Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)

Sequence–Form and Evolutionary Dynamics: Realization Equivalence to Agent Form and Logit Dynamics

Nicola Gatti and **Marcello Restelli**
Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano
Piazza Leonardo da Vinci, 32
I-20133, Milan, Italy
{nicola.gatti, marcello.restelli}@polimi.it

Abstract

Evolutionary game theory provides the principal tools to model the dynamics of multi–agent learning algorithms. While there is a long–standing literature on evolutionary game theory in strategic–form games, in the case of extensive–form games few results are known and the exponential size of the representations currently adopted makes the evolutionary analysis of such games unaffordable. In this paper, we focus on dynamics for the sequence form of extensive–form games, providing three dynamics: one realization equivalent to the normal–form logit dynamic, one realization equivalent to the agent–form replicator dynamic, and one realization equivalent to the agent–form logit dynamic. All the considered dynamics require polynomial time and space, providing an exponential compression w.r.t. the dynamics currently known and providing thus tools that can be effectively employed in practice. Moreover, we use our tools to compare the agent–form and normal–form dynamics and to provide new "hybrid" dynamics.

Introduction

Evolutionary game theory provides the most elegant tools to model the dynamics of players' strategies in strategic–interaction situations (Cressman 2003). Differently from classical game theory, it drops the assumption of *rationality* and assumes players to be *adaptive*. Nevertheless, the steady states of evolutionary dynamics constitute a subset of solutions of (non–evolutionary) game theory, e.g., in the inner of the strategy space only Nash equilibria can be steady states. Evolutionary game theory plays a prominent role in artificial intelligence, especially in *multi–agent learning* (Tuyls and Parsons 2007), where it is shown that the dynamics of any multi–agent learning algorithm can be modeled by means of evolutionary equations, providing thus a formal tool to study the expected dynamics of the algorithms and to design novel algorithms starting from known evolutionary dynamics (Tuyls, Hoen, and Vanschoenwinkel 2006; Panait, Tuyls, and Luke 2008)—see also the recent comprehensive survey (Bloembergen et al. 2015).

While there is a long–standing literature on evolutionary game theory for strategic–form games, only sporadic results are known for extensive–form games and no coherent treatment is available so far (Cressman 2003). The most of the works focus on the application of the *replicator dynamic* (one of the most famous dynamic equations) to the *normal form* of extensive–form games. However, the normal form is exponentially large in the size of the game tree, making its application affordable only to toy instances. Very few works study the application of the replicator dynamic to the *agent form* of extensive–form games (Cressman, Gaunersdorfer, and Wen 2000; Cressman 2000), showing that in games with nature the agent–form dynamics may be not *realization equivalent* to the normal–form ones (i.e., the probability distributions over the terminal nodes may be different). The agent form is more compact than the normal form, but it is exponentially large in the size of the game tree. Recently, Gatti, Panozzo, Restelli (2013) resorted to the *sequence form* to design polynomially (in the game tree size) concise dynamics. However, the constraints over the sequence–form strategies (different from those of agent and normal forms) obstacle the design of evolutionary dynamics directly in the sequence form. Nevertheless, the sequence form may be an effective tool to provide, without loss of information, an exponentially concise representation of dynamics defined in the normal form, as shown in (Gatti, Panozzo, and Restelli 2013), where the authors provide a dynamic defined on the sequence form that is realization equivalent to the normal–form replicator dynamic and that requires polynomial time and space in the game tree size. Interestingly, this result paved the way to the design of novel multi–agent learning algorithms working directly on the sequence form and whose dynamics can be modeled as variation of the replicator dynamic (Panozzo, Gatti, and Restelli 2014; Lanctot 2014).

Original contributions In this paper, we extend the state of the art on evolutionary dynamics defined on the sequence form. Specifically, we provide a sequence–form logit dynamic that is realization–equivalent to the normal–form logit dynamic (Blume 1993; Ferraioli 2013) and that requires polynomial time and space—the logit dynamic is strictly related to Quantal Response Equilibrium (Mckelvey and Palfrey 1998), that is widely used in literature to model bounded–rational players (Yang, Ordóñez, and Tambe 2012). Furthermore, we show that the agent–form

---

replicator dynamic and the agent–form logit dynamic admit, by means of the sequence form, a representation that is polynomial in the size of the game tree. The problem whether any agent–form or normal–form dynamic admit a sequence–form realization–equivalent dynamic requiring polynomial computation time and space remains open. The representation of the normal and agent forms by means of the same (sequence) form allows us to provide a direct comparison of the two dynamics, showing for the first time that, except in degenerate games, they are realization equivalent if and only if all the sequences have length equal to one. Finally, we show that the sequence form provides a tool to combine the agent–form and normal–form dynamics, giving rise to new "hybrid" dynamics unexplored so far.

Game theoretical preliminaries

Extensive–form game definition A *perfect–information* extensive–form game (Fudenberg and Tirole 1991) is a tuple $(N, A, V, T, \iota, \rho, \chi, \mathbf{u})$, where: N is the set of players ($i \in N$ denotes a generic player), A is the set of actions ($A_i \subseteq A$ denotes the set of actions of player i and $a \in A$ denotes a generic action), V is the set of decision nodes ($V_i \subseteq V$ denotes the set of decision nodes of i), T is the set of terminal nodes ($w \in V \cup T$ denotes a generic node and w_0 is root node), $\iota : V \to N$ returns the player that acts at a given decision node, $\rho : V \to \wp(A)$ returns the actions available to player $\iota(w)$ at w, $\chi : V \times A \to V \cup T$ assigns the next (decision or terminal) node to each pair $\langle w, a \rangle$ where a is available at w, and $\mathbf{u} = (u_1, \ldots, u_{|N|})$ is the set of players' utility functions $u_i : T \to \mathbb{R}$. Games with *imperfect information* extend those with perfect information, allowing one to capture situations in which some players cannot observe some actions undertaken by other players. We denote by $V_{i,h}$ the h–th *information set* of player i. An information set is a set of decision nodes such that when a player plays at one of such nodes she cannot distinguish the node in which she is playing. For the sake of simplicity, we assume that every information set has a different index h, thus we can univocally identify an information set by h. Furthermore, since the available actions at all nodes w belonging to the same information set h are the same, with abuse of notation, we write $\rho(h)$ in place of $\rho(w)$ with $w \in V_{i,h}$. An imperfect–information game is a tuple $(N, A, V, T, \iota, \rho, \chi, \mathbf{u}, H)$ where $(N, A, V, T, \iota, \rho, \chi, \mathbf{u})$ is a perfect–information game and $H = (H_1, \ldots, H_{|N|})$ induces a partition $V_i = \bigcup_{h \in H_i} V_{i,h}$ such that for all $w, w' \in V_{i,h}$ we have $\rho(w) = \rho(w')$. We focus on games with *perfect recall* where each player recalls all the own previous actions and the ones of the opponents (Fudenberg and Tirole 1991).

(Reduced) Normal form (von Neumann and Morgenstern 1944) It is a tabular representation in which each normal–form action, called *plan* and denoted by $p \in P_i$ where P_i is the set of plans of player i, specifies one action $a \in A_i$ per information set of player i. We denote by $\boldsymbol{\pi}_i$ a normal–form strategy of player i and by $\pi_i(p)$ the probability associated with plan p. The number of plans (and therefore the size of the normal form) is exponential in the size of the game tree. The *reduced normal form* is obtained from the normal form by deleting replicated strategies (Vermeulen and Jansen 1998). Although reduced normal form can be much smaller than normal form, it is still exponential in the size of the game tree. Hereafter, we shall only consider the plans in the reduced normal form.

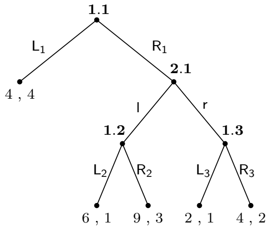

Figure 1: Example of two–player perfect–information extensive–form game, $\mathbf{x}.\mathbf{y}$ denote the $\mathbf{y}$–th node of player $\mathbf{x}$.

Example 1. *The reduced normal form of the game in Figure 1 and a pair of normal–form strategies are:*

player 2

player 1		l	r
	L_1*	$4, 4$	$4, 4$
	$R_1L_2L_3$	$6, 1$	$2, 1$
	$R_1L_2R_3$	$6, 1$	$4, 2$
	$R_1R_2L_3$	$9, 3$	$2, 1$
	$R_1R_2R_3$	$9, 3$	$4, 2$

$$\pi_1(p) = \begin{cases} \frac{1}{3} & L_1* \\ 0 & R_1L_2L_3 \\ \frac{1}{3} & R_1L_2R_3 \\ 0 & R_1R_2L_3 \\ \frac{1}{3} & R_1R_2R_3 \end{cases} \qquad \pi_2(p) = \begin{cases} 1 & l \\ 0 & r \end{cases}$$

Agent form (Kuhn 1950; Selten 1975) It is a tabular representation in which each player i is replicated in a number of fictitious players, called agents and denoted by $i.h$, each per information set, and all the agents of the same player have the same utility U_i. An agent–form strategy is commonly said *behavioral* and is denoted by $\boldsymbol{\sigma}_i$. We denote the strategy of agent $i.h$ by $\boldsymbol{\sigma}_{i.h}$ and by $\sigma_{i.h}(a)$ the probability associated with action $a \in A_i$ where h is such that $a \in \rho(h)$. In the agent form, each agent has a number of strategies that is linear in the size of the game, but the size of the tabular representation is exponential in the number of information sets of the game tree.

Example 2. *The agent form of the game in Figure 1 and a pair of behavioral strategies are:*

agent 2.1

agent 1.1		l	r
	L_1	$4, 4, 4, 4$	$4, 4, 4, 6$
	R_1	$6, 1, 6, 6$	$2, 1, 2, 2$

L_2

agent 2.1

agent 1.1		l	r	
	L_1	$4, 4, 4, 4$	$4, 4, 4, 4$	L_3
	R_1	$9, 3, 9, 9$	$4, 2, 4, 4$	

R_2

agent 1.2

agent 2.1

agent 1.1		l	r
	L_1	$4, 4, 4, 4$	$4, 4, 4, 6$
	R_1	$6, 1, 6, 6$	$2, 1, 2, 2$

L_2

agent 2.1

agent 1.1		l	r	
	L_1	$4, 4, 4, 4$	$4, 4, 4, 4$	R_3
	R_1	$9, 3, 9, 9$	$4, 2, 4, 4$	

R_2

agent 1.2

$$\sigma_{1.1}(\cdot) = \begin{cases} 1 & L_1 \\ 0 & R_1 \end{cases} \sigma_{1.2}(\cdot) = \begin{cases} 0 & L_2 \\ 1 & R_2 \end{cases} \sigma_{1.3}(\cdot) = \begin{cases} 0 & L_3 \\ 1 & R_3 \end{cases} \sigma_{2.1}(\cdot) = \begin{cases} \frac{8}{10} & l \\ \frac{2}{10} & r \end{cases}$$

Sequence form (von Stengel 1996) It is a tabular representation that presents additional constraints. Sequence–form actions are called *sequences*. A sequence $q \in Q_i$ of player i is a set of $|q|$ consecutive actions $a \in A_i$ where $Q_i \subseteq Q$ is the set of sequences of player i and Q is the set of all the sequences. A sequence can be *terminal*, if, combined with some sequence of the opponents, it leads to a terminal node, or *non–terminal* otherwise. The initial sequence of every player, denoted by $\mathsf{q}_\emptyset$, is said *empty sequence*. We denote by $q(a)$ the sequence whose last action is a, by $a(q)$ the last action of q, and by $a \in q$ the fact that a is contained in q. Furthermore, we denote by $h(a)$ the information set in which a can be played, and similarly $h(q)$ the information set in which the last action of q can be played. We denote by $q|a$ the *extended* sequence obtained by appending a to sequence q, by $q \setminus a$ the sequence obtained by removing the last action a from q, and by $q \rightarrow h$ the fact that q leads to information set h (i.e., there are $a \in A, q' \in Q$ such that $q' = q|a$ and $h(q') = h$). Finally, we denote by $\mathbf{x}_i$ the sequence–form strategy of player i and by $x_i(q)$ the probability associated with sequence $q \in Q_i$. Well–defined $\mathbf{x}_i$ are such that, for every information set $h \in H_i$, the probability $x_i(q)$ assigned to the sequence q such that $h(q) = h'$ is equal to the sum of the probabilities $x_i(q')$s where $q' = q|a$ for all the $a \in h'$. Sequence–form constraints are $x_i(\mathsf{q}_\emptyset) = 1$ and $x_i(q) = \sum_{a \in \rho(h(q))} x_i(q|a)$ for every sequence q and for every player i. The player i's utility is represented as a sparse multi–dimensional array, denoted, with an abuse of notation, by U_i, specifying the value associated with every combination of terminal sequences of all the players. The size of the sequence form is linear in the game tree size.

Example 3. *The sequence form of the game in Figure 1 and a pair of sequence–form strategies are:*

		player 2		
		$\mathsf{q}_\emptyset$	l	r
player 1	$\mathsf{q}_\emptyset$			
	L_1	4, 4		
	R_1			
	$\mathsf{R}_1\mathsf{L}_2$		6, 1	
	$\mathsf{R}_1\mathsf{R}_2$		9, 3	
	$\mathsf{R}_1\mathsf{L}_3$			2, 1
	$\mathsf{R}_1\mathsf{R}_3$			4, 2

$$x_1(q) = \begin{cases} 1 & \mathsf{q}_\emptyset \\ \frac{1}{3} & \mathsf{q}_{\mathsf{L}_1} \\ \frac{2}{3} & \mathsf{q}_{\mathsf{R}_1} \\ \frac{1}{3} & \mathsf{q}_{\mathsf{R}_1\mathsf{L}_2} \\ \frac{1}{3} & \mathsf{q}_{\mathsf{R}_1\mathsf{R}_2} \\ 0 & \mathsf{q}_{\mathsf{R}_1\mathsf{L}_3} \\ \frac{2}{3} & \mathsf{q}_{\mathsf{R}_1\mathsf{R}_3} \end{cases} \quad x_2(q) = \begin{cases} 1 & \mathsf{q}_\emptyset \\ 1 & \mathsf{q}_\mathsf{l} \\ 0 & \mathsf{q}_\mathsf{r} \end{cases}$$

Realization equivalence The relation of realization equivalence, requiring two strategies to induce the same probability distribution over the terminal nodes, can be written, for sequence form strategies, as follows:

Definition 4. *Strategies* $\mathbf{x}_i$ *and* $\boldsymbol{\pi}_i$ *are realization equivalent if and if,* $\forall q \in Q_i$, $x_i(q) = \sum_{p \in P_i : a(q) \in p} \pi_i(p)$.

Definition 5. *Strategies* $\mathbf{x}_i$ *and* $\boldsymbol{\sigma}_i$ *are realization equivalent if and if,* $\forall q \in Q_i$, $x_i(q) = \prod_{a \in q} \sigma_{i.h(a)}(a)$.

Evolutionary dynamics For the sake of brevity, we report here only evolutionary dynamics with continuous time and two players for player 1, for details see (Sandholm 2010). The continuous–time normal–form replicator and logit dynamic equations with two players are, respectively:

$$\dot{\pi}_1(p,t) = \pi_1(p,t)[(\mathbf{e}_p - \boldsymbol{\pi}_1(t))U_1\boldsymbol{\pi}_2(t)], \tag{1}$$

$$\dot{\pi}_1(p,t) = \frac{\exp\left[\frac{\mathbf{e}_p U_1 \boldsymbol{\pi}_2(t)}{\eta}\right]}{\sum_{p' \in P_1} \exp\left[\frac{\mathbf{e}_{p'} U_1 \boldsymbol{\pi}_2(t)}{\eta}\right]} - \pi_1(p,t), \tag{2}$$

where $\mathbf{e}_i$ is the vector in which the component corresponding to the i–th plan/action is "1" and the others are "0". The continuous–time agent–form replicator and logit dynamics equations with two players are, respectively:

$$\dot{\sigma}_{1.h}(a,t) = \sigma_{1.h}(a,t) \cdot \left[(\mathbf{e}_a - \boldsymbol{\sigma}_{1.h}(t))U_1 \prod_{h' \in H_1 : h' \neq h} \boldsymbol{\sigma}_{1.h'}(t) \prod_{h' \in H_2} \boldsymbol{\sigma}_{2.h'}(t)\right], \tag{3}$$

$$\dot{\sigma}_{1.h}(a,t) = -\sigma_{1.h}(a,t) + \frac{\exp\left[\frac{\mathbf{e}_a U_1 \prod_{h' \in H_1 : h' \neq h} \boldsymbol{\sigma}_{1.h'}(t) \prod_{h' \in H_2} \boldsymbol{\sigma}_{2.h'}(t)}{\eta}\right]}{\sum_{a' \in \rho(h)} \exp\left[\frac{\mathbf{e}_{a'} U_1 \prod_{h' \in H_1 : h' \neq h} \boldsymbol{\sigma}_{1.h'}(t) \prod_{h' \in H_2} \boldsymbol{\sigma}_{2.h'}(t)}{\eta}\right]}. \tag{4}$$

where $a \in \rho(h)$ and $\eta \in (0, +\infty)$ is the exploration parameter.

Normal–form equivalent logit dynamic

We initially provide the sequence–form continuous–time evolutionary dynamic for player 1, the equation for player 2 is analogous:

$$\dot{x}_1(q,t) = \frac{\sum_{p \in P_1 : a(q) \in p} \exp\left[\frac{\sum_{q' \in Q_1 : a(q') \in p} \mathbf{e}_{q'} U_1 \mathbf{x}_2(t)}{\eta}\right]}{\sum_{p' \in P_1} \exp\left[\frac{\sum_{q' \in Q_1 : a(q') \in p'} \mathbf{e}_{q'} U_1 \mathbf{x}_2(t)}{\eta}\right]} - x_1(q,t). \tag{5}$$

Theorem 6. *Continuous–time evolutionary dynamic (5) is realization equivalent to the continuous–time normal–form logit dynamic (2).*

Proof. Define $\mathbf{e}_q$ as a vector with 1 in the position related to sequence q and 0 elsewhere. Note that $\mathbf{e}_q$ is not, generally, a well–defined sequence–form strategy. The proof follows from the following calculations for every $q \in Q_1$:

$$\dot{x}_1(q,t) = \frac{d}{dt}\left(\sum_{p \in P_1 : a(q) \in p} \pi_1(p,t)\right) \tag{6}$$

$$= \sum_{p \in P_1 : a(q) \in p} \dot{\pi}_1(p,t) \tag{7}$$

$$= \frac{\sum_{p \in P_1 : a(q) \in p} \exp\left[\frac{\mathbf{e}_p U_1 \boldsymbol{\pi}_2(t)}{\eta}\right]}{\sum_{p' \in P_1} \exp\left[\frac{\mathbf{e}_{p'} U_1 \boldsymbol{\pi}_2(t)}{\eta}\right]} - \underbrace{\sum_{p \in P_1 : a(q) \in p} \pi_1(p,t)}_{x_1(q,t)} \tag{8}$$

$$= \frac{\sum_{p \in P_1 : a(q) \in p} \exp\left[\frac{\sum_{q' \in Q_1 : a(q') \in p} \mathbf{e}_{q'} U_1 \mathbf{x}_2(t)}{\eta}\right]}{\sum_{p' \in P_1} \exp\left[\frac{\sum_{q' \in Q_1 : a(q') \in p'} \mathbf{e}_{q'} U_1 \mathbf{x}_2(t)}{\eta}\right]} - x_1(q,t) \tag{9}$$

where in (6) we apply the derivative w.r.t. time to the condition stating that $\mathbf{x}_1(t)$ and $\boldsymbol{\pi}_i(t)$ are realization equivalent (see Definition 4), in (7) we move the derivative operator inside the sum operator, in (8) we apply the definition of normal–form logit dynamic to $\pi_i(p.t)$ forcing, by the condition of realization equivalence applied in (6), $x_1(q,t)$ to evolve as the corresponding normal–form strategy, in (9) we expand the term $\mathbf{e}_p U_1 \boldsymbol{\pi}_2(t)$ as $\sum\limits_{q' \in Q : a(q') \in p} \mathbf{e}_{q'} U_1 \mathbf{x}_2(t)$ making the evolution to depend on sequence–form strategies (notice that U_1 in $\mathbf{e}_p U_1 \boldsymbol{\pi}_2(t)$ and in $\mathbf{e}_q U_1 \mathbf{x}_2(t)$ are different, being defined in two different strategy spaces). This expansion is possible as follows. Denote by $\bar{\mathbf{x}}_1$ the pure sequence–form strategy realization–equivalent to plan p. $\bar{\mathbf{x}}_1$ is such that $\bar{x}_1(q) = 1$ if $q \in p$ and 0 in the other positions. Then, we have $\mathbf{e}_p U_1 \boldsymbol{\pi}_2(t) = \bar{\mathbf{x}}_1 U_1 \mathbf{x}_2(t) = \left(\sum\limits_{q \in Q_1 : a(q) \in p} \mathbf{e}_q \right) U_1 \mathbf{x}_2(t) = \sum\limits_{q \in Q_1 : a(q) \in p} \mathbf{e}_q U_1 \mathbf{x}_2(t)$. This completes the proof. □

Notice that evolutionary dynamic (5) allows one to work only with sequence–form strategies in place of normal–form strategies, using thus a number of equations that is linear in the size of the game tree and overcoming the numerical–stability issues the normal–form strategies suffer of (i.e., the number of plans rising exponentially, the probability of each plan decreases exponentially). However, although the above dynamic provides an exponential compression of the normal–form dynamic, it requires exponential time due to the sum over all the plans $p \in P_1$. However, we provide a recursive polynomial–time procedure that, given a sequence $\bar{q}$, returns the value of the term $\sum\limits_{p \in P_1 : a(\bar{q}) \in p} \exp\left[\frac{\sum\limits_{q' \in Q_1 : a(q') \in p} \mathbf{e}_{q'} U_1 \mathbf{x}_2(t)}{\eta} \right]$. This is possible because such a term can be factorized in a polynomial number of sub terms.

The recursive procedure is reported in Algorithm 1 and exploits the formulas, requiring a polynomial number of operations, reported in the following box.

$$\sum_{p \in P_1 : a(\bar{q}) \in p} \exp\left[\frac{\sum\limits_{q \in Q_1 : a(q) \in p} \mathbf{e}_q U_1 \mathbf{x}_2(t)}{\eta} \right] = \mathbf{E}_{\mathsf{q}_\emptyset}(\bar{q})$$

$$\Lambda_{q'}(q) = \begin{cases} 1 & \text{if there is } \mathbf{x}'_1 \text{ s.t. } x'_1(q) = 1 \text{ and } x'_1(q') = 1 \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{E}_{q'}(q) = \Lambda_{q'}(q) \exp\left[\frac{\mathbf{e}_{q'} U_1 \mathbf{x}_2(t)}{\eta} \right] \prod_{h \in H_1 : q' \to h} \mathbf{E}_h(q)$$

$$\mathbf{E}_h(q) = \sum_{q' | a \in Q_1 : a \in \rho(h)} \mathbf{E}_{q' | a}(q)$$

Algorithm 1 performs a depth–first search, visiting each node and edge of the game tree once. Algorithm 1 must be executed $|Q_1|$ times, one for each sequence and thus its complexity is linear in the size of the game tree.

Algorithm 1 RECURSIVE–EXP

```
procedure RECURSIVE–EXP(z, q)
    if $z \in Q$ then
        $H_z \leftarrow \{h \in H$ s.t. $\iota(h) = 1$ and $z \to h\}$
        for all $h' \in H_z$ do
            $\mathbf{E}_{h'}(q) \leftarrow$ RECURSIVE–EXP($h', q$)
        return $\mathbf{E}_z(q)$ as defined in the box
    else  ($z \in H$)
        $Q_z \leftarrow \{q | a \in Q$ s.t. $a \in \rho(h)\}$
        for all $q' \in Q_z$ do
            $\mathbf{E}_{q'}(q) \leftarrow$ RECURSIVE–EXP($q', q$)
        return $\mathbf{E}_z(q)$ as defined in the box
```

The term $\Lambda_{q'}(q)$ is equal to 1 when $a(q)$ and $a(q')$ can coexist in the same plan p and 0 otherwise. It is used to discard all the contributions due to sequences that are mutually exclusive to $\bar{q}$. $\mathbf{E}_{q'}(q)$ is the contribution due to sequence q' when we are computing $\mathbf{E}_{\mathsf{q}_\emptyset}(q)$ and it is defined as the multiplication of all the contributions $\mathbf{E}_h(q)$ due to the information set h directly achievable from q' and the term due to the utility immediately achievable by playing q, while $\mathbf{E}_h(q)$ is given by the sum of all the contributions due to the actions played at h when we are computing $\mathbf{E}_{\mathsf{q}_\emptyset}(q)$. Below, we report a sketch of the application of the algorithm.

Example 7. *By the application of Algorithm 1 to compute* $\mathbf{E}_{\mathsf{q}_\emptyset}(\mathsf{q}_\emptyset)$ *for player 1 in the game of Figure 1, we obtain (for the sake of presentation, we omit time t):*

$$\exp\left[\frac{\mathbf{e}_{\mathsf{q}_{\mathsf{L}_1}} U_1 \mathbf{x}_2}{\eta}\right] + \left(\exp\left[\frac{\mathbf{e}_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}} U_1 \mathbf{x}_2}{\eta}\right] + \exp\left[\frac{\mathbf{e}_{\mathsf{q}_{\mathsf{R}_1\mathsf{R}_2}} U_1 \mathbf{x}_2}{\eta}\right] \right) \left(\exp\left[\frac{\mathbf{e}_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_3}} U_1 \mathbf{x}_2}{\eta}\right] + \exp\left[\frac{\mathbf{e}_{\mathsf{q}_{\mathsf{R}_1\mathsf{R}_3}} U_1 \mathbf{x}_2}{\eta}\right] \right). \quad (10)$$

Computing the corresponding value by using the normal–form strategies, we obtain:

$$\exp\left[\frac{\mathbf{e}_{\mathsf{L}*} U_1 \boldsymbol{\pi}_2}{\eta}\right] + \exp\left[\frac{\mathbf{e}_{\mathsf{R}_1\mathsf{L}_2\mathsf{L}_3} U_1 \boldsymbol{\pi}_2}{\eta}\right] + \exp\left[\frac{\mathbf{e}_{\mathsf{R}_1\mathsf{L}_2\mathsf{R}_3} U_1 \boldsymbol{\pi}_2}{\eta}\right] + \exp\left[\frac{\mathbf{e}_{\mathsf{R}_1\mathsf{R}_2\mathsf{L}_3} U_1 \boldsymbol{\pi}_2}{\eta}\right] + \exp\left[\frac{\mathbf{e}_{\mathsf{R}_1\mathsf{R}_2\mathsf{R}_3} U_1 \boldsymbol{\pi}_2}{\eta}\right].$$

By replacing:

$$\exp\left[\frac{\mathbf{e}_{\mathsf{R}_1\mathsf{L}_2\mathsf{L}_3} U_1 \boldsymbol{\pi}_2}{\eta}\right] = \exp\left[\frac{(\mathbf{e}_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}} + \mathbf{e}_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_3}}) U_1 \mathbf{x}_2}{\eta}\right]$$

and doing the same with the other terms based on normal–form strategies and subsequently by factorizing the terms, we obtain (10).

Hence, we can state the following result.

Corollary 8. *The compute time and space of the continuous–time normal–form logit dynamic can be exponentially compressed without loss of information by means of the sequence form.*

The above results rise the following question:

Question 9. *Does any (continuous–time and/or discrete–time) normal–form evolutionary dynamic admit a realization–equivalent sequence–form dynamic requiring compute time and space that are polynomial in the size of the game tree?*

We leave the question open here. Our conjecture is that the answer to the question is negative and is based on the existence of some dynamics (e.g., Smith and Brown–von Neumann–Nash) using highly non–linear operators (i.e., absolute value) that seem not admitting any exponential compression from the normal form to the sequence form.

Agent–form equivalent dynamics

Replicator dynamic

We initially provide the sequence–form continuous–time evolutionary dynamic for player 1, the equation for player 2 is analogous:

$$\dot{x}_1(q,t) = x_1(q,t)\,|q| \left[\Big(\mathbf{d}_q(\mathbf{x}_1(t)) - \mathbf{x}_1(t) \Big) U_1 \mathbf{x}_2(t) \right], \qquad (11)$$

where:

$$\mathbf{d}_q(\mathbf{x}_1(t)) = \frac{\sum\limits_{a \in q} \mathbf{r}_a(\mathbf{x}_1(t))}{|q|},$$

$$r_a(q,\mathbf{x}_1(t)) = \begin{cases} x_1(q,t) & \forall a' \in q, a' \notin \rho(h(a)) \\ 0 & \exists a' \in q, a' \neq a, a' \in \rho(h(a)) \\ \frac{x_1(q,t) x_1(q',t)}{x_1(q'|a,t)} & \exists q', q'|a \subseteq q \end{cases}.$$

Vector $\mathbf{r}_a$ is $|Q_1|$–dimensional and has the following property. Given a behavioral strategy $\boldsymbol{\sigma}_1(t)$ and its realization–equivalent sequence–form strategy $\mathbf{x}_1(t)$, $\mathbf{r}_a(\mathbf{x}_1(t))$ is the sequence–form strategy realization–equivalent to the behavioral strategy $\boldsymbol{\sigma}'_1(t)$ obtained by setting $\boldsymbol{\sigma}'_1(t) = \boldsymbol{\sigma}_1(t)$ except for the information set h in which $a \in \rho(h)$ where $\sigma_{1.h(a)}(a)$ is set equal to 1 and $\sigma_{1.h(a')}(a')$ is set equal to 0 for all the $a' \neq a$ with $a' \in \rho(h)$. Formally, we have $\mathbf{r}_a(\mathbf{x}_1(t)) = \mathbf{e}_a \prod_{h' \in H_1 : h' \neq h(a)} \boldsymbol{\sigma}_{1.h'}(t)$. The definition of vector $\mathbf{r}_a$ is: $r_a(q,\mathbf{x}_1(t))$ is equal to $x_1(q,t)$ for all the sequences q that do not include actions played at $h(a)$, it is equal to 0 for all the sequences that include an action a' played at the same information set of a such that $a' \neq a$, and finally it is equal to $x_1(q,t)$ scaled by $\frac{x_1(q',t)}{x_1(q'|a,t)}$ with q' such that $q'|a \subseteq q$ forcing thus that at $h(a)$ action a is the only played action.

Vector $\mathbf{d}_q$ is $|Q_1|$–dimensional and is the average over all the $\mathbf{r}_a$ such that $a \in q$. Finally, it can be easily observed that vector $\mathbf{d}_q$ can be computed in polynomial time in the size of the game tree.

We can state the following theoretical result.

Theorem 10. *Continuous–time evolutionary dynamic (11) is realization equivalent to the continuous–time agent–form replicator dynamic (3).*

Proof. The proof is given by the following calculations, in which we start from the condition of realization equivalence between $\mathbf{x}_1(t)$ and $\boldsymbol{\sigma}_1(t)$ (see Definition 5), we apply the derivative w.r.t. time to such a condition, and we force $\sigma_1(t)$ to evolve as prescribed by the agent–form replicator dynamic. Finally, we derive Equation (11).

$$\begin{aligned}
\dot{x}_1(q,t) =& \frac{d}{dt} \left(\prod_{a \in q} \sigma_{1.h(a)}(a,t) \right) \\
=& \sum_{a \in q} \left(\dot{\sigma}_{1.h(a)}(a,t) \prod_{a' \in q : a' \neq a} \sigma_{1.h(a')}(a',t) \right) \\
=& \left(\prod_{a' \in q} \sigma_{1.h(a')}(a',t) \right) \left(\sum_{a \in q} \frac{\dot{\sigma}_{1.h(a)}(a,t)}{\sigma_{1.h(a)}(a,t)} \right) \\
=& x_1(q,t) \left(\sum_{a \in q} \frac{\dot{\sigma}_{1.h(a)}(a,t)}{\sigma_{1.h(a)}(a,t)} \right) \\
=& x_1(q,t) \Bigg(\sum_{a \in q} \frac{\sigma_{1.h(a)}(a,t)(\mathbf{e}_a - \boldsymbol{\sigma}_{1.h(a)}(t)) U_1 \cdot}{\sigma_{1.h(a)}(a,t)} \\
& \frac{\cdot \prod\limits_{h' \in H_1 : h' \neq h(a)} \boldsymbol{\sigma}_{1.h'}(t) \prod\limits_{h' \in H_2} \boldsymbol{\sigma}_{2.h'}(t)}{\sigma_{1.h(a)}(a,t)} \Bigg) \\
=& x_1(q,t) \Bigg(\sum_{a \in q} \Big(\mathbf{e}_a - \boldsymbol{\sigma}_{1.h(a)}(t) \Big) U_1 \cdot \\
& \cdot \prod_{h' \in H_1 : h' \neq h(a)} \boldsymbol{\sigma}_{1.h'}(t) \prod_{h' \in H_2} \boldsymbol{\sigma}_{2.h'}(t) \Bigg) \\
=& x_1(q,t) \left(\sum_{a \in q} \Big(\mathbf{r}_a(\mathbf{x}_1(t)) - \mathbf{x}_1(t) \Big) U_1 \mathbf{x}_2(t) \right) \\
=& x_1(q,t)\,|q| \left(\left(\frac{\sum\limits_{a \in q} \mathbf{r}_a(\mathbf{x}_1(t))}{|q|} - \mathbf{x}_1(t) \right) U_1 \mathbf{x}_2(t) \right) \\
=& x_1(q,t)\,|q| \left(\Big(\mathbf{d}_q(\mathbf{x}_1(t)) - \mathbf{x}_1(t) \Big) U_1 \mathbf{x}_2(t) \right).
\end{aligned}$$

This completes the proof. □

Therefore, we can state the following.

Corollary 11. *The compute time and space of the continuous–time agent–form replicator dynamic can be exponentially compressed without loss of information by means of the sequence form.*

A direct comparison of Equation (11) w.r.t. the equation of sequence–form replicator dynamic realization equivalent to the normal–form one can be provided, studying thus the differences between the agent–form and the normal–form replicator dynamics. We recall that the sequence–form replicator dynamics realization–equivalent to the normal–form one is (Gatti, Panozzo, and Restelli 2013):

$$\dot{x}_1(q,t) = x_1(q,t) \left(\Big(\mathbf{g}_q(\mathbf{x}_1(t)) - \mathbf{x}_1(t) \Big) U_1 \mathbf{x}_2(t) \right),$$

where component q' of vector $\mathbf{g}_q(\mathbf{x}_1(t))$ is 1 when $q' \subseteq q$, is 0 when there are $a \in q$ and $a' \in q'$ so that $a \neq a'$ and $h(a) = h(a')$, and is $\frac{x_i(q',t)}{x_i(q \cap q',t)}$ otherwise. The differences between the agent–form and the normal–form replicator dynamics are that in the agent–form replicator dynamic:

- there is a gain of $|q| \geq 1$, and therefore the longer the sequence the higher the gain making the dynamics of short sequences slower than the dynamics of long sequences;

- vector $\mathbf{g}_q$ is replaced by vector $\mathbf{d}_q = \frac{\sum_{a \in q} \mathbf{r}_a(\mathbf{x}_1(t))}{|q|}$ that is an average of the contributions $\mathbf{r}_a$ given by all the $a \in q$.

We provide an example, showing the differences between $\mathbf{d}_q$ and $\mathbf{g}_q$ in a specific game instance.

Example 12. *Given $\mathbf{x}_1$ introduced in Example 3, we report the values of $\mathbf{d}_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathbf{x}_1)$ and $\mathbf{g}_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathbf{x}_1)$:*

$$\mathbf{d}_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathbf{x}_1) = \qquad\qquad \mathbf{g}_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathbf{x}_1) =$$

$$\begin{cases} d_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathsf{q}_\emptyset, \mathbf{x}_1) & = 1 \\ d_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathsf{q}_{\mathsf{L}_1}, \mathbf{x}_1) & = \frac{1}{6} \\ d_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathsf{q}_{\mathsf{R}_1}, \mathbf{x}_1) & = \frac{5}{6} \\ d_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}, \mathbf{x}_1) & = \frac{7}{12} \\ d_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathsf{q}_{\mathsf{R}_1\mathsf{R}_2}, \mathbf{x}_1) & = \frac{1}{4} \\ d_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathsf{q}_{\mathsf{R}_1\mathsf{L}_3}, \mathbf{x}_1) & = 0 \\ d_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathsf{q}_{\mathsf{R}_1\mathsf{R}_3}, \mathbf{x}_1) & = \frac{5}{6} \end{cases} \qquad \begin{cases} g_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathsf{q}_\emptyset, \mathbf{x}_1) & = 1 \\ g_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathsf{q}_{\mathsf{L}_1}, \mathbf{x}_1) & = 0 \\ g_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathsf{q}_{\mathsf{R}_1}, \mathbf{x}_1) & = 1 \\ g_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}, \mathbf{x}_1) & = 1 \\ g_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathsf{q}_{\mathsf{R}_1\mathsf{R}_2}, \mathbf{x}_1) & = 0 \\ g_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathsf{q}_{\mathsf{R}_1\mathsf{L}_3}, \mathbf{x}_1) & = 0 \\ g_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathsf{q}_{\mathsf{R}_1\mathsf{R}_3}, \mathbf{x}_1) & = 1 \end{cases} .$$

It can be observed that, while $\mathbf{g}_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}$ is not affected by $x_1(\mathsf{q}_{\mathsf{L}_1}, t)$ and $x_1(\mathsf{q}_{\mathsf{R}_1\mathsf{R}_2}, t)$ that are sequences mutually exclusive w.r.t. $\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}$, $\mathbf{d}_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}$ is. As a consequence, the two dynamics are different:

$$\dot{x}_1(\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}, t) = x_1(\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}, t) \underbrace{|\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}|}_{2} \underbrace{\left[\Big(\mathbf{d}_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathbf{x}_1(t)) - \mathbf{x}_1(t)\Big) U_1 \mathbf{x}_2(t)\right]}_{\frac{1}{12}}$$

$$\dot{x}_1(\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}, t) = x_1(\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}, t) \qquad \underbrace{\left[\Big(\mathbf{g}_{\mathsf{q}_{\mathsf{R}_1\mathsf{L}_2}}(\mathbf{x}_1(t)) - \mathbf{x}_1(t)\Big) U_1 \mathbf{x}_2(t)\right]}_{-\frac{1}{3}} .$$

We can show that agent–form and sequence–form replicator dynamics are equivalent only in the trivial case in which each agent has only sequences of length one.

Theorem 13. *Except in degenerate games, agent–form replicator dynamics and normal–form replicator dynamics are realization equivalent only when each agent has only sequences of length one.*

Proof sketch. The proof easily follows from the derivation of the formulas of the two sequence-form replicator dynamics (the one realization equivalent to the normal–form replicator dynamic and the one realization equivalent to the agent–form replicator dynamic) for the game depicted in Figure 2. The two dynamics are the same if and only if parameters B and C have the same value, but, if this holds, the game is degenerate. □

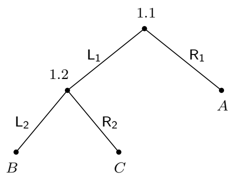

Figure 2: Game tree used in the proof of Theorem 13.

The sequence form, providing a unique representation for both the normal–form and agent–form replicator dynamics, can be used to define new *hybrid* replicator dynamics. More precisely, we can state the following result, whose proof is based on the linearity of the equation.

Theorem 14. *Given any $\mu \in [0, 1]$, the replicator dynamics:*

$$\dot{x}_1(q, t) = x_1(q, t) \Bigg[\bigg(\mu\, |q| \Big(\mathbf{d}_q(\mathbf{x}_1(t)) - \mathbf{x}_1(t)\Big) - (1 - \mu)\Big(\mathbf{g}_q(\mathbf{x}_1(t)) - \mathbf{x}_1(t)\Big)\bigg) U_1 \mathbf{x}_2(t)\Bigg], \tag{12}$$

obtained by the convex combination of the agent–form realization–equivalent and the normal–form realization–equivalent replicator dynamics, is a well–defined dynamic equation in the space of the sequence–form strategies.

Logit dynamic

We initially provide the sequence–form continuous–time evolutionary dynamic for agent 1, the equation for agent 2 is analogous:

$$\dot{x}_1(q, t) = x_1(q, t) \left(\sum_{a \in q} \left(\frac{x_1(q(a) \setminus a, t) \exp\left[\frac{\mathbf{r}_a(\mathbf{x}_1(t)) U_1 \mathbf{x}_2(t)}{\eta}\right]}{x_1(q(a), t) \sum_{b \in \rho(h(a))} \exp\left[\frac{\mathbf{r}_b(\mathbf{x}_1(t)) U_1 \mathbf{x}_2(t)}{\eta}\right]} \right) - |q| \right) . \tag{13}$$

We can state the following result, whose proof is similar to that one of Theorem 10.

Theorem 15. *Continuous–time evolutionary dynamic (13) is realization equivalent to the continuous–time agent–form logit dynamic (4).*

From the above result and the fact that the right term of Equation (13) can be computed in polynomial time in the size of the game tree, we can state the following result.

Corollary 16. *The compute time and space of the continuous–time agent–form logit dynamic can be exponentially compressed without loss of information by means of the sequence form.*

The comparison of the two logit dynamics is not immediate as it is instead the comparison of the two replicator dynamics, since their corresponding sequence–form dynamics are extremely different mainly due to the recursive definition of the normal–form realization–equivalent dynamic. However, we provide an example showing that the evolutions in the two dynamics are different also in the logit case.

We can show that agent–form and normal–form logit dynamics are equivalent only in the trivial case in which each agent has only sequences of length one.

Theorem 17. *Except in degenerate games, agent–form logit dynamic and normal–form logit dynamic are realization equivalent only when each agent has only sequences of length one.*

Proof. We provide a counterexample in which the evolutions are different. The counterexample is the same used in the proof of Theorem 13. Additionally, we set $\eta = 1$. The normal–form realization–equivalent logit dynamic prescribes $\dot{x}_1(\mathsf{q}_{R_1L_2}, t) \simeq -0.29$, while the agent–form realization–equivalent dynamic prescribes $\dot{x}_1(\mathsf{q}_{R_1L_2}, t) \simeq -0.10$. $\square$

Also in the case of logit dynamics, the hybrid dynamic equation obtained by the convex combination of the two dynamics is well–defined in the space of the sequence–form strategies.

Theorem 18. *The logit dynamics obtained by the convex combination of the agent–form realization–equivalent and the normal–form realization–equivalent logit dynamics, (13) and (5) respectively, is a well–defined dynamic equation in the space of the sequence–form strategies.*

The above results rise the following question:

Question 19. *Does any (continuous–time and/or discrete–time) agent–form evolutionary dynamic admit a realization–equivalent sequence–form dynamic requiring compute time and space that are polynomial in the size of the game tree?*

As in the normal–form case, we leave the question open here. We have no conjecture for this case, even if our preliminary analysis suggests that the answer is positive, showing that all the known agent–form dynamics admit a realization–equivalent sequence–form dynamics whose computational cost is polynomial in the size of the game tree.

Conclusions and future work

In this paper, we provide three sequence–form dynamics for extensive–form games: one realization equivalent to the normal–form logit dynamic, one realization equivalent to the agent–form replicator dynamic, and one realization equivalent to the agent–form logit dynamic. All our dynamics require polynomial compute time and space, providing an exponential compression w.r.t. the dynamics currently available in the literature and providing thus effective tools that can be employed for the evolutionary analysis of extensive–form games. Such an analysis is crucial for multi–agent learning algorithms, where the dynamics of the algorithms can be described by means of evolutionary game theory models. Furthermore, we use our tools to compare the agent–form and normal–form dynamics and to provide new "hybrid" dynamics. In particular, we show that agent–form and normal–form replicator are equivalent if and only if all the sequences are not longer than 1.

Some questions remain open, such as whether or not all the normal–form and all the agent–form dynamics can be exponentially compressed without loss of information by means of sequence form. In particular, two well–known dynamics, BNN and Smith, have not been studied formulated in terms of sequence form so far and the derivation of polynomial formulations for these dynamics do not seem a straightforward application of the approach we used here.

References

Bloembergen, D.; Tuyls, K.; Hennes, D.; and Kaisers, M. 2015. Evolutionary dynamics of multi–agent learning: A survey. *J ARTIF INTELL RES* 53:659–697.

Blume, L. E. 1993. The statistical mechanics of strategic interaction. *GAME ECON BEHAV* 5(3):387–424.

Cressman, R.; Gaunersdorfer, A.; and Wen, J.-F. 2000. Evolutionary and dynamic stability in symmetric evolutionary games with two independent decisions. *INT GAME THEOR REV* 2(1):67–81.

Cressman, R. 2000. Subgame monotonicity in extensive form evolutionary games. *GAME ECON BEHAV* 32:183–205.

Cressman, R. 2003. *Evolutionary dynamics and extensive form games*. MIT Press.

Ferraioli, D. 2013. Logit dynamics: a model for bounded rationality. *SIGecom Exchanges* 12(1):34–37.

Fudenberg, D., and Tirole, J. 1991. *Game Theory*.

Gatti, N.; Panozzo, F.; and Restelli, M. 2013. Efficient evolutionary dynamics with extensive–form games. In *AAAI*.

Kuhn, H. 1950. Extensive games. *Proc. Natl. Acad. Sci. U. S. A.* 36:570–576.

Lanctot, M. 2014. Further developments of extensive–form replicator dynamics using the sequence–form representation. In *AAMAS*, 1257–1264.

Mckelvey, R., and Palfrey, T. 1998. Quantal response equilibria for extensive form games. *EXP ECON* 1(1):9–41.

Panait, L.; Tuyls, K.; and Luke, S. 2008. Theoretical advantages of lenient learners: An evolutionary game theoretic perspective. *J MACH LEARN RES* 9:423–457.

Panozzo, F.; Gatti, N.; and Restelli, M. 2014. Evolutionary dynamics of Q–learning over the sequence form. In *AAAI*, 2034–2040.

Sandholm, W. H. 2010. *Population Games and Evolutionary Dynamics*. MIT Press.

Selten, R. 1975. Reexamination of the perfectness concept for equilibrium points in extensive games. *INT J GAME THEORY* 4(1):25–55.

Tuyls, K., and Parsons, S. 2007. What evolutionary game theory tells us about multiagent learning. *ARTIF INTELL* 171(7):406–416.

Tuyls, K.; Hoen, P. J.; and Vanschoenwinkel, B. 2006. An evolutionary dynamical analysis of multi–agent learning in iterated games. *AUTON AGENT MULTI–AG* 12(1):115–153.

Vermeulen, D., and Jansen, M. 1998. The reduced form of a game. *EUR J OPER RES* 106(1):204–211.

von Neumann, J., and Morgenstern, O. 1944. *Theory of games and economic behavior.* Princeton University Press.

von Stengel, B. 1996. Efficient computation of behavior strategies. *GAME ECON BEHAV* 14(2):220–246.

Yang, R.; Ordóñez, F.; and Tambe, M. 2012. Computing optimal strategy against quantal response in security games. In *AAMAS*, 847–854.