman, and Yishay Mansour. Convergence time to Nash equilibrium in load balancing. *ACM Transactions on Algorithms*, 3(3), 2007.
10. Alex Fabrikant, Christos H. Papadimitriou, and Kunal Talwar. The complexity of pure Nash equilibria. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing (STOC)*, pages 604–612, 2004.
11. Michael R. Garey and David S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness.* W. H. Freeman & Co., New York, NY, USA, 1990.
12. Dov Monderer and Lloyd S. Shapley. Potential games. *Games and Economic Behavior*, 14:124–143, 1996.
13. Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani. *Algorithmic Game Theory.* Cambridge University Press, New York, NY, USA, 2007.
14. R.W. Rosenthal. A class of games possessing pure-strategy Nash equilibria. *International Journal of Game Theory*, 2:65–67, 1973.