K-Memory Strategies in Repeated Games

(Extended Abstract)

Lijie Chen† wjmzbmr@gmail.com
Fangzhen Lin‡ flin@cs.ust.hk
Pingzhong Tang† kenshinping@gmail.com
Kangning Wang† wkn13@mails.tsinghua.edu.cn
Ruosong Wang† wrs13@mails.tsinghua.edu.cn
Shiheng Wang‡ swangbv@connect.ust.hk

†Institute of Interdisciplinary Information Sciences, Tsinghua University, Beijing, China

‡Department of Computer Science and Engineering, HKUST, Hong Kong, China

ABSTRACT

In this paper, we study k-memory strategies in two-person repeated games. An agent adopting such a strategy makes his decision only based on the action profiles of the previous k rounds. We show that in finitely repeated games, the best response of a k-memory strategy may not even be a constant-memory strategy. However, in infinitely repeated games, one of the best responses against any given k-memory strategy must be k-memory. Our results are enabled by a graph-structural characterization of the best responses of k-memory strategies. We put forward polynomial algorithms to compute best responses.

Keywords

bounded rationality, repeated games, k-memory strategies, efficient algorithms

1. INTRODUCTION

Bounded rationality has been a topic of extensive interest in artificial intelligence and multi-agent system research [8, 9, 2, 19, 20, 3, 18]. The notion of bounded rationality refers to the limitations (time, space, information, etc) agents encounter that prevent them from making a fully rational decision in reality. This phenomenon has been widely studied in the realm of repeated games (cf. eg. [13, 17]).

In repeated games, to describe a strategy, a player needs to specify his action choice for any possible history. This leads to the difficulty that the description of a general strategy is exponential in the game size and is thus highly unrealistic. To mitigate this difficulty, stylized approach models them as *finite automata strategies* [15, 13], where "equivalent" histories are grouped into states in the automata. Under this compact formulation, the set of equilibria has been characterized in [15]. This topic has been further investigated in [5, 1, 21].

The second approach to model bounded rationality of agents is by Turing machine strategies. In reality, agents could write programs to compute strategies and this inspired researchers to consider the possibility of modeling bounded rational agents as general Turing machines. Megiddo and Wigderson [11] first model strategies as a general Turing machines. [7] and [12] show that in infinitely repeated games, there exists a Turing machine strategy such that no Turing machine can implement its best response. Chen et al. [4] studied *restricted Turing machine strategies*, i.e., agents could only use Turing machine strategies with limited amount of time and space.

Another natural approach to model bounded rationality is to put a limit on the size of agents' memory. Lindgren studied the effect of different memory size in repeated Prisoner's Dilemma in [10]. Hauert and Schuster [6] used numerical approach to study influences of increased memory sizes in the iterated two-player Prisoner's Dilemma. In a novel work by Press and Dyson [14], the authors showed that in two-player repeated Prisoner's Dilemma in which the agents could only remember the outcome of the last round (in fact, this is exactly 1-memory strategy in our model), one player can enforce a unilateral claim to an unfair share of rewards. The results by Press and Dyson are based on studying *zero-determinant strategies*.

In this paper, we explore this direction further, by studying a novel realistic model of bounded rationality (what we called "k-memory strategies") where the memory size of agents is bounded. Our model and results can be regarded as a natural extension of those by Press and Dyson [14], by studying the effect of a weaker memory constraint and its related computational issues. We first present a novel characterization of k-memory strategies via a graph model, called the "transition graph". Enabled by the characterization, we can deploy graph-theoretical methods to thoroughly study k-memory strategies.

2. PRELIMINARIES

When we talk about the payoff in an infinitely repeated game, we always refer to the infimum of the limit of means: $\liminf_{T\to\infty} \frac{1}{T}\sum_{i=1}^{T} p_i$, where p_i is the payoff at round i (like in [16]).

2.1 Definition of K-Memory Strategies

Definition 1. In a finitely or infinitely repeated game, a *k-memory strategy* of player A is a strategy in which A determines his action in the current round only based on the action profiles of previous k rounds.

Appears in: *Proc. of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2017), S. Das, E. Durfee, K. Larson, M. Winikoff (eds.), May 8–12, 2017, São Paulo, Brazil.*
Copyright © 2017, International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

Strategy	C	D
C	$(80, 80)$	$(0, 100)$
D	$(100, 0)$	$(30, 30)$

Table 1: The payoff matrix for Prisoner's Dilemma.

To illustrate, consider the famous Prisoner's Dilemma. In this game each player has two choices of action: C (for Cooperate) and D (for Defect). Table 1 shows a possible payoff matrix. Any 1-memory pure strategy (we will consider mixed strategies later) can be described as a strategy vector s in $\{C, D\}^4$ in the following way: (i) map each one-round history of action profiles to an integer in $\{0, 1, 2, 3\}$; (ii) for each history of action profiles h mapped to $i \in \{0, 1, 2, 3\}$, set s_i equal to the action performed by the strategy if the history of action profiles is h. For example, if we map (C, C) to 0, (C, D) to 1, (D, C) to 2 and (D, D) to 3, the vector (D, C, D, C) will correspond to the strategy in which the player

- plays D when the action profile of the previous round is (C, C) or (D, C), and
- plays C when the action profile of the previous round is (D, D) or (C, D).

2.2 Transition Graph Representation

When both players adopt k-memory strategies, we can draw a *transition graph* representing these two strategies.

Definition 2. A *transition graph* is a directed graph in which each vertex represents a possible k-round history, and each edge (u, v) exists if and only if the players will be in v in the next round given the current history to be u.

In a repeated game where each player has n actions to choose, there are n^{2k} vertices and n^{2k} edges in the transition graph.

To compute the result of this repeated game, we begin from the starting vertex, and go along the edges at each step. Once we have found a cycle, we know we are going to loop in the cycle forever. The average payoff in the cycle is the limit-of-means utility in an infinitely repeated game.

When only one player has fixed his strategy, we can draw an *incomplete transition graph* for it.

Definition 3. When one player has fixed his strategy to be s, an *incomplete transition graph* G_s is a directed graph in which each vertex represents a possible k-round history, and each edge (u, v) exists if and only if the other player can choose to be in v in the next round given the current history to be u,

The idea is, for each action profile, since the other player has not decided his strategy, there are several possible outcomes next round so we add one edge for each possibility. If each player has n actions to choose, there will be n^{2k} vertices and n^{2k+1} edges in the incomplete transition graph.

3. K-MEMORY STRATEGIES IN FINITELY REPEATED GAMES

With our previous knowledge, we are ready to show several properties of k-memory strategies in repeated games. We first consider how to compute the best response to a k-memory strategy in a finitely repeated game. Unfortunately, the best response is not necessarily of a k'-memory form.

THEOREM 1. *There exists a 1-memory strategy S such that in a finitely repeated game of T rounds, no best response to S is k-memory, for any constant integer k.*

However, we can still efficiently compute the best response to a given k-memory strategy, even enough such a strategy may not be of constant memory. Note that here we assume the k-memory strategy (to which we compute the best response) is given as a look-up table, i.e., the input is a table of size n^{2k}, where n is the number of possible actions for each player. Each entry corresponds to the action performed by the strategy in a specific history of action profiles.

THEOREM 2. *In a finitely repeated game of T rounds, where in each round each player has n possible actions, the best response to a k-memory strategy can be computed in $O(\min(n^{6k}\log T, Tn^{2k+1}))$ time.*

4. K-MEMORY STRATEGIES IN INFINITELY REPEATED GAMES

THEOREM 3. *In an infinitely repeated game, for each k-memory (pure or mixed) strategy, there exists a k-memory strategy finally looping in a simple cycle as a best (unrestricted) response.*

The following theorem shows that $(k-1)$-memory strategies are not enough to implement a best response in general. Therefore, an agent needs exactly a memory size of k to implement a best response to a given k-memory strategy.

THEOREM 4. *For every k, there exists a k-memory strategy in an infinitely repeated game, such that for any $k' < k$, there is no k'-memory strategy as its best response.*

We show that computing a best response to a k-memory strategy is efficiently doable. Again we assume the k-memory strategy is given as a look-up table as in Section 3.

THEOREM 5. *In an infinitely repeated game, where in each round each player has n possible actions, the best response to a pure or mixed k-memory strategy can be computed in time $n^{O(k)}$.*

5. CONCLUSIONS

In this paper, we studied k-memory strategies. We developed graph models to represent these strategies. Our graph tools enable was-complicated analysis of the structures of best responses and best commitments. We showed that in finitely repeated games, the best responses to k-memory strategies may not be constant-memory, but can be efficiently computed. In infinitely repeated games, best responses to k-memory strategies have to be k-memory, and is not always $(k-1)$-memory. Such strategies can be computed in polynomial time.

Acknowledgement

This work was supported by the National Basic Research Program of China Grant 2011CBA00300, 2011CBA00301, the Natural Science Foundation of China Grant 61033001, 61361136003, 61303077, 61561146398, a Tsinghua Initiative Scientific Research Grant and a China Youth 1000-talent program.

REFERENCES

[1] E. Ben-porath. The complexity of computing a best response automaton in repeated games with mixed strategies. *Games and Economic Behavior*, 2(1):1–12, 1990.

[2] R. Cavallo and D. C. Parkes. Efficient metadeliberation auctions. In *Proceedings of AAAI*, pages 50–56, 2008.

[3] L. E. Celis, A. R. Karlin, K. Leyton-Brown, C. T. Nguyen, and D. R. M. Thompson. Approximately revenue-maximizing auctions for deliberative agents. In *Proceedings of AAAI*, 2012.

[4] L. Chen, P. Tang, and R. Wang. Bounded rationality of restricted turing machines. In *Proceedings of AAAI*, 2017.

[5] I. Gilboa. The complexity of computing best-response automata in repeated games. *Journal of Economic Theory*, 45(2):342–352, 1988.

[6] C. Hauert and H. G. Schuster. Effects of increasing the number of players and memory size in the iterated prisoner's dilemma: a numerical approach. *Proceedings of the Royal Society of London B: Biological Sciences*, 264(1381):513–519, 1997.

[7] V. Knoblauch. Computable strategies for repeated prisoner' s dilemma. *Games and Economic Behavior*, 7(3):381–389, 1994.

[8] K. Larson and T. Sandholm. Experiments on deliberation equilibria in auctions. In *International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 394–401, 2004.

[9] K. Larson and T. Sandholm. Mechanism design and deliberative agents. In *International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 650–656, 2005.

[10] K. Lindgren. Evolutionary phenomena in simple dynamics. In *Artificial life II*, pages 295–312, 1992.

[11] N. Megiddo and A. Wigderson. On play by means of computing machines: preliminary version. In *Proceedings of the 1986 Conference on Theoretical aspects of reasoning about knowledge*, pages 259–274. Morgan Kaufmann Publishers Inc., 1986.

[12] J. H. Nachbar and W. R. Zame. Non-computable strategies and discounted repeated games. *Economic theory*, 8(1):103–122, 1996.

[13] M. J. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, 1994.

[14] W. H. Press and F. J. Dyson. Iterated prisoner's dilemma contains strategies that dominate any evolutionary opponent. *Proceedings of the National Academy of Sciences*, 109(26):10409–10413, 2012.

[15] A. Rubinstein. Finite automata play the repeated prisoner's dilemma. *Journal of Economic Theory*, pages 83–96, 1986.

[16] A. Rubinstein. Equilibrium in supergames. In *Essays in Game Theory*, pages 17–27. Springer, 1994.

[17] Y. Shoham and K. Leyton-Brown. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press, 2008.

[18] P. Tang, Y. Teng, Z. Wang, S. Xiao, and Y. Xu. Computational issues in time-inconsistent planning. In *Proceedings of AAAI*, 2017.

[19] J. R. Wright and K. Leyton-Brown. Beyond equilibrium: Predicting human behavior in normal-form games. In *Proceedings of AAAI*, 2010.

[20] J. R. Wright and K. Leyton-Brown. Behavioral game theoretic models: a bayesian framework for parameter analysis. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 921–930, 2012.

[21] S. Zuo and P. Tang. Optimal machine strategies to commit to in two-person repeated games. In *Proceedings of AAAI*, 2015.