Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Distributing Potential Games on Graphs Part II. Learning with application to platoon matching

Mohamed I. El-Hawwary [*,**] Jonas Mårtensson [*]

[*] Division of Decision and Control Systems, School of Electrical Engineering & Computer Science, KTH Royal Institute of Technology, SE-100 44 Stockholm, Sweden
[**] Department of Electrical Power Engineering, Faculty of Engineering, Cairo University, Giza 12613, Egypt
(e-mail: melhawwary@eng.cu.edu.eg, jonas1@kth.se)

Abstract: In part I of the paper the problem of distributing potential games over undirected graphs was formulated. A restricted information potential game was designed using state-based formulation. Here, learning Nash equilibria for this game is studied. An algorithm is developed with mainly two phases, an estimation phase and a learning phase. The setting allows for available learning methods of the full information game to be directly incorporated in the learning phase. The result matches the outcome (i.e. converges to the same equilibria) of the full information game. In addition, the design takes into account considerations of convergence time, and synchrony of actions update. The developed distributed game and learning algorithm are used to solve a platoon matching problem for heavy duty vehicles. This serves two objectives. First, it provides a motivation for the presented gaming results. Second, the problem addressed can facilitate platoon matching where it provides a basis for an on-the-go strategy.

Keywords: Potential games, distributed optimization, multi-agents, platoon matching.

1. INTRODUCTION

Learning equilibria earned significant attention in literature of game theory Fudenberg and Levine (1998), Young (2005). In engineering applications one can identify several classes of learning algorithms. For example classes based on information include, full information where agents know structural form of their own utilities, and are able to observe all others' actions; oracle-based information where agents can evaluate their own utility values for all possible actions; and pay-off based where agents can observe their actions and payoffs. The learning algorithm presented in this paper can be seen as part of communication-based information class where agents exchange information with local neighbours.

Several learning algorithms have been developed for potential games Lã et al. (2016). Standard algorithms include best and better reply dynamics Nisan et al. (2008). Fictitious play is an example of an algorithm based on full information, and joint strategy fictitious play is an example of an oracle-based information algorithm Marden et al. (2009). The issue of convergence to specific equilibria, such as maximizers of potential functions, has been handled by algorithms like log-linear dynamics and its derivatives Marden and Shamma (2012). The learning algorithm developed here allows for full information algorithms for potential games to be directly applied to the distributed potential games. The algorithm is composed of two phases. An estimation phase where the game converges to its full information counterpart, and a learning phase where equilibrium convergence properties of the full information game can migrate to the restricted information one.

Consecutively, the distributed potential game will be used to solve a platoon matching problem for heavy duty trucks. Truck platooning has acquired significant attention in recent years due to demonstrated potential for fuel savings Browand et al. (2004), Tsugawa et al. (2016) Bishop et al. (2017). In addition, platooning can have other beneficial aspects on traffic, safety, drivers and on the environment Davila et al. (2013), Alam et al. (2015).

One of the main challenges of platooning is the matching problem, i.e. how to decide when, where and with whom to platoon. Several strategies have been used to solve this problem, and one could distinguish two main solution classes: centralized and game theoretic. Centralized solutions rely on formulating the problem as a centralized optimization problem with a goal of minimizing/maximizing global objective functions. In game theoretic solutions agents are modeled as competing agents seeking to optimize individual profit functions. Several approaches have been proposed in literature for the centralized methods. The main challenges such approaches would face are related to the size of the problem, compatibility of goals and information sharing. A platooning problem involves a large number of agents with a larger number of solution variables, e.g. time and road plans parameters, driver parameters (rest times, work loads, restrictions, salaries), and vehicles parameters (mass, size, braking and communication capabilities), which can yield a centralized solution intractable. Another challenge for centralized platoon matching is related to incentives. Goals for agents belonging to different fleets or operators can be difficult to consolidated in a common one. Similarly, in that context, those agents might not be willing to share the necessary information for a global solution, or this information cannot be shared for a different reasons such as lake of communication. Different centralized solutions for platoon matching have been proposed in literature Larsson et al. (2015), Liang et al. (2016), van de Hoef et al. (2018).

The game theoretic approach for platoon matching has received lesser attention. Two results are cited here. In Farokhi and Johansson (2013) a platoon matching problem of a group of trucks with the same origin and destination was modeled as a non-cooperative game. Platooning decisions were made by varying departure times where each agent tries to minimize its travel cost. In Johansson et al. (2018) a problem was addressed where the vehicles do not necessarily have the same destination. Instead, they may have different plans (with the same origin) on a road network defined by a graph with tree topology. A strategic potential game was developed where agents strategies (actions) are their departure times, and their utility functions are defined as the difference between savings from platooning, and the cost of deviations from the preferred departure times. The development here presents a way to generalize those results to a setting that can handle general networks where trucks don't share the same origin. This could mean that complete strategy sharing is not possible, and hence the utilization of the previously developed restricted information game.

The paper is organized as follows, The distributed potential game developed in part I is reviewed in Section 2. The learning algorithm for this game is developed in Section 3, and the the platoon matching application is presented in Section 4. Conclusions are provided in Section 5.

2. THE DISTRIBUTED POTENTIAL GAME

In part I of the paper the problem of distributing potential games over communication graphs was defined as follows.

Consider a strategic form potential game Lã et al. (2016)

$$g = \langle \mathcal{N}, (S_i)_{i\in\mathcal{N}}(u_i)_{i\in\mathcal{N}}\rangle \tag{1}$$

with potential function $\phi : S \to \mathbb{R}$, and suppose that each agent $i \in \mathcal{N}$ can access the strategies of only a subset $\mathcal{N}_i$ of the agents according to a connection graph $\mathcal{G}$ under the following assumptions.

Assumption 1. $\mathcal{G}$ is an undirected graph. If agent i sees agent j according to $\mathcal{G}$ then agent i has access to the full strategy profile as seen by agent j. It is not assumed that agent i has access to $\mathcal{N}_j$.

Assumption 2. For all $i \in \mathcal{N}$, $j \in \mathcal{N}_i$ passes on to i all the values it receives for s_i from the other agents in $\mathcal{N}_j$.

By distributing the potential game g on the graph $\mathcal{G}$, it is meant to

- find new utility functions and $\Phi(S, \cdot)$ with codomain $\mathbb{R}$, satisfying Assumptions 1 and 2, and utilizing if possible the full information utility functions and the potential function ϕ, such that in some sense (determine what) the game is a potential game with potential function Φ; and
- determine the Nash equilibria for the new game.

The distributed potential game on $\mathcal{G}$ is defined as follows.

Let $e = (e^1, \cdots, e^n) \in (\mathbb{R}^n)^n$ $((\mathbb{R}^n)^n := \mathbb{R}^n \times \cdots \times \mathbb{R}^n)$ define the estimation profile, where $e^i = (e^i_1, \cdots, e^i_n)$ is agent's i estimate of the joint strategy profile s.

Assumption 3. For $j \in \mathcal{N}_i$, $e^i_j = s_j$, i.e. agent i only estimates the strategies of the agents it cannot see.

g distributed over $\mathcal{G}$ is given by a state-based game [1]

$$G = \langle \mathcal{N}, \mathcal{X}, \mathcal{A}, (v_i)_{i\in\mathcal{N}}, f\rangle$$

where

- the state of the game is defined as $x = (s, e)$, the state space is given by $\mathcal{X} = S \times (\mathbb{R}^n)^n$, and the individual state is $x_i := (s_i, e^i)$;
- each agent's actions is a set $a_i = (\hat{s}_i, \hat{e}^i_j|_{j\in\mathcal{N}\setminus\mathcal{N}_i})$ where $\hat{s}_i$ indicates some change in the agent i strategy s_i, and $\hat{e}^i_j$ indicates some change in the agent's estimation strategy e^i_j;
- the joint action profile is defined as $a = (\hat{s}, \hat{e})$ where by Assumption 3, $\hat{e}^i_j = \hat{s}_j$, for all $j \in \mathcal{N}_i$, $i \in \mathcal{N}$; and
- the state transition function is defined as $f = (f_s, f_e)$ where the ensuing state $\tilde{x} = (\tilde{s}, \tilde{e})$ is given by
$$\tilde{s} = f_s(s, \hat{s}),\ \tilde{e} = f_e(e, \hat{e})$$
with $\tilde{e}^i_j = \tilde{s}_j$, for all $j \in \mathcal{N}_i$.

By choosing utility functions

$$\begin{aligned} v_i(x,a) = &\sum_{k\in\mathcal{N}_i} u_i(\tilde{s}_j|_{j\in\mathcal{N}_k}, \tilde{e}^k_j|_{j\in\mathcal{N}\setminus\mathcal{N}_k}) \\ &- \alpha \sum_{k\in\mathcal{N}_i}\sum_{l\in\mathcal{N}_k} \mathfrak{p}\left(\tilde{s}_i - \tilde{e}^k_l\right) - \alpha \sum_{k\in\mathcal{N}_i}\sum_{j\in\mathcal{N}\setminus\mathcal{N}_i} \mathfrak{p}\left(\tilde{e}^i_j - \tilde{e}^k_j\right) \end{aligned} \tag{2}$$

where $\alpha > 0$ and $\mathfrak{p} : \mathbb{R} \to [0, \infty)$, it was shown in Proposition 8 of part I that G is a potential game [2] with potential function

$$\begin{aligned} \Phi(x,a) = &\sum_{i\in\mathcal{N}} \phi\left(\tilde{s}_j|_{j\in\mathcal{N}_i}, \tilde{e}^i_j|_{j\in\mathcal{N}\setminus\mathcal{N}_i}\right) \\ &- \frac{\alpha}{2}\sum_{i\in\mathcal{N}}\sum_{k\in\mathcal{N}_i}\sum_{j\in\mathcal{N}\setminus(\mathcal{N}_k\cup\mathcal{N}_i)} \mathfrak{p}\left(\tilde{e}^i_j - \tilde{e}^k_j\right) \\ &- \alpha\sum_{i\in\mathcal{N}}\sum_{k\in\mathcal{N}_i}\sum_{j\in\mathcal{N}_k\setminus(\mathcal{N}_k\cap\mathcal{N}_i)} \mathfrak{p}\left(\tilde{e}^i_j - \tilde{s}_j\right) \end{aligned}$$

where $\alpha > 0$, and $\mathfrak{p} : \mathbb{R} \to [0, \infty)$.

3. LEARNING

In this section learning algorithms for the Nash equilibria of the state-based potential game G are studied with the following objectives considered.

(1) The equilibrium learned should coincide with one of the underlying strategic game (the full information game) equilibria.
(2) A learning algorithm should be applicable for both continuous and discrete action spaces.

[1] Refer to Appendix A for definition of state-based games.
[2] Definition of state-based potential game can be found in Appendix A.

(3) The algorithm should work whether players update their strategies one by one or in groups at a time.
(4) The time increase from a similar algorithm applied to the underlying strategic game should be linear in the number of the new variables (the estimates).

Remark 1. Under some conditions of differentiability of $\Phi(\cdot)$, and convexity of the action sets $\mathcal{A}_i(x)$ one can design a gradient-based algorithm, like the one used in Li and Marden (2013), that satisfies objectives 3 and 4.

3.1 State transition

Consider the following state transition function $f = (f_s, f_e)$ given by

$$\tilde{s} = f_s(s, \hat{s}) = s + \hat{s}$$
$$\tilde{e} = f_e(e, \hat{e}) = \begin{cases} \tilde{e}^i_j = \tilde{s}_j, \ j \in \mathcal{N}_i \\ \tilde{e}^i_j = \frac{1}{|\mathcal{N}_i - 1|} \sum_{k \in \mathcal{N}_i \setminus \{i\}} e^k_j, \ j \in \mathcal{N} \setminus \mathcal{N}_i \end{cases} \tag{3}$$

The next result shows that when there is no strategies update, i.e. $\hat{s} = 0$, $\tilde{e}^i_j$ asymptotically converges to s_j, for all $j \in \mathcal{N} \setminus \mathcal{N}_i$.

Define the s parametrized function

$$\begin{aligned} \mathrm{E}_s(e') = &- \frac{\alpha}{2} \sum_{i \in \mathcal{N}} \sum_{k \in \mathcal{N}_i} \sum_{j \in \mathcal{N} \setminus (\mathcal{N}_k \cup \mathcal{N}_i)} \frac{1}{|\mathcal{N}_i - 1|} \mathfrak{p}\left(e^i_j - e^k_j\right) \\ &- \alpha \sum_{i \in \mathcal{N}} \sum_{k \in \mathcal{N}_i} \sum_{j \in \mathcal{N}_k \setminus (\mathcal{N}_k \cap \mathcal{N}_i)} \frac{1}{|\mathcal{N}_i - 1|} \mathfrak{p}\left(e^i_j - s_j\right) \end{aligned} \tag{4}$$

with

$$e' := [e'_1 \ \cdots \ e'_{n'}]^\top, \ e'_{o(j)} = \left[e^i_j, i \in \mathcal{N} \setminus \mathcal{N}_j\right]^\top, \ j \in \mathcal{N}'.$$

where $\mathcal{N}' \subset \mathcal{N}$ is the set of agents whose strategies are not seen by all agents, $n' = |\mathcal{N}'|$, and $o : j \in \mathcal{N}' \to 1, \cdots, n'$ is a ordering of $\mathcal{N}'$. Note that the size of vector e'_i equals $\bar{n}_{o^{-1}(i)}$. The function (4) can be written as

$$\mathrm{E}_s(e') = \sum_{j \in \mathcal{N}'} \mathrm{E}_{s_j}(e')$$

where $\mathrm{E}_{s_j}(e')$ is composed of the j components of $\mathrm{E}_s(e')$.

Proposition 2. Consider the state based potential game with state transition function (3). If $\hat{s} = 0$ and $\mathcal{G}$ is connected then $\tilde{e}^i_j$ asymptotically converges to s_j, $\forall j \in \mathcal{N} \setminus \mathcal{N}_i$, under synchronous or asynchronous transition.

The proof is omitted for space limitations, and will be provided elsewhere.

3.2 Learning algorithm

Consider the following learning algorithm: for each $i \in \mathcal{N}$

1. Update the search space ($\mathcal{S}_i$) using the strategies of the agents seen by i.
2. (Estimation) Update agent i estimates $(e^i_j|_{j \in \mathcal{N} \setminus \mathcal{N}_i})$ using the state transition (3).
3. (Learning) If updates reach required tolerance ($\epsilon_i > 0$), i.e.
$$\|e^i_j|_{j \in \mathcal{N} \setminus \mathcal{N}_i}(t+1) - e^i_j|_{j \in \mathcal{N} \setminus \mathcal{N}_i}(t)\| \leq \epsilon_i, \tag{5}$$
and persists for a minimum number of iterations $\bar{k}_i$, apply an algorithm to update agent i strategy (s_i) based on current strategies vector $(s_l|_{l \in \mathcal{N}_i}, e^i_j|_{j \in \mathcal{N} \setminus \mathcal{N}_i})$.

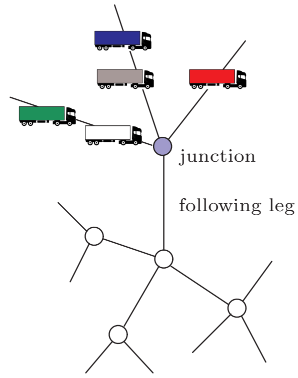

Fig. 1. Platoon matching problem

4. Repeat until a certain convergence measure, or maximum number of learning cycles is reached.

Proposition 3. If the search space is compact, for all $i \in \mathcal{N}$ there exists $c_i > 0$ such that for all $s \in \mathcal{S}$, $|u_i(s) - u_i(s')| \leq c_i |s - s'|$, and the update in step 3 of the learning algorithm is carried out only if $|u_i(t+1) - u_i(t)| \geq 2 c_i \epsilon_i$, then this algorithm inherits the equilibrium convergence properties of the corresponding full information one.

The proof is omitted for space limitations, and will be provided elsewhere.

4. PLATOONING APPLICATION

Consider n trucks approaching a road junction from the same or different directions Figure 1, and assume that those trucks share part of their planned journeys with at least common following leg (road segment) after the junction. The trucks are assumed to have different preferred departure times from that junction, but are able to deviate from these times in order to platoon. Hence the departure time can be considered as the strategy of the trucks. If trucks platoon this would mean a saving in their utility functions. On the other hand, a deviation from their preferred departure times would translate into a cost.

4.1 Strategic potential game for platoon matching

In Johansson et al. (2018) the authors formulated a potential game for a situation where trucks are starting their journeys from a common point in a road network (e.g. parking lot). This corresponds to the trucks being at the same time at the junction in Figure 1. Similar to the previous assumptions, the trucks have preferred departure times which can be modified to achieve platooning. In that work the authors deduced utility functions where the trucks share parts of their journeys including but not limited to the following road segment.

Assuming here that the trucks share only the following leg, the utility function, from Johansson et al. (2018), for the previous situation can be given as

$$u_i(s) = k_p \frac{|\mathcal{P}_i| - 1}{|\mathcal{P}_i|} d - k_t |s_i - t_i|$$

where $k_p, k_t > 0$ are some constants, $\mathcal{P}_i = \{j : s_j = s_i, j \in \mathcal{N}\}$ is the set of trucks with the same departure time as i's, d is the length of the outgoing road segment,

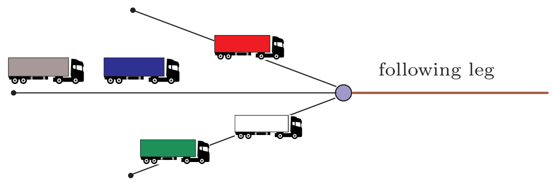

Fig. 2. On-the-go platoon matching

$s_i \in S_i$ is i's departure time, and t_i is i's preferred departure time. From Theorem 1 in Johansson et al. (2018) the strategic game $\langle \mathcal{N}, (S_i)_{i\in\mathcal{N}}, (u_i)_{i\in\mathcal{N}} \rangle$ modeling the platoon matching problem in this situation is an exact potential game with potential function

$$\phi(s) = k_p d \sum_{l\in\{1,\cdots,|\mathcal{P}|\}} \left(|\mathcal{P}_{o(l)}| - 1\right) - \sum_{i\in\mathcal{N}} k_t |s_i - t_i|$$

where $\mathcal{P}$ is the set of distinct P_i's, and $o : \mathbb{Z} \to \mathcal{N}$. This function is equal to the sum of utilities, i.e.

$$\phi(s) = \sum_{i\in\mathcal{N}} u_i(s).$$

Example 1. Consider 5 vehicles with preferred departure times $t = [600, 700, 800, 900, 1000]$, road segment of 100 km, $k_t = 1.5$ \$/hr, $k_p = 5$ \$/100km. This example presents the result of applying better reply dynamics. The search in t is done in a cycle from the current strategy. The strategy is updated only if the utility at the updated value is greater that its current value. Learning is executed in a synchronous manner one agent at a time starting from 1, and cycling through. Figure 3 shows the strategies, utilities, and the sum of utilities (potential function).

4.2 On-the-go platoon matching

The algorithm presented in Johansson et al. (2018) for platoon matching requires the following.

- All trucks need to start from the same point.
- Trucks need to share their complete journey plans with all other trucks.

This limits the applicability of the algorithms on actual road networks where trucks can start from different places, the times of arrival and departure at junctions are affected by external factors such as traffic, and where exchange of itineraries can be limited due to privacy or communication restrictions. Instead consider the following approach.

On-the-go platooning. When trucks are approaching a junction, were platooning is possible, they apply a gaming algorithm for platoon matching, while moving, based on savings/costs of platooning on the following leg only Figure 2. If trucks can communicate strategies with all others then the algorithm in Johansson et al. (2018) can be directly applied. If trucks cannot share strategies with all, due to privacy restrictions or lack of communication, then a restricted information version of the potential game in Section 4.1 is applied. This method can be carried out at any junction.

This would have the advantage of handling platooning on large networks by solving several local matching problems and so, would offer the advantages of simplicity and replicability. However, a comprehensive algorithm is needed to coordinated the different instances of these local problems. This algorithm will be provided in a different work. Next, an example of the local problem is solved.

4.3 Distributed potential game for platooning

Consider trucks approaching a junction as in Figure 2. If they share their strategies according to a bidirectional connected graph, the full information game in Section 4.1 can be directly distributed over that graph utilizing the development in part I.

Example 2. Consider the 5 vehicles in Example 1 with the same parameters. Assume that the vehicles exchange strategies according to an undirected graph with a cyclic Laplacian

$$L = \begin{bmatrix} 2 & -1 & 0 & 0 & -1 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ -1 & 0 & 0 & -1 & 2 \end{bmatrix}.$$

Assume the agents initialize their strategies and those of the unseen agents as their preferred departure times. It is assumed here that the search space for all agents is the vector t. Note that the agents can fill the missing values in the vector t after one iteration. The search in t is done in a cycle from the current strategy value as in Example 1. Assume that the tolerance in step 3 of the learning algorithm of Section 3 is $\epsilon = [5\ 4\ 3\ 2\ 1]$. This example presents the results of applying the algorithm using better reply dynamics for the update in step 3. Figures 4-8 shows the agents strategy vectors, Figure 9 shows the individual utilities, and Figure 10 shows the sum of utilities (potential function). The agents that reach required tolerance are indicated at the iterations. Note that the initial utility values are not shown as the initializations of the unseen strategies are rather optimistic.

5. CONCLUSION

The paper addressed learning for the distributed potential game formulated in part I of the paper. A learning algorithm with prescribed properties was developed. The algorithm is composed of two phases of estimation and learning, and allows for available algorithms of learning Nash equilibria for strategic potential games to be directly incorporated. The results were used to develop an on-the-go method for platoon matching of heavy duty vehicles. A learning example was presented.

Appendix A

Consider a set of $\mathcal{N}$ agents of a strategic form game (1), and a state space $\mathcal{X}$ (here, $\mathcal{X}$ encompasses the set of strategies and their estimates). Define state dependent actions sets $\mathcal{A}_i(x)$ for $i \in \mathcal{N}$ and $x \in \mathcal{X}$. The joint action profile is $a \in \mathcal{A}(x)$ where $\mathcal{A}(x) := \mathcal{A}_1(x) \times \cdots \times \mathcal{A}_n(x)$. The agents utilities are defined by functions $v_i : \mathcal{X} \times \mathcal{A}(x) \to \mathbb{R}$. This game is denoted by $\langle \mathcal{N}, \mathcal{X}, \mathcal{A}, (v_i)_{i\in\mathcal{N}}, f \rangle$ where $\mathcal{A} := \cup_{x\in\mathcal{X}} \mathcal{A}(x)$, and $f : \mathcal{X} \times \mathcal{A} \to \mathcal{X}$ is a deterministic function which defines the state transition (evolution) as influenced in part by the actions. At learning time $\mathtt{t} \geq 0$, each agent's action is selected such that $a_i(\mathtt{t}) \in \mathcal{A}_i(x(\mathtt{t}))$, i.e. it depends on the current state. The current state $x(\mathtt{t})$,

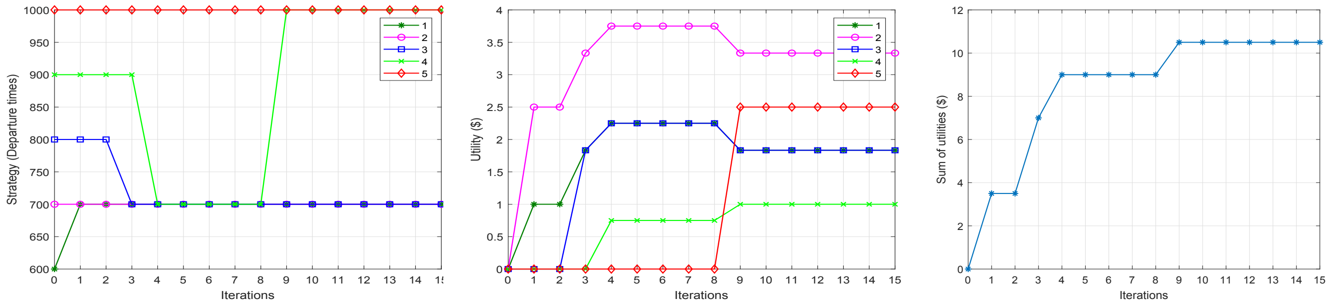

Fig. 3. Full information game

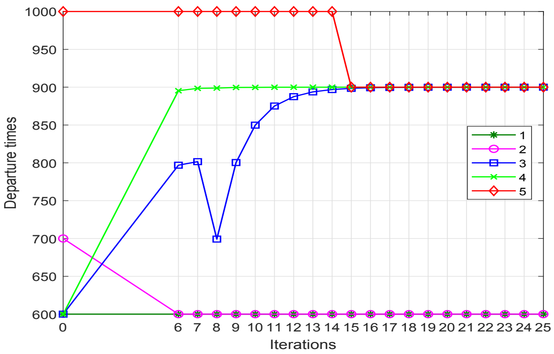

Fig. 4. Restricted information - Strategies vector 1

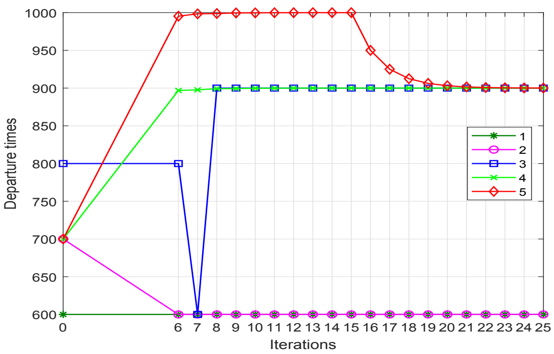

Fig. 5. Restricted information - Strategies vector 2

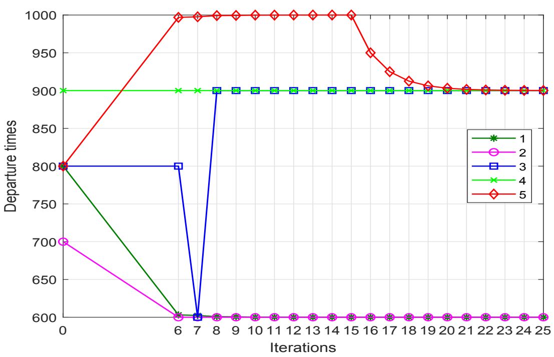

Fig. 6. Restricted information - Strategies vector 3

and the joint action $a(\mathtt{t}) \in \mathcal{A}(x(\mathtt{t}))$ determine the ensuing state $x(\mathtt{t}+1) = f(x(\mathtt{t}), a(\mathtt{t}))$, and the current utilities $v_i(x(\mathtt{t}), a(\mathtt{t}))$.

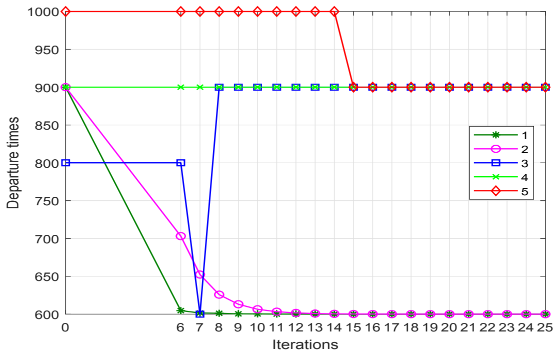

Fig. 7. Restricted information - Strategies vector 4

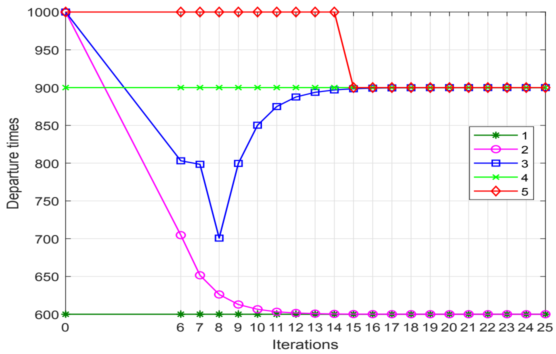

Fig. 8. Restricted information - Strategies vector 5

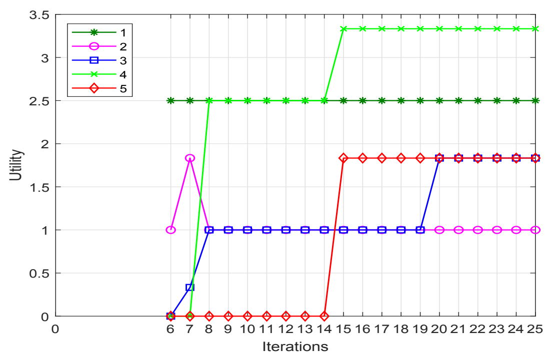

Fig. 9. Restricted information game - Utilities

Definition 4. For the state-based game $\langle \mathcal{N}, \mathcal{X}, \mathcal{A}, (v_i)_{i\in\mathcal{N}}, f \rangle$ an action profile $\mathbf{0} \in \mathcal{A}(x)$ is called a null action if $x = f(x, \mathbf{0})$.

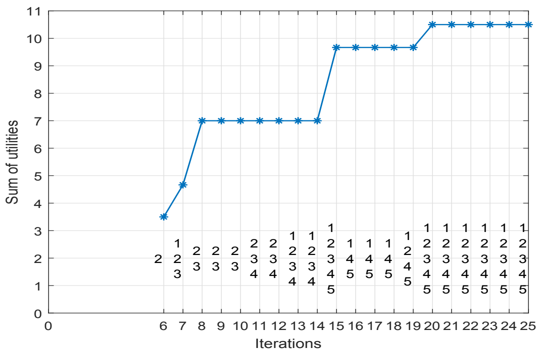

Fig. 10. Restricted information game - Potential function

The definition (corresponding to the Definition of strategic potential games Lã et al. (2016)) of (exact) potential state-based games was given in Li and Marden (2013) as follows.

Definition 5. A (deterministic) state-based game $\langle \mathcal{N}, \mathcal{X}, \mathcal{A}, (v_i)_{i\in\mathcal{N}}, f\rangle$ with null action $\mathbf{0}$ is called a (deterministic) state-based potential game if there exists a function $\Phi : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ such that for every $x \in \mathcal{X}$, $i \in \mathcal{N}$, $a_{-i} \in \mathcal{A}_{-i}(x)$ [3],

$$v_i(x, a_i, a_{-i}) - v_i(x, a_i', a_{-i}) = \Phi(x, a_i, a_{-i}) - \Phi(x, a_i', a_{-i}),$$

for all $a_i, a_i' \in \mathcal{A}_i(x)$, and for every $a \in \mathcal{A}(x)$ the potential function satisfies

$$\Phi(x, a) = \Phi(\tilde{x}, \mathbf{0})$$

where $\tilde{x} = f(x, a)$ denotes the ensuing state.

Two definitions of Nash equilibria for state-based games was given in part I as follows.

Definition 6. A state action pair (x^*, a^*) is a stationary state Nash equilibrium for the game G if

- $(\forall i \in \mathcal{N})(\forall a_i \in \mathcal{A}_i(x^*))\ v_i(x^*, a^*) \geq v_i(x^*, a_i, a^*_{-i})$
- $x^* = f(x^*, a^*)$

This is a restatement of Definition 3 in Li and Marden (2013).

Let $a_i = (a_{i1}, \cdots, a_{in_i})$, where $n_i = 1 + \bar{n}_i$, $\bar{n}_i = |\mathcal{N}\backslash\mathcal{N}_i|$, and $a_{i\_j} = a_i\backslash\{a_{ij}\}$. Also, let $x_i = (x_{i1}, \cdots, x_{in_i})$, where $x_{i\_j} = x_i\backslash\{x_{ij}\}$.

Definition 7. A state action pair (x^*, a^*) is a stationary state Nash equilibrium for the game G if

- $(\forall a_i \in \mathcal{A}_i(x^*))\ v_i(x^*, a^*) \geq v_i(x^*, a_{ij}, a^*_{i\_j}, a^*_{-i})$, for all $i \in \mathcal{N}$, $j \in (1, \cdots, n_i)$
- $x^* = f(x^*, a^*)$

Properties of these definitions were discussed in part I of the paper.

REFERENCES

Alam, A., Besselink, B., Turri, V., Mårtensson, J., and Johansson, K.H. (2015). Heavy-duty vehicle platooning for sustainable freight transportation: A cooperative method to enhance safety and efficiency. *IEEE Control Systems Magazine*, 35(6), 34–56.

Bishop, R., Bevly, D., Humphreys, L., Boyd, S., and Murray, D. (2017). Evaluation and testing of driver-assistive truck platooning: Phase 2 final results. *Transportation Research Record*, 2615(1), 11–18.

Browand, F., McArthur, J., and Radovich, C. (2004). Fuel saving achieved in the field test of two tandem trucks. *Technical report, University of Sourthern California.*

Davila, A., del Pozoa, E., Aramburu, E., and Freixas, A. (2013). Environmental benefits of vehicle platooning. In *Symposium on International Automotive Technology.* Maharashtra, India.

Farokhi, F. and Johansson, K.H. (2013). A game-theoretic framework for studying truck platooning incentives. In *Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013).* The Hague, Netherlands.

Fudenberg, D. and Levine, D. (1998). *The Theory of Learning in Games.* MIT Press, Cambridge, MA.

Johansson, A., Nekouei, E., Johansson, K.H., and Mårtensson, J. (2018). Multi-fleet platoon matching: A game-theoretic approach. In *Proceedings of the 21st International Conference on Intelligent Transportation Systems (ITSC 2018).* Maui, Hawaii, USA.

Lã, Q.D., Chew, Y.H., and Soong, B.H. (2016). *Potential Game Theory.* Springer.

Larsson, E., Sennton, G., and Larson, J. (2015). The vehicle platooning problem: Computational complexity and heuristics. *Transportation Research Part C: Emerging Technologies*, 60, 258–277.

Li, N. and Marden, J.R. (2013). Designing games for distributed optimization. *IEEE Journal of Selected Topics in Signal Processing*, 7(2), 230–242.

Liang, K., Mårtensson, J., and Johansson, K.H. (2016). Heavy-duty vehicle platoon formation for fuel efficiency. *IEEE Transactions on Intelligent Transportation Systems*, 17(4), 1051–1061.

Marden, J., Arslan, G., and Shamma, J. (2009). Joint strategy fictitious play with inertia for potential games. *IEEE Transactions on Automatic Control*, 54, 208–220.

Marden, J. and Shamma, J. (2012). Revisiting log-linear learning: Asynchrony, completeness and payoff-based implementation. *Games and Economic Behavior*, 75(2), 788–808.

Nisan, N., Schapira, M., and Zohar, A. (2008). Asynchronous best-reply dynamics. In C. Papadimitriou and S. Zhang (eds.), *Internet and Network Economics*, volume 5385 of *Lecture Notes in Computer Science.* Springer, Berlin, Heidelberg.

Tsugawa, S., Jeschke, S., and Shladover, S.E. (2016). A review of truck platooning projects for energy savings. *IEEE Transactions on Intelligent Vehicles*, 1(1), 68–77.

van de Hoef, S., Johansson, K.H., and Dimarogonas, D.V. (2018). Fuel efficient en route formation of truck platoons. *IEEE Transactions on Intelligent Transportation Systems*, 19(1), 102–112.

Young, H.P. (2005). *Strategic Learning and its Limits.* Oxford University Press.

[3] a_{-i} and $\mathcal{A}_{-i}(x)$ are congruent to s_{-i} and S_{-i}, the strategies and set of available strategies, respectively, of all but i players.