21

A New Class of Control Systems Based on Non-equilibrium Games*

Yifen Mu and Lei Guo

Key Laboratory of Systems and Control, ISS, Academy of Mathematics and Systems
Science, Chinese Academy of Sciences, Beijing 100190, P.R. China

Summary. In this paper, a new class of control systems based on non-equilibrium dynamic games is introduced. Specifically, we consider optimization and identification problems modeled by an infinitely repeated 2×2 generic games between a human and a machine, where the machine takes a fixed and k-step-memory strategy while the human is more intelligent in the sense that she can optimize her strategy. This framework is beyond the frameworks of both the traditional control theory and game theory. By using the concept of state transfer graphs, the optimal strategy of the human will be characterized and the win-lose situation will be discussed. These are carried out for three typical games, i.e., the Prisoner's Dilemma game, the Snowdrift game and the Battle-of-sex game, but with different win-loss results. The problem of strategy identification will also be investigated.

21.1 Introduction

The current theoretical framework for control systems mainly aims at designing control laws for dynamical systems to achieve a certain prescribed performance (e.g. stability, optimality and robustness, etc.). In the control process, the systems (or plants) to be controlled are completely "passive" in an essential way, in the sense that they have no intension to compete with the controllers to achieve their own objectives or "payoffs". This is so even when the structures or parameters of the dynamical systems under control are uncertain and changing in time, because they are again of "passive" character. In these cases, the controllers can be made adaptive by incorporating certain online estimation algorithms, see e.g. [1]-[6].

However, in many practical systems, especially social, economical, biological and ecological systems, which involve adaptation and evolution, people often encounter with the so-called complex adaptive systems (CAS) as described in, e.g. [7]. In a CAS, as summarized in [8], a large number of components, called agents, interact with and adapt to (or learn) each other and their environment actively, leading

*This work was supported by the National Natural Science Foundation of China under grant 60821091 and by the Knowledge Innovation Project of CAS under Grant KJCX3-SYW-S01.

X. Hu et al. (Eds.): Three Decades of Progress in Control Sciences, pp. 313–326, 2010.
© Springer Berlin Heidelberg 2010

to some (possibly unexpected) macro phenomena called emergence. Despite of the flexibility in modeling a wide class of complex systems by CAS, it brings great challenge to understand the evolution of a CAS mathematically, since the traditionally used mathematical tools appear to give limited help in the study of CAS, as pointed out in [8].

As an attempt towards initiating a theoretical investigation for dynamical systems when some parts of the plants to be controlled have intentions to gain their own payoffs, we will, in this paper, consider a dynamic game framework that is somewhat beyond the current control theoretical framework. Intuitively, we will consider a simple scenario where we have two heterogeneous agents (players) in a system playing a repeated noncooperative game [9], where each agent makes its decision based on the previous actions and payoffs of both agents, but the law for generating the actions of one agent is assumed to be fixed. Thus, it may still be referred as a "control system" from the other agent's standpoint.

We would like to point out that, to the best of the authors' knowledge, the above non-equilibrium dynamic game framework seems to be neither contained in the traditional control theory, nor considered in the classical game theory. In fact, in the classical framework of game theory, all agents (players) stand in a symmetric position in rationality, in order to reach some kind of equilibrium; whereas, in our framework, the agents do not share a similar mechanism for decisions making and do not have the same level of rationality. This difference is of fundamental importance, since in many complex systems, such as non-equilibrium economy (e.g. see [10]), the agents are usually heterogenous, and they may indeed differ in either their information obtained or their ability in utilizing it.

We would also like to mention that there have been considerable investigations in game theory in relation to adaptation and learning, which can be roughly divided into two directions. One is called evolutionary game theory (e.g. see [11] and [12]) in which all agents in a large population are programmed to use certain actions to play. An action will spread or diminish according to the value of its corresponding payoff. The other direction is called learning in game theory, see, e.g. [13], [14]], which considers whether the long-run behaviors of individual agents will arrive at some equilibrium [15]-[16]. In both the directions, all the agents in the games are equal in their ability to learn or adapt to the strategies of their opponents. Some recent works and reviews can be found in [17]-[21]. The dynamic game framework to be studied in this paper is partly inspired by the evolutionary game framework in [22], where the best strategy is emerged as a result of evolution, while our optimal strategy to be studied in the next sections will be obtained by optimization and identification.

More specifically, we will consider infinitely repeated games between a human (or "controller") and a machine (or "plan") based on a generic 2×2 game model, which includes standard games such as Prisoners' Dilemma, Snowdrift, and Battle of Sex. The machine's strategy is assumed to be fixed with k-step memory, which may be unknown to the human.

To this end, we need to analyze the state transfer graph for machine strategy with k-memory. We will show that, similar to the Prisoners' Dilemma game as studied recently in [25], the optimal strategy for the current generic games that maximizes the human's averaged payoff is also periodic after finite steps. However, different from the result of [25], even for the case of $k = 1$, the human may lose to the machine while optimizing his own averaged payoff in the Snowdrift game, and the similar conclusion will depend on more conditions on the parameters of the Battle of Sex game. When the machine's strategy is unknown to the human, we will give a necessary and sufficient condition for identifiability, and will investigate the consequences of identification in our non-equilibrium dynamic game problem. Finally, we will discuss possible extensions to games with 2-players but with 3-actions.

The remainder of this paper is organized as follows. In Section 21.2, the main theorems will be stated following the problem formulation; In Section 21.3, the state transfer graph (STG) will be described and some useful properties will be studied. The proofs of some theorems will be given in Section 21.4, and Section 21.5 extends the modeling to games of 2-players with 3-actions. Finally, Section 21.6 will conclude the paper with some remarks.

21.2 Problem Statement and Main Results

Consider a generic 2×2 game with its payoff matrix described in figure 21.1. This matrix can be used to describe many standard games, either symmetric or asymmetric, when the parameters satisfy certain conditions.In the symmetric case, the payoff matrix can be specified as in figure 21.2.

		Player II	
		A	B
Player I	A	(a_{11}, b_{11})	(a_{12}, b_{12})
	B	(a_{21}, b_{21})	(a_{22}, b_{22})

Fig. 21.1. The payoff matrix of the generic 2×2 game

		Player II	
		A	B
Player I	A	(a, a)	(c, b)
	B	(b, c)	(d, d)

Fig. 21.2. The payoff matrix of the symmetric 2×2 game

.

One well-known example is the Prisoners' Dilemma game where the parameters satisfy $c > a > b > d$ and $2 \cdot a > b + c$, while the actions "A" and "B" mean "Cooperate" and "Defect" respectively.

Another typical example is the Snow Drift game. In this game, two players, called Player 1 and 2, can be two drivers who are on their way home, caught by the snowdrift and thus must decide whether or not to shovel it. They simultaneously choose their actions A or B, where "A" means the player will shovel the snow on the road, and "B" means the player will not. Different action profiles will result in different payoffs for the players. The parameters in the payoff matrix of this game satisfy $d = 0 < c < a < b$.

As for the asymmetric case, the game of Battle of Sex is a typical example (see figure 21.3). Here, the Player 1 can be assumed to be the wife while the Player 2 be the husband, with the action A may stand for watching the ballet while the action B may stand for watching the football. The parameters are assumed to satisfy $a_{21} = b_{21} = 0, a_{11} > a_{12} > 0, a_{11} > a_{22} > 0$, and $b_{22} > b_{11} > 0, b_{22} > b_{12} > 0$. Without loss of generality, we may specify the matrix as follows where $a > b > 0$, $a > c > 0$:

		Player II	
		A	B
Player I	A	(a, b)	(c, c)
	B	(0, 0)	(b, a)

Fig. 21.3. The payoff matrix of the Battle of Sex game

From the parameter inequalities, it is easy to compute the Nash Equilibria of these games. Our purpose is, however, not to investigate the Nash equilibrium in the game theory. Instead, we will consider the scenario where Player 1 has the ability to search for the best strategy so as to optimize his payoff, while Player 2 acts according to a given strategy. Clearly, this non-equilibrium dynamic game problem is different from either the standard control problem or the classical game problem, and thus may be regarded as a new class of "control systems". A preliminary study was initiated recently for the Prisoners' Dilemma game in [25], where some basic notations and ideas will be adopted in what follows.

Vividly, let Player 1 be a human (we say it is a "he" henceforth) while his opponent Player 2 is a machine. Assume they both know the payoff matrix. The action set of both players is denoted as $\mathscr{A} = \{A, B\}$, and the time set is discrete, $t = 0, 1, 2, \ldots$. At time t, both players will choose their actions and get their payoffs simultaneously. Let $h(t)$ denote the human's action at t and $m(t)$ the machine's.

Define the history of time t, H_t, as the sequence of two players' action profiles before time t i.e.

$$H_t \triangleq (m(0),h(0);m(1),h(1);...;m(t-1),h(t-1)).$$

Denote the set of all histories for all time t as $H = \bigcup_t H_t$.

As a start, we consider the case of pure strategy and define the **strategy** of either player as a function $f:\ H \to \mathscr{A}$. In this paper, we will further confine the machine's strategy with finite k-memory as follows:

$$m(t+1) = f(m(t-k+1),h(t-k+1);...;m(t),h(t)) \tag{21.1}$$

which, obviously, is a discrete function from $\{0,1\}^{2k}$ to $\{0,1\}$, where and hereafter, 0 and 1 stands for A and B respectively. Moreover, the following mapping can establish a one-to-one correspondence between the vector set $\{0,1\}^{2k}$ and the integer set $\{1,2,......2^{2k}\}$:

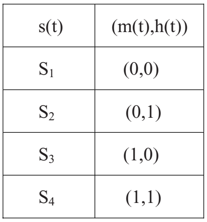
$$s(t) = \sum_{l=0}^{k-1} \{2^{2l+1} \cdot m(t-l) + 2^{2l} \cdot h(t-l)\} + 1 \tag{21.2}$$

For convenience, in what follows we will denote $s_i = i$ and call it a **state** of the game under the given strategies.

In the simplest case where $k = 1$, the above mapping reduces to

$$s(t) = 2 \cdot m(t) + h(t) + 1, \tag{21.3}$$

which establishes a one-to-one correspondence between the value set $s(t) \in \{s_1, s_2, s_3, s_4\}$ with $s_i = i$ and $(m(t),h(t))$:



s(t)	(m(t),h(t))
S_1	(0,0)
S_2	(0,1)
S_3	(1,0)
S_4	(1,1)

and the machine strategy (21.1) can be written as

$$\begin{aligned} m(t+1) &= f(m(t),h(t)) \\ &= a_1 I_{\{s(t)=s_1\}} + ... + a_4 I_{\{s(t)=s_4\}} \\ &= \sum_{i=1}^{4} a_i I_{\{s(t)=s_i\}} \end{aligned} \tag{21.4}$$

which can be simply denoted as a vector $A = (a_1, a_2, a_3, a_4)$ with a_i being 0 or 1.

Given any strategies of both players together with any initial state, the game will be carried on and a unique sequence of states $\{s(1), s(2), ...\}$ will be produced. Such a sequence will be called a **realization** [15].

Obviously, each state $s(t)$ corresponds to a pair $(m(t),h(t))$, and so by the definition of the payoff matrix, the human and the machine will obtain their payoffs, denoted by $p(s(t))$ and $p_m(s(t))$, respectively. Let us further define the extended payoff vector for the human as **P(s(t))** $\triangleq (p(s(t)),w(s(t)))$, where $w(s(t))$ indicates the relative payoff to the machine at t, i.e.,

$$w(s(t)) = sgn\{p(s(t)) - p_m(s(t))\} \triangleq w(t), \tag{21.5}$$

where $sgn(\cdot)$ is the sign function and $sgn\{0\} = 0$.

For the above infinitely repeated games, the human may only observe the payoff vector **P(s(t))**, but since there is an obvious one-to-one correspondence between **P(s(t))** and $s(t)$, we will assume that $s(t)$ is observable to the human at each time t throughout the paper.

Now, for any given human and machine strategies with their corresponding realization, the **averaged payoff** (or ergodic payoff) [23] of the human can be defined as

$$P_\infty^+ = \overline{\lim_{T\to\infty}} \frac{1}{T}\sum_{t=1}^{T} p(t). \tag{21.6}$$

In the case where the limit actually exists, we may simply write $P_\infty^+ = P_\infty$. Similarly, W_∞^+ can be defined.

The basic questions that we are going to address are as follows:

1. How can the human choose his strategy g so as to obtain an optimal averaged payoff?
2. Is the human's optimal strategy necessarily gives a payoff that is better than the machine's?
3. Can the human still obtain an optimal payoff when the machine's strategy is unknown to him?

The following theorems and proposition will give some answers to these questions.

Theorem 21.2.1. *Consider the generic* 2×2 *game described in Fig 1, and any machine strategy with finite k-memory. Then, there always exists a human strategy also with k-memory, such that the human's payoff is maximized and the resulting system state sequence* $\{s(t)\}$ *will become periodic after some finite time.*

The proof of Theorem 21.2.1 is just the same as that in [25] for the case of the Prisoners' Dilemma game, so we refer the readers to [25] for the proof details. Also, One can see from the proof that the optimal payoff values will remain the same for different initial values of the state transfer graph (STG), as long as they share the same reachable set. In particular, this observation is true when the STG is strongly connected, see Section 21.3 for the definition of STG.

Moreover, as will be illustrated by Example 21.3.1, Theorem 21.2.1 will enable us to find the optimal human strategy by searching on the STG with considerably reduced computational complexity. Furthermore, since Theorem 21.2.1 only concerns with the properties of the optimal human trajectory, a natural question is: whether or not the human's optimal averaged payoff value is better than that of the machine's. This is a subtle question, and will be addressed in the following theorem.

Theorem 21.2.2.

1. *For the standard Prisoners' Dilemma game, the optimal strategy of the human will not lose to any machine whose strategy is of 1-memory. However, when $k > 1$, there exists such machine strategies, that the human's optimal strategy will lose to them.*
2. *For the Snowdrift game, there exists such a machine strategy with 1-memory, that the optimal strategy of the human will lose to the machine.*
3. *For the game of Battle of Sex, whether or not the human will always win the machine with 1-memory is indefinite, i.e., it depends on more conditions on the payoff parameters.*

Remark 21.2.1.

(1) For the Prisoners' Dilemma game, when $k \geq 2$, the game becomes more complicated and subtle. As demonstrated in Section 21.4 of [25], whether the human can win while getting his optimal payoff depends on delicate relationships among s, p, r, t.

(2) Theorem 21.2.2 (2) will remain valid for the machine strategy with k-memory in general, since $k = 1$ is a special case.

Remark 21.2.2. As has been noted in [25], it is the game structure that brings about a somewhat unexpected win-loss phenomenon: such an one-sided optimization problem (for the human) may not always win even if the opponent has a fixed strategy. Similar phenomena do exist practically, but, of course, cannot be observed in the traditional framework of optimal control. We would also like to note that the differences among the results of the three games can be attributed to the differences in the game structures.

As will be shown in Section 21.3, when the machine strategy is known to the human, the human can find the optimal strategy with the best payoff. A natural question is: What if the machine strategy is unknown to the human?

One may hope to identify the machine strategy within finite steps before making optimal decision. A machine strategy which is parameterized by a vector A (like in (21.4) for the case of $k = 1$), is called **identifiable** if there exists a human strategy such that the vector A can be constructed from the corresponding realization and the initial state.

Proposition 21.2.1. *A machine strategy with k-memory is identifiable if and only if its corresponding STG is strongly connected.*

Proposition 21.2.1 is somewhat intuitive, which can be used to identify non-identifiable machine strategies. Consider the simple case where $k = 1$. Then it is easy to see that the STG corresponding to the machine strategy $A = (0,0,*,*)$ or $A = (*,*,1,1)$

will not be strongly connected, and so will not be identifiable by Proposition 21.2.1. In fact, as can be easily seen, only part of the entries of such $A = (a_1, a_2, a_3, a_4)$ can be identified from any given initial state.

If the machine makes mistakes with a tiny possibility, however, the machine strategy may become identifiable. For example, if it changes its planed decision with a small positive probability to any other decisions, then the corresponding STG will be a Markovian transfer graph which is strongly connected. Hence, all strategies will be identifiable.

To illustrate how to identify the machine strategy, let us again consider the case of $k = 1$. In this case, one effective way for the human to identify the machine strategy is to randomly choose his action at each time. One can also use the following method to identify the parameters:

$$h(t+1) = \begin{cases} 0 & a_{s(t)} \textit{ is not known at time } t, \\ & \textit{or } a_{s(t)} \textit{ is known, but } a_{2\cdot a_{s(t)}+1} \textit{ is not}; \\ 1 & \textit{otherwise}. \end{cases} \tag{21.7}$$

Theorem 21.2.3. *For any identifiable machine strategy with $k = 1$, it can be identified using the above human strategy with at most 7 steps from any initial state.*

Remark 21.2.3. For non-identifiable machine strategies, one may be surprised by the possibility that identification may lead to a worse human's payoff. We have shown that this is true for the PD game [25]. It is true for the Snow drift game too. For example, if the machine takes the non-identifiable strategy $A = (0, 1, 1, 1)$, then by acting with "*A*" blindly, the human can get a payoff a by the payoff matrix at each time. However, once he tries to identify the machine's strategy, he may use the "*B*" to probe it. Then the machine will be provoked and act with "*B*" forever. That will lead to a worse human payoff $c < a$ afterwards.

21.3 The State Transfer Graph

In order to provide the theoretical proofs for the main results stated in the above section, we need to use the concept of State Transfer Graph (STG) together with some basic properties, as in the paper [25]. Throughout this section, the machine strategy $A = (a_1, a_2, a_3, a_4)$ is assumed to be known.

Given an initial state and a machine strategy, any human strategy $\{h(t)\}$ can lead to a realization of the states $\{s(1), s(2), ..., s(t), ...\}$. Hence, it also produces a sequence of human payoffs $\{p(s(1)), p(s(2)), ..., p(s(t)), ...\}$. Thus the Question 1) raised in Section 21.2 becomes to solve

$$\{h(t)\}_{t=1}^{\infty} = \arg\max P_{\infty}^{+}$$

among all possible human strategies.

In order to solve this problem, we need the definition of STG, and we refer to [24] for some standard concepts in graph theory, e.g. walk, path and cycle. We will only consider finite graphs (with finite vertices and finite edges) in the sequel.

Let $G = (V,E)$ be a directed graph with vertex set V and edge set E.

Definition 21.3.1. *A* ***walk*** *W is defined as an alternating sequence of vertices and edges, like $v_0e_1v_1e_2...v_{l-1}e_lv_l$, abbreviated as $v_0v_1...v_{l-1}v_l$, where $e_i = \overline{v_{i-1}v_i}$ is the edge from v_{i-1} to vi, $1 \le i \le l$. The total number of edges l is called the length of W.*

If $v_0 = v_l$, then W is called closed, otherwise is called open.

Definition 21.3.2. *A walk W, $v_0v_1...v_{l-1}v_l$, is called a* ***path*** *(directed), if the vertices $v_0,v_1,...v_l$ are distinct.*

Definition 21.3.3. [1] *A closed walk W: $v_0v_1...v_{l-1}v_l$, $v_0 = v_l$, $l \ge 1$, is called a* ***cycle*** *if the vertices $v_1,...,v_l$ are distinct.*

Definition 21.3.4. *A graph is called* ***strongly connected*** *if for any distinct vertices v_i,v_j, there exists a path starting from v_i and ending with v_j.*

Now, we are in a position to define the STG.

Note that any given machine strategy of k-memory, together with a human strategy, will determine an infinite Walk representing the state transfer process of the game.

Definition 21.3.5. *A directed graph with 2^{2k} vertices $\{s_1,s_2,......s_{2^{2k}}\}$ is called the* ***State Transfer Graph*** *(STG), if it contains all the possible infinite walks corresponding to all possible human strategies, that equals to say, it contains all the possible one-step path or cycle in the walk.*

In the case of $k = 1$, for a machine strategy $A = (a_1,a_2,a_3,a_4)$, the STG is a directed graph with the vertices being the state $s(t) \in \{s_1,s_2,s_3,s_4\}$ with $s_i = i$.

An edge $\overline{s_is_j}$ exists if $s(t+1) = s_j$ can be realized from $s(t) = s_i$ by choosing $h(t+1) = 0$ or 1. Since $s_i = i$, by (21.3) and (21.4), that means,

$$\text{the edge } \overline{s_is_j} \text{ exists} \Leftrightarrow s_j = 2\cdot a_i + 1 \text{ or } s_j = 2\cdot a_i + 2 \tag{21.8}$$

and the way to realize this transfer is taking human's action as $h = (s_j - 1) mod\ 2$ by (21.3).

By the definition above, one machine strategy leads to one STG, and vice versa.

Definition 21.3.6. *A state s_j is called* ***reachable*** *from the state s_i, if there exists a path (or cycle) starting from s_i and ending with s_j. All the vertices which are reachable from s_i constitute a set, called the* ***reachable set*** *of the state s_i. A STG is called* ***strongly connected*** *if any vertex s_i has all vertices in its reachable set.*

[1] The Definition 21.3.3 of cycle is a little different from [24]. We ignore the constraint that the length $l \ge 2$ and include 'loop' in the concept of "cycle".

Thus, the reachability of s_j from s_i means that there exists a finite number of human actions, such that the state $s(\cdot)$ can be transferred from s_i to s_j with the same number of steps.

Furthermore, we need to define the payoff of a walk on STG as follows:

Definition 21.3.7. *The averaged payoff of an open walk $W = v_0v_1...v_l$ on a STG, with $v_0 \neq v_l$, is defined as*

$$p_W \triangleq \frac{p(v_0)+p(v_1)+...+p(v_l)}{l+1}, \tag{21.9}$$

and the averaged payoff of a closed walk $W = v_0v_1...v_l$, with $v_0 = v_l$, is defined as

$$p_W \triangleq \frac{p(v_0)+p(v_1)+...+p(v_{l-1})}{l}. \tag{21.10}$$

Now, we can give some basic properties of STG below.

Lemma 21.3.1. *For a given STG, any closed walk can be divided into finite cycles, such that the edge set of the walk equals the union of the edges of these cycles. In addition, any open walk can be divided into finite cycles plus a path.*

Lemma 21.3.2. *Assume that a closed walk $W = v_0v_1...v_L$ with length L, can be partitioned into cycles $W_1,W_2,...,W_m, m \geq 1$, with their respective lengths being $L_1,L_2,...,L_m$. Then, p_W , the averaged payoff of W can be written as*

$$p_W = \sum_{j=1}^{m} \frac{L_j}{L} p_j, \tag{21.11}$$

where $p_1,p_2,...,p_m$ are the averaged payoffs of the cycles $W_1,W_2,...,W_m$, respectively.

By Theorem 21.2.1, the state of the repeated games will be periodic under the optimal human strategy. This enables us to find the optimal human strategy by searching on the STG, as will be illustrated in the example below. Similar to [25], we give an example for the Snowdrift game.

Example 21.3.1. Consider the “ALL A” strategy $A = (0,0,0,0)$ of the machine. Then the STG can be drawn as shown in Figure 21.4, in which $s_1(a,0)$ means that under the state s_1, the human gets his payoff vector $P(s_1) = (p(s_1),w(s_1)) = (a,0)$. The directed edge $\overline{s_1s_2}$ illustrates that if the human takes action D, he can transfer the state from s_1 to s_2 with payoff vector $(b,1)$. Others can be explained in the same way. Now we take the initial state as $s(0) = s_3 = (c,-1)$.Then the reachable set of s_3 is $\{s_1,s_2\}$, and we just need to search the cycle whose vertices are on this set.

Obviously, there are three possible cycles $W_1 = \{s_1\}$, $W_2 = \{s_2\}$, $W_3 = \{s_1,s_2\}$ and by (21.10), the averaged payoffs of the human are respectively $p_{W_1} = p(s_1) = a$, $p_{W_2} = p(s_2) = b$, $p_{W_3} = \frac{p(s_1)+p(s_2)}{2} = \frac{a+b}{2}$.

Obviously, the optimal payoff lies in the cycle $W_2 = \{s_2\}$. To induce the system state enters into this cycle, the human just take $h(1) = 1$. Then by taking $h(t) = 1, t \geq 2$, the optimal state sequence $s(t) = s_2, t \geq 1$ will be obtained from $s(0) = s_3$.

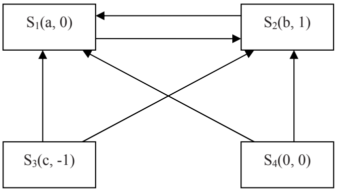


Fig. 21.4. STG of ALL A machine strategy $A = (0,0,0,0)$ in Snowdrift game

Remark 21.3.1. The search procedure above can be accomplished in the general case by an algorithm which is omitted here for brevity, and it can also be seen that for any given machine strategy with k-memory, there always exists a search method to find the optimal strategy of the human. Moreover, the optimal payoff remains the same when the initial state varies over a reachable set.

21.4 Proofs of the Main Results

First of all, it is not difficult to see that in the current general case, Theorem 21.2.1, Proposition 21.2.1 and Theorem 21.2.3 can be proven along the proof lines of those in [25], and so the details will be omitted here.

Remark 21.4.1. It is worth mentioning that the form of the averaged payoff criteria is important in Theorem 21.2.1. For other payoff criteria, similar results may not hold.

As for the proof of Theorem 21.2.2, the first conclusion on the Prisoners' Dilemma game can be seen in [25], and so we just need to prove the conclusions (2) and (3).

Proof of Theorem 21.2.2 (2).
The conclusion will be proven if we can find the required machine strategy. To this end, we just need take the "ALL B" strategy $(1,1,1,1)$ as the machine's strategy. Then starting from any initial state, to optimize his payoff value, the human has to take the action "A" always, which will lead to a payoff c for him while the machine will get b. Hence he will lose.

Proof of Theorem 21.2.2 (3).
For the game of the Battle of Sex, consider the following two cases:

Case 1: when $b > c$, the pure Nash equilibria of the game are the profile (A,A) and (B,B);

Case 2: when $b < c$, the pure Nash equilibrium of the game is the profile (A,B).

Then in Case 1, if the machine takes the "Always B" strategy $(1,1,1,1)$, then the optimal human strategy will always act "B" too. Thus the state will repeat the profile (B,B) and the human will get a payoff of b while the machine get a, which implies that the human will lose. In Case 2, similar to the proof of Theorem 21.2.2 in [25], we can prove that the human cannot lose to the machine in this case. This completes the proof of Theorem 21.2.2. □

Remark 21.4.2. Note that all the three games in Theorem 21.2.2 have some characters about “need for coordination”, and the differences in the three assertions of Theorem 21.2.2 result from the different game structures. Specifically, the Snowdrift game has a more cruel assumption, i.e. the player must never choose “both B” which is the worst case for both, while the “A” player have to sacrifice in a (A,B) profile. For the Battle of Sex game, we can imagine that the parameters may measure whether the two players care more about the time they share or care more about their own interest. If they care more about the sharing time, i.e. when $b > c$, then the selfisher one can use this feature to win.

21.5 Extensions to 3×3 Matrix Games

In this section, we consider possible extensions of the results in the previous sections. Consider the 2-player 3-action games, in which there are 2 players while either one has three actions. The payoff matrix is then as in figure 21.5

		Player II		
		A	B	C
Player I	A	(a_{11}, b_{11})	(a_{12}, b_{12})	(a_{13}, b_{13})
	B	(a_{21}, b_{21})	(a_{22}, b_{22})	(a_{23}, b_{23})
	C	(a_{31}, b_{31})	(a_{32}, b_{32})	(a_{33}, b_{33})

Fig. 21.5. The payoff matrix of 2-player 3-action game

Similar to Section 21.2, we can formulate a repeated game and describe the corresponding dynamic rules by a STG. To this end, we need to define the system state first. For the 1-memory machine strategy, there are 3 actions for each player, which can be denoted as 0,1,2 like a ternary signal. So there are $3 \times 3 = 9$ 1-memory histories, and thus we can define the state as

$$s(t) = 3 \cdot m(t) + h(t) + 1, \tag{21.12}$$

and the machine strategy can be written as

$$m(t+1) = f(m(t), h(t)) = \sum_{i=1}^{9} a_i I_{\{s(t)=s_i\}} \tag{21.13}$$

Thus, the STG with have 9 vertices and can be formed and analyzed by similar methods as those in Section 21.3. It can be easily seen that Theorem 21.2.1 and Proposition 21.2.1 will hold true in this case since the proofs only use the finite state information. However, Theorem 21.2.2 must be checked for specific games and Theorem 21.2.3 must be modified for this kind of 2-player 3-action games. Also, extensions to 2-player *n*-action games can be carried out in a similar way.

A well-known example of 2-player 3-action games is the "Rock-Paper-Scissors" game whose payoff matrix can be specified by as in figure 21.6. This game is a zero-sum game, and the relationship between the optimality and the wining of the human is consistent. In fact, the human can select one from the three actions to beat his opponent, and the game is like history independent. So, once the machine's strategy is known, the human can always get his optimal payoff and win at the same time.

		Player II		
		rock	paper	scissors
Player I	rock	(0, 0)	(-1, 1)	(1, -1)
	paper	(1, -1)	(0,0)	(-1,-1)
	scissors	(-1, 1)	(1, -1)	(0, 0)

Fig. 21.6. The payoff matrix of "Rock-Paper-Scissors" game

21.6 Concluding Remarks

In an attempt to study dynamical control systems which contain game-like mechanisms in the system structure, we have, in this paper, presented a preliminary investigation on optimization and identification problems for a specific non-equilibrium dynamic game where two heterogeneous agents, called "Human" and "Machine", play repeated games modeled by a generic 2×2 game. Some typical games including the Prisoner Dilemma game, Snowdrift game and the Battle of Sex game have been studied in certain detail. By using the concept and properties of the state transfer graph, we are able to establish some interesting theoretical results, which have not been observed in the traditional control framework. For example, we have shown that the optimal strategy of the game will be periodic after finite steps, and that optimizing one's payoff solely may lose to the opponent eventually. Possible extensions to more general game structures like 2-player 3-action games are also discussed. It goes without saying that there may be many implications and other extensions of these results. However, it would be more challenging to establish a mathematical theory for more complex systems, where many (possibly heterogeneous) agents interact with learning and adaptation, cooperation and competition, etc.

References

1. Astrom, K.J., Wittenmark, B.: Adaptive Control, 2nd edn. Addison-Wesley, Reading (1995)
2. Chen, H.F., Guo, L.: Identification and Stochastic Adaptive Control. Birkhäuser, Boston (1991)
3. Goodwin, G.C., Sin, K.S.: Adaptive Filtering, Prediction and Control. Prentice-Hall, Englewood Cliffs (1984)
4. Kumar, P.R., Varaiya, P.: Stochastic Systems: Estimation, Identification and Adaptive Control. Prentice Hall, Englewood Cliffs (1986)
5. Kristic, M., Kanellakopoulos, I., Kokotoric, P.: Nonlinear Adaptive Control Design. A Wiley-Interscience Publication, John Wiley & Sons, Chichester (1995)
6. Guo, L.: Adaptive Systems Theory: Some Basic Concepts, Methods andResults. Journal of Systems Science and Complexity 16, 293–306 (2003)
7. Holland, J.: Hidden Order: How Adaptation Builds Complexity. Addison-Wesley, Reading (1995)
8. Holland, J.: Studying Complex Adaptive Systems. Journal of System Science and Complexity 19, 1–8 (2006)
9. Basar, T., Olsder, G.J.: Dynamic Noncooperative Game Theory, the Society for Industrial Applied Mathematics. Academic Press, New York (1999)
10. Arthur, W.B., Durlauf, S.N., Lane, D.: The Economy As An Evolving Complex System II. Addison-Wesley, Reading (1997)
11. Weibull, J.W.: Evolutionary Game Theory. MIT Press, Cambridge (1995)
12. Hofbauer, J., Sigmund, K.: Evolutionary game dynamics. Bulletin of the American Mathematical Society 40, 479–519 (2003)
13. Fudenberg, D., Levine, D.K.: The Theory of Learning in Games. MIT Press, Cambridge (1998)
14. Fudenberg, D., Levine, D.K.: Learning and equilibrium (2008), Available: `http://www.dklevine.com/papers/annals38.pdf`
15. Kalai, E., Lehrer, E.: Rational learning leads to Nash equilibrium. Econometria 61, 1019–1045 (1993)
16. Kalai, E., Lehrer, E.: Subjective equilibrium in repeated games. Econometrica 61, 1231–1240 (1993)
17. Marden, J.R., Arslan, G., Shamma, J.S.: Joint strategy fictitious play with inertia for potential games. IEEE Trans. Automatic Control 54, 208–220 (2009)
18. Foster, D.P., Young, H.P.: Learning, hyperthesis testing ans Nash equilibrium. Games and Economic Behavior 45, 73–96 (2003)
19. Marden, J.R., Young, H.P., Arslan, G., Shamma, J.S.: Payoff based dynamics for multiplayer weakly acyclic games. In: Prodeedings of the 46th IEEE Conferrence on Decision and Control, New Orleans, USA, pp. 3422–3427 (2007)
20. Chang, Y.: No regrets about no-regret. Artificial Intelligence 171, 434–439 (2007)
21. Young, H.P.: The possible and the impossible in multi-agent learning. Artificial Intelligence 171, 429–433 (2007)
22. Axelrod, R.: The Evolution of Cooperation. Basic Books, New York (1984)
23. Puterman, M.: Markov Decision Processes:Discrete Stochastic Dynamic Programming. John Wiley & Sons, New York (1994)
24. B-Jensen, J., Gutin, G.: Digraphs: Theory, Algorithms and applications. Springer, London (2001)
25. Mu, Y.F., Guo, L.: Optimization and Identification in a Non-equilibrium Dynamic Game. In: Prodeedings of the 48th IEEE Conferrence on Decision and Control, Shanghai, China, December 16-18 (2009)