Deep Learning to Collude

Clemens Possnig
Vancouver School of Economics, University of British Columbia

Abstract. This paper considers the limiting behavior for a general class of independent reinforcement learning (RL) algorithms in repeated games. We allow RL agents to learn state-dependent repeated game strategies and show that the limit points of their independent learning process act as an equilibrium selection mechanism. Asymptotic stability of equilibria of an underlying differential equation acts as the selection channel. Our class contains model-free actor-critic and gradient-based algorithms for continuous controls as special case. We allow for bias in the critic (gradient) estimator, give sufficient conditions and a full example of an algorithm in our class. Insights from this project can be used to determine under which conditions on the underlying game and algorithms collusive strategies may be learned. We argue that our framework opens up an important comparative statics exercise allowing to determine which types of learners, under which market and payoff conditions, are more likely to arrive at collusion.

JEL classification. C73, D43, D83.
Keywords. Reinforcement Learning, Stochastic Approximation, Repeated Games, Collusion, Learning in Games.

I thank my committee members Li Hao, Vitor Farinha Luz and Michael Peters for years of guidance and conversations. I also thank participants of the theory lunches at VSE for their extensive feedback and patience.

1. Introduction

Reinforcement learning (RL) algorithms are designed to learn from data and adapt actions toward an optimal policy with minimal assumptions about the underlying decision process. Such algorithms have become a prevalent tool for decision making in complex economic environments, with applications in airline pricing, gasoline pricing, stock markets, security force deployment, and more. Economists have become increasingly interested in how RL affects strategic behavior of firms and markets. Simulations of RL by economists have suggested that firms often end up learning how to collude with each other. This raises the theoretical question of how RL facilitates collusion. The aim of this project is to shed light on the complex process that appears to allow independent RL agents to learn to play collusive, state dependent strategies in a large class of repeated games with continuous actions. While holding in a broader environment, my results are of relevance for regulators interested in which types of learners, under which market and payoff conditions, are more likely to arrive at collusion.

Despite the fast growing economic literature on deep learning, machine learning and related advancements in computer science, there is still a lack of understanding in the effect RL can have on economic outcomes stemming from strategic interactions between such agents. So far, and perhaps to an extend due to the complexity of the problem, most attempts at an answer have been either empirical or simulation-based.
This project attempts to fill this gap. I aim to shed light on the analytical underpinnings of the effect large scale deployment of RL agents has on economic outcomes.

(1) I provide a framework to study the limiting behavior of a general class of RL agents who compete in strategic environments.
(2) I show that limiting strategies of RL in my class act as a selection mechanism among repeated game strategies, based on details of the game and algorithms in play.
(3) I argue that this framework opens up an important comparative statics exercise: Changes in the game can affect the set of possible limiting behaviors observed when RL agents play.

My class of RL agents contains, among others, actor-critic and gradient-based algorithms playing on continuous action spaces and discrete state spaces. I allow agents to maintain a possibly biased estimator of their current gradient or value function (the critic), and update policies (the actor) towards a presumed optimal direction. Such agents are able to support state-dependent repeated game policies. The RL agents can maintain estimates of their payoff functions and distributions of observed shocks, but do not have explicit models of

their opponent behavior or payoffs. This can best be thought of as a model free solution to a firm's problem of finding an optimal behavior based on observed state variables, when information about competitors is hard to come by or the risk of misspecifying an opponent model is deemed too high. I provide sufficient conditions for algorithms to fall into my class as well as an example algorithm that satisfies those conditions.

While my analysis applies to general repeated continuous action games, its potential is perhaps best shown via the example of competing firms and collusion. The case of competing algorithms has attracted attention from economists in the recent years. Both empirical evidence (Assad et al. 2020) and numerical simulations (Klein 2021, Calvano, Calzolari, Denicolo, et al. 2020,Calvano, Calzolari, Denicoló, et al. 2021) suggest that a market composed of reinforcement algorithms may be vulnerable to collusive pricing strategies. Their evidence suggests that RL agents in the limit may not only be able to sustain supra-competitive profits, but also learn correctly to play repeated game strategies akin to typical strategies analysed in the literature on repeated games. One can see from their observations that not all payoffs of the repeated game that are supportable through a Folk theorem are observed as limiting payoffs of the RL agents. Simulations however can take us not much further than that.

My analytical framework allows to ask precisely: which payoffs of the repeated game are supportable? And what strategies that support those payoffs are feasible limiting strategies of the RL agents? Furthermore, my framework will allow us to evaluate how robust the simulation results are to choices of the underlying game and details of the algorithms playing that game.

Specifically, my analysis shows which equilibrium strategies can be selected in the limit as RL agents play. Arguments from stochastic approximation theory allow me to connect the limiting process of RL strategy profiles to an underlying differential equation that depends on details of the game and the algorithms involved. I show that, given the state space agents play on, limits of the process of RL strategy profiles get selected based on their asymptotic stability with respect to that underlying differential equation. In the case of actor-critic Q-learning, this differential equation is a state-dependent best response dynamic, where best responses are computed among the set of stationary strategies with respect to the fixed state space.

To fix ideas, consider the state space only consisting of the previous period's price as in Calvano, Calzolari, Denicoló, et al. 2021's imperfect public monitoring game. My result then implies that if an equilibrium is asymptotically stable with respect to a state-dependent best response dynamics, then there is positive probability that it will be reached by the algorithms. Conversely, if it is unstable, it can never be a limiting point of algorithm play. We can then ask how the price distribution, and cost functions of the firms affect the stability of collusive equilibria of that game given the state space. This will allow to understand details of games as more or less amenable to collusion among specific types of algorithms.

Relation to the Literature

Broadly speaking, this project speaks to results in asymptotic behavior of algorithms in the computer science literature, the classical theory of learning in games, as well as the theory of repeated games and equilibrium refinements.

Firstly, this paper makes use of an extensive body of research related to stochastic approximation theory (see for example Borkar 2009) and hyperbolic theory (Palis Jr, Melo, et al. 1982). There is a growing strand of the computer science literature devoted to establishing convergence proofs in multi agent algorithmic environments. The paper in that area closest to this one is by Mazumdar, Ratliff, and Sastry 2020. They establish a connection between gradient-based learning algorithms for continuous action games and asymptotic stability of equilibria of the underlying game. While nested in our RL class, the updating rules that Mazumdar, Ratliff, and Sastry 2020 consider implicitly assume that agents observe each other's per period policies, or at least observe an unbiased estimator of their per-period value function gradient. I argue that this assumption is difficult to satisfy, especially in the case of continuous action games. I give lower level sufficient conditions algorithms must satisfy in order to fall into the class I consider, and also provide a full example of an algorithm that satisfies all those conditions. My results suggest that Mazumdar, Ratliff, and Sastry 2020's results are robust to the type of bias in the gradient estimation that my RL class allows.

Other papers related to asymptotic analysis of multi-agent systems commonly focus on one specific algorithm, allow communication across agents, require information on the primitives of the game, or do not ask about the nature of the limiting points. Notably, Ramaswamy and Hullermeier 2021 give a more general treatment on asymptotic analysis of RL without considering stability properties of rest points. Others focus on specific classes of games, for example zero sum games (Sayin et al. 2021) and show convergence of multi-agent learning

there.

Leslie, Perkins, and Xu 2020's paper takes us from project intended for computer scientist audiences to those more intended for economists. They consider zero-sum Markov games and construct an updating scheme related to best response dynamics that converges to equilibria of the game. As they also keep track of separate policy and value function updates, their scheme falls into the class of actor-critic learning rules. Perhaps more due to notation and framing than due to content, Leslie, Perkins, and Xu 2020's paper is considered as research in the theory of learning in games much more so than algorithmic learning theory.

When it comes to the theory of learning in games more generally, this paper looks into the ability of agents following a heuristic, uncoupled learning rule to learn how to play repeated game strategies. I impose an informational constraint on agents, namely that they be unable to observe other agent's payoffs or actions, and not able to build a model of their opponent's behaviors. The RL class I consider can thus be seen as agents following uncoupled learning rules as defined in Hart and Mas-Colell 2003. However, the fact that in my paper agents learn to play repeated game policies is in contrast to the classical game theoretic learning literature, that generally considers the process of learning to play static game Nash equilibria. In that sense, this project sheds light on the ability of agents to learn to play equilibria of a repeated game other than the static equilibrium of the underlying stage game. At the same time, since the agents I consider learn policies on a fixed state space, one may recast their payoffs as expected discounted payoffs based on stationary strategy profiles that can only condition on that state space. Taking that view, one can say that agents in my class of RL algorithms learn to play Nash equilibria of a repeated stage game with multi-dimensional continuous actions. In this sense my analysis ties neatly into classical analysis of the theory of learning in games with minimal information requirements.

Finally, this paper casts RL competition as an equilibrium selection mechanism. It is a common observation in the learning literature that when agents follow heuristic learning rules, they may not be able to learn to play all possible equilibria of the game, or not even converge to an equilibrium. This paper is no different, and also allows for the possibility of games for which agents may converge to cycles. However, the classical literature was developed as model to understand how rational agents may learn to play Nash equilibria, whereas here we consider real economic agents that happen to be algorithmic and show that their behavior can be understood through the theory of learning in games. We refer to

Fudenberg and Levine 2009 for an excellent review of issues regarding the theory of learning in games, including algorithmic learning and applications of stochastic approximation.

The paper is structured as follows: In section 2 we define the general class of RL agents we analyse, and provide general limiting results in section 3. In section 4 we define the specific game environment we want to analyse our class of RL on. In section 5 we define actor-critic deep Q-learning as important motivation and give sufficient conditions for such learners to be part of our class when playing our game. In section 6 we argue how our class of learners act as equilibrium selectors. Section 7 concludes.

2. Modelling Reinforcement Learning

We first give a set of mathematical definitions we will use throughout the paper.

2.1. Mathematical Preliminaries

Let X, Y be two metric spaces.

- Let $\mathcal{C}^i[X, Y]$ be the set of functions that is i times continuously differentiable, with domain X and range Y.
- For $\gamma > 0$, let $\mathcal{B}^i_\gamma$ be the set of C^i functions with bounded derivatives :

$$\mathcal{B}^i_\gamma = \left\{ g : E \mapsto E; \mid \sup_{x \in E} \|g(x)\| + \sum_{j=1}^{i} \sup_{x \in E} \|D^j g(x)\| \leq \gamma \right\}, \tag{1}$$

  where $D^j g$ represents the jth derivative.
- For any set B, define $conv[B]$ as the convex closure.
- The Hausdorff distance between sets A, B is defined as

$$H(A, B) = \max \left\{ \sup_{x \in A} d(x, B), \sup_{x' \in B} d(x', A) \right\},$$

  with

$$d(x, A) = \inf_{x' \in A} \|x - x'\|.$$

2.2. Preliminaries to RL

We will define a model of policy-updating algorithms that perform well up to some bias. We model RL agents that update policies using information gathered about an underlying value function of the problem they're facing. In general we assume that these agents don't know the true value function of the repeated game, and neither the stage game. In such

cases, RL algorithms update policies using some estimate of a value function or at least a value-increasing direction. There are multiple reasons why this is a difficult situation for such agents when it comes to learning a good policy.

The fact that there are multiple agents involved in updating independent policies implies that each agent learns in a nonstationary environment. Nonstationarity means that agents are trying to follow a moving target, which can cause value function approximations to be inconsistent. Furthermore, as we consider learning under continuous controls, unbiasedly estimating a value function becomes an even more daunting task. Very commonly in such situations agents use some form of parametric function approximation to generate an estimate, which can introduce bias. Often that involves deep neural networks due to their flexibility and scaleability. We will be abstract about the estimation of the value function and therefore introduce a class of algorithms that perform reasonably well in the function approximation step, according to our conditions.

We believe that allowing for a bias significantly increases the number of learning algorithms that fall into our class of RL agents, due to the inherent problems these agents face while learning, as outlined above. We refer to François-Lavet et al. 2018 for an excellent introduction to state of the art RL techniques and a deeper dive into issues of biased estimation of value functions and their gradients. We will show that the bias we allow in our class does not affect the main results in the next section.

First we need to define the class of functions to be approximated:

Definition 1. *We define the set $\mathcal{M}^1$ of (possibly multivalued) maps G with compact domain $X \subset \mathbb{R}^k$ and range $\mathcal{P}[R]$ for compact set $R \subset \mathbb{R}^d$ s.t.*

- *$G(x) \subset R$ is convex, compact valued.*
- *There exists $c > 0$ such that $\sup\{\|y\| : y \in G(x)\} \leq c(1 + \|x\|)$ for all $x \in X$, i.e. linear growth.*
- *There is a union of connected sets $C \subseteq X$ of positive measure, $\mathcal{U}_G = \bigcup C$, such that $G(x)$ is C^1 for $x \in \mathcal{U}_G$.*

Remark 1. *We allow for multivaluedness to be able to handle to common learning scheme of actor-critic Q-learning, which maintains estimates of the argmax of a value function. Note however that $\mathcal{C}^1 \subset \mathcal{M}^1$.*

Definition 2 (C^1 Approximation)**.**

Let Y be some space of observations to be used to approximate a function. Given $\gamma > 0$, we say that a function approximation operator $\mathcal{A}_g : \mathcal{M}^1 \times Y \mapsto \mathcal{M}^1$ is a C^1 Approximation of a $G \in \mathcal{M}^1$ if there is an increasing sequence of σ-fields $\mathcal{F}_n$ generated by datasets $D_n \in Y$, an error function $g \in \mathcal{B}^1_\gamma$ and an integer $N > 0$ such that we can write for all $n \geq N$:

(i) For all $x \in X$,

$$H\big(G(x),\, \mathcal{A}_g[G, D_n](x)\big) < \gamma + \delta(x, D_n),$$

where $\delta(x, D_n) \geq 0$ is such that $\sup_{x\in X} \delta(x, D_n) \to_p 0$ as $n \to \infty$. "$\to_p$" denotes convergence in probability.

(ii) For all $x \in \mathcal{U}_G$,

$$\mathcal{A}_g[G, D_n](x) = G(x) + g(x) + R(x, D_n),$$

with $g \in \mathcal{B}^1_\gamma$, and $R(x, D_n)$ is a (possibly singleton) set such that

$$\sup_{x\in X} \sup_{\delta_n(x)\in R(x,D_n)} \|\delta_n(x)\| \to_p 0,$$

as $n \to \infty$.

One can interpret $g(x)$ as representing the bias part of the function approximation, and $\delta(x, D_n)$ as a random variable such that $\mathbb{E}[\|\delta(x, D_n)\|^2 \,|\, \mathcal{F}_n]$ represents the variance part.

In the case of Q learning, D_n only needs to consist of $(s_t, a_t, r_t, s_{t+1})_{t=1}^n$, i.e. past observations of states, actions, payoffs, state transitions, and the initial Q_0.
Generally one can think of $\mathcal{A}_g[G, D_n](\cdot)$ as a parametric or non-parametric function approximation, with bounded errors that can be approximated by a small C^1 function after enough data (large n) has been accumulated.

2.3. A class of Reinforcement Learners

Here we provide a general model of reinforcement learning abstracting away from underlying details of the environment that is being played on. We assume there is a set of I agents with $|I| = n$. Agents observe states on some fixed, finite state space S with $|S| = k$, and make per period choices (actions) in compact interval A_i. They are thus able to iterate over policies $x^i \in \bar{A}_i = A_i^k$, with policy profile space $E = \times_{i\in I} \bar{A}_i$. Agents then follow a fixed rule (algorithm) to update their strategy profiles over time.

Definition 3. *Given $g \in \mathcal{B}^1_\gamma$ and observation space Y, let $D_n \in Y$ be a sequence of datasets and $\mathcal{A}_g[F, D_n]$ be a C^1 approximation of $F(x) \in \mathcal{M}^1$ (3). Then for profiles $x_n \in E$ we model our algorithm as*

$$x_{n+1} = x_n + \alpha_n \left[\mathcal{A}_g[F, D_n](x_n) + M_{n+1}\right], \tag{2}$$

We assume:

(1) *Agent are independent and do not communicate. Each agent i runs a separate updating scheme*

$$x^i_{n+1} = x^i_n + \alpha_n \left[\mathcal{A}^i_g[F, D_n](x^i_n) + M^i_{n+1}\right],$$

such that $\mathcal{A}_g[F, D_n](x_n)$ is the stacked vector of function approximators $\mathcal{A}^i_g[F, D_n]$ and similarly for shocks M^i_{n+1}. The stacked result $\mathcal{A}_g[F, D_n](x_n)$ is a C^1 approximation of $F(x)$, so we can write the overall updating to profiles x_n as in 2.

(2) *$\mathcal{F}_n$ is the σ-field generated by $\{x_n, D_n, M_n, x_{n-1}, D_{n-1}, M_{n-1}..., x_0, D_0, M_0\}$, i.e. all the information available to the updating rule at a given period n.*

(3) *M_{n+1} are shocks the algorithm designer generates in order to induce exploration. There is $0 < \bar{M} < \infty, q \geq 2$ such that for all n*

$$\mathbb{E}[M_{n+1} \,|\mathcal{F}_n] = 0; \quad \mathbb{E}[\|M_{n+1}\|^q \,|\mathcal{F}_n] < \bar{M}$$

(4) *Support condition: Recall $K = \sup A$, the upper bound of the action set. For n large enough,*

$$-\frac{x_n}{\alpha_n} - \mathcal{A}_g[F, D_n](x_n) \leq M_{n+1} \leq \frac{K - x_n}{\alpha_n} - \mathcal{A}_g[F, D_n](x_n)$$

holds almost surely, conditional on $\mathcal{F}_n$. Since the algorithm designer samples M_{n+1} themselves, this can always be satisfied.

(5) *Whenever $x_n \in \mathcal{U}$,*

$$\Omega_n \equiv \mathbb{E}[M_{n+1}M'_{n+1} \,|\mathcal{F}_n],$$

where Ω_n is symmetric positive definite for all n.

(6) *Write $\varepsilon_n = \mathcal{A}_g[F, D_n](x_n) - F(x_n)$. Then M_{n+1}, ε_n are independent conditional on $\mathcal{F}_n$.*

(7) *Robbins-Monro Condition on stepsizes:*
$\alpha_n \to 0$ with

$$\sum_{n=0}^{\infty} \alpha_n = \infty; \quad \sum_{n=0}^{\infty} \alpha_n^2 < \infty.$$

Note that we have assumed that $F(x)$ is single valued only on $\mathcal{U}$. If for some $x \notin \mathcal{U}$, $F(x)$ is not a singleton, we allow the algorithm to pick an arbitrary selection.

Remark 2. *The only assumption above that is non-standard with respect to the algorithmic learning literature is item 4. This assumption is made to ensure that updates stay within their compact strategy spaces. When noise is generated by the algorithm as a means of exploration, at every period this can be satisfied by drawing from a support that satisfies*

this condition. There are multiple different options when it comes to exploration of continuous control spaces. The method assumed here falls into the case of 'parameter space noise' (Plappert et al. 2017). It is sufficient for the results to go through, but alternative interpretations of M_{n+1} exist, such as noise generated by the function approximation $\mathcal{A}$.

Remark 3. *Notice that Definition 2 does not exclude the case in which the function to be approximated is fully known, or there is no bias term. Our results thus include the case where agents know their value functions and follow a simple heuristic in updating their payoffs, taking as an input the current strategies of their opponent. In the case where $F(X)$ is a gradient, this scenario is similarly treated in Mazumdar, Ratliff, and Sastry 2020. We also refer to Mazumdar, Ratliff, and Sastry 2020 for a list of classes of algorithms that are included in our definition.*

3. Limiting Behavior

Definition 4. *Take the algorithm 2. The limit set is defined as*

$$L_g = \bigcap_{n \geq 0} \overline{\{x_s \, | s \geq n\}},$$

the set of limits of convergent subsequences x_{t_k}. We write g as subscript to underline the dependence on bias function g.

Definition 5. *Let x^* be a rest point of $F(x)$, and $= eigv[DF(x^*)]$ the set of eigenvalues. For a complex number z, let $\mathbf{Re}[z] \in \mathbb{R}$ be the real part. x^* is*

- *Hyperbolic if $\mathbf{Re}[\lambda] \neq 0$ holds for all $\lambda \in \Lambda$.*
- *Asymptotically stable if $\mathbf{Re}[\lambda] < 0$ holds for all $\lambda \in \Lambda$.*
- *Linearly unstable if $\mathbf{Re}[\lambda] > 0$ holds for at least one $\lambda \in \Lambda$.*

Proposition 1. *With probabilitiy one, L_g is an internally chain transitive (ICT) set[1] of the differential inclusion*

$$\dot{x} \in F_g(x(t)) \equiv conv[F(x(t))] + g(x(t)).$$

Proof Sketch of Proposition 1

The full proof for this and the following Propositions can be found in Appendix A. This proof follows from celebrated results in stochastic approximation theory. In a nutshell, we relate a time-interpolated version of the recursion x_n in 2 to the solution of the differential inclusion in Proposition 1. The limiting behavior of x_n can then be deduced from a subset

[1]Importantly, these sets include rest points and limit cycles (if they exist). We refer to Benaim, Hofbauer, and Sorin 2005 Definition 6 for a definition, and Papadimitriou and Piliouras 2018 for an intuitive discussion.

of the limiting behaviors of the differential inclusion, which are precisely the internally chain transitive sets.

Proposition 2. *Let $x^* \in \mathcal{U}_F$ be asymptotically stable for F. Then for all γ small enough and all $g \in \mathcal{B}^1_\gamma$ there is a profile x^g such that*

(1) $\sup_{g \in \mathcal{B}^1_\gamma} |x^g - x^*| \to 0$ *as* $\gamma \to 0$.
(2) $P[L_g = \{x^g\}] > 0$.

Proof Sketch of Proposition 2

The proof first establishes a firm connection between x^* and x^g. We use a more general version of the inverse function theorem to show that since $g(x)$ is a well behaved, differentiable bias term, for every x^* there is a unique rest point x^g. Further, stability of x^* must carry over to stability of x^g. Then we use the stochastic approximation method to relate, for large enough n, the recursion 2 to the solution of the differential inclusion defined in Proposition 1. Once it is established that x_n tracks solutions to such a differential system over time, it is then intuitive that attracting points of the differential system will also attract x_n over time.

Proposition 3. *Let $x^* \in \mathcal{U}_F$ be linearly unstable for F. Then for all γ small enough and all $g \in \mathcal{B}^1_\gamma$ there is an open neighborhood U_γ with $x^* \in U_\gamma$ such that*

$$P[L_g \in U_\gamma] = 0.$$

Proof Sketch of Proposition 3

Firstly, as in the proof of Proposition 2, we establish a one to one relationship between the stability properties of x^* and the rest points x^g. x^g being unstable hyperbolic implies that there exists an unstable manifold that x^g lies on, which acts as a repeller to the differential inclusion F_g. We go on to show that due to the instability of x^g and nonvanishing variance of M_{n+1}, no matter how close the algorithm updates come to x^g, and no matter how large n is, there is always a high probability that x_n lands on the unstable manifold and therefore must move away from x^g. Finally we show the existence of a neighborhood U_γ. We show that due to the hyperbolicity of x^*, x^g, there is a neighborhood U around x^g with $x^* \in U$ such that x^g is the only internally chain transitive set within U. We recall that x^* is not internally chain transitive for the perturbed system F_g, and the result follows.

4. The Game Application

Having given technical results connecting limiting policies of general RL algorithms to stability of rest points of underlying differential equations, we can take a closer look at

what specific interactions multiple RL agents could have, and how those can translate to an underlying differential equation. We first define the underlying environment the agents can play on.
Throughout, it is important to keep in mind that we are defining an environment played on not by rational agents, but by algorithms constrained to play a certain type of policies.

Definition 6 (The Game Played by Algorithms)**.**

- *Set of agents* I, $|I| = N$.
- *Common finite state space* S *with* $|S| = k$.
- *Interval-action space* $A_i = [0, K_i]$ *with some large* $K_i < K < \infty$. $A = \times_{i \in I} A_i$
- *Stationary strategy space based on* S: $\bar{A}_i = A_i^k$.
- *Strategy profiles in* $E = \times_{i \in I} \bar{A}_i$.
- *Stage game payoff function* $u^i(r, s)$, C^2 *in* $r \in A$.
- *States transition from* s *to* s' *according to controlled Markov Transition kernel* $P_{ss'}(r)$ *for* $r \in A$.
- *For every* $r \in A$, *states follow an irreducible positive recurrent Markov chain with stationary distribution* $\lambda(s, r)$.
- *There exists* $c \in (0, 1)$ *such that* $\lambda(s, r) > c$ *for all* $s \in S, r \in A$.

At first glance, the description of the game above may remind one of a stochastic game. Stochastic games, which take the notation $u^i(r, s)$ seriously and by definition take the state as stage-game payoff relevant are included. The notation however allows us be more flexible than that. Firstly, recall that RL agents in our class (3) are defined to be constrained to playing strategies that condition on a fixed, pre-specified state space. This affects how we define the strategy space for the game above. A different state space means not only a different game-form, but also different available strategy sets. We are silent about how this state space came to be and we do not model any incentives to design a state space.
Secondly, as we take this pre-specified state space seriously, we can include in our definition also auxiliary games that arise from players playing finite automaton strategies given a number of k states. In that case we consider the subpproblem of finding optimal strategies within the set of stationary strategies there.
I give three examples that are within the breadth of Definition 6:

Example 1.

(1) $S = Y$, *where* Y *is a set of outcomes a random variable can take that affects the payoffs of the current stage game. This would be a stochastic game.*

(2) $s_t = p_{t-1}$*, where* p_t *is the realization of a random variable can take that affects the payoffs of the stage game in period* t*. Note that* s_t *is not payoff relevant, but gives a constraint on the strategies RL agents can play (akin to bounded-recall strategies).*

(3) $S = \{C, D\}$ *is a set abstract, automaton states. One can think of equilibria in which* $s = C$ *inspires agents to collude, while* $s = D$ *initializes punishment.*

For any $i \in I$, let $\bar{A}_{-i} = \times_{j \neq i} \bar{A}_j$. We can define repeated game payoff functions $W^i(x_i, x_{-i}, s_0)$ given stationary strategy profiles $[x_i, x_{-i}] \in E$:

$$W^i(x_i, x_{-i}, s_0) = \mathbb{E} \sum_{t=0}^{\infty} \delta^t u^i(x(s_t), s_t).$$

Next, define $B_S^i(x^{-i})$ as the optimal strategy given a profile $x^{-i} \in \bar{A}_{-i}$, chosen from the constraint set of stationary, S-state strategies:

$$B_S^i(x^{-i}) = argmax_{x \in \bar{A}_i} W^i(x, x_{-i}, s_0),$$

where due to the irreducibility assumption in Definition 6 the optimal strategy does not depend on the initial state s_0. We let $\bar{B}_S(x)$ be the stacked optimal strategy, stacked over i.

We introduce these concepts because they will allow us to make sense of the strategies and updating that the algorithms we consider are able to make. This will become especially apparent when considering the subclass of actor-critic Q-learning as will be introduced in subsection 5.

Assumption 1 (Equilibrium existence and differentiability)**.**

- *We assume stationary equilibrium profiles* $x^* \in E$ *exist on state space* S*. Call the set of such equilibria* E_S*.*
- *For all* $x^* \in E_S$*,* x^* *are interior to* E *and there is an open neighborhood* U_{x^*} *with* $x^* \in U$ *such that* $B_S(x)$ *is single valued for all* $x \in U_{x^*}$*.*

We define $\mathcal{U} = \bigcup_{E_S} U_{x^*}$. A sufficient condition for the first point in Assumption 1 to hold, is the existence of a static Nash equilibrium given $u(r, s)$ for all $s \in S$. As for the second point, such equilibria x^* are sometimes referred to as 'differential Nash equilibria'. A sufficient condition for such equilibria would be that the Hessian of each agent's optimization problem at the equilibrium be negative definite. As our analysis of limiting strategies will depend on a smoothness condition of an underlying differential equation at the given rest point, this assumption will prove crucial.

Next, for $x \in E$

$$F_B(x) = \bar{B}_S(x) - x, \tag{3}$$

be the state dependent best response dynamics gradient field. This gradient field will be a useful example of limiting differential equation for a prominent class of RL algorithms within our class. In the next section we give an example of important algorithms that fall into our class, and for which 3 would be the limiting differential equation.

5. Example: Actor-Critic DQN Learning

The RL class we defined in Definition 3 is given in general terms and it is not obvious to see which algorithms can satisfy it. This section gives an example of very common RL agents known as actor-critic, and a sufficient condition on their learning behavior so that they satisfy our Definition 3 when playing a game as defined in 6.

In line with a canonical problem in reinforcement learning and also the main problem we want to study, we will consider the task of approximating the maximizer of $Q(s, a, y)$ action-value function for a game as stated in Definition 6. This function is defined as the fixed point of a Bellman equation:

$$Q^i(s, a, y) = u^i(s, a, y(s)) + \delta \sum_{s' \in S} P_{ss'}(a, y(s)) \max_{a' \in A_i} Q^i(s', a', y).$$

We include opponent policy profile $y \in \bar{A}_{-i}$ as an argument to be clear about the dependence of payoffs on opponent's profiles.

This class of algorithm is called actor-critic because it iterates over two separate objects, where one the update of one is based on the other. Every period, it updates both an estimate of the Q-function (the critic), and the policy (the actor) using the estimate of the Q-function.

Define the argmax operator $\mathcal{T}$, so that

$$\mathcal{T} Q(s, a, y) = argmax_{a' \in A} Q(s, a, y).$$

Here $\theta_n \in \Theta \subset \mathbb{R}^D$ is element of a sequence of large but finite dimensional vectors that pin down the DQN approximation. For a concise introduction, see Mnih et al. 2015 or chapter 3 in Busoniu et al. 2017.

Suppose we consider agents whose policy profiles x^i_n change according to the following recursion:

$$x^i_{n+1}(s) = x^i_n(s) + \alpha_n \left[\mathcal{T} \tilde{Q}^i(s, a, \theta^i_n) - x^i_n + M_{n+1} \right], \tag{4}$$

for all $s \in S$, where we take our assumptions in Definition 3 to hold for all i, letting $\mathcal{T} \tilde{Q}^i(s, a, \theta^i_n) - x^i_n$ take the place of the approximation operator $\mathcal{A}^i_g[F^i, D^i_n](x^i_n)$ in that definition. For convenience we drop the i- superscript as often as possible.

We define a (possibly growing) window size $B_n < n$ for all n and define

$$D_n = \{(s_t, a_t, r_t, s_{t+1}) : n - B_n + 1 \leq t \leq n\},$$

as the B_n-sized data set used to construct $\tilde{Q}(s, a; \theta_n)$. D_n is often referred to as 'experience-replay buffer'. Using this set, data points are sampled to update θ_n at every period, for example according to some gradient descent procedure (Mnih et al. 2015).
Define the space of functions in the range of the DQN approximation as

$$\mathcal{DQ} = \left\{ Q(s, a; \theta) : S \times A \times \Theta \mapsto \mathbb{R} \middle| \text{ Q is twice differentiable in } (a, \theta) \right\}.$$

Assumption 2 (Q-approximation: Sufficient Conditions)**.**
Let x_n^i be generated by 4 for all i. We assume

(i) For all i, s, $conv[\mathcal{T}Q(s, a, y)] \in \mathcal{M}^1$. (See Definition 1)

(ii) $\tilde{Q}(s, a; \theta_n)$ is a synchronous DQN approximation, based for example on Mnih et al. 2015. See chapter 3 in Busoniu et al. 2017 for a review of successful Q- function approximators.

(iii) There is $g_Q \in \mathcal{B}^2_{\gamma_Q}$ such that

$$\sup_{s,a} \left\| \tilde{Q}(s, a; \theta_n) - \left[Q(s, a, x_n^{-i}) + g_Q(s, a, x_n^{-i}, \theta_n) \right] \right\| \to_p 0,$$

as $n \to \infty$.

(iv) For all $x \in \mathcal{U}, \theta \in \Theta$ and all i, s, $D^2 Q^i(s, x^i(s), x^{-i}) + D^2 g_Q^i(s, x^i(s), x^{-i}, \theta)$ has full rank. D^2 is the second derivative in a.

Remark 4. *Note that point (iii) is doing the main work among these assumptions. The convergence behavior of DQN is a complex problem, and only very recently has there been progress in providing asymptotic analysis in general settings. See for example Ramaswamy and Hullermeier 2021.*
In the best case with respect to asymptotics, one can have that $\tilde{Q}(s, a; \theta_n)$ converges to the function in $\mathcal{DQ}$ closest to $Q(s, a, x_n^{-i})$. Define

$$\tilde{Q}(s, a; \theta^*(x_n^{-i})) \equiv \min_{\theta \in \Theta} \sup_{(s,a)} \left\| \tilde{Q}(s, a; \theta) - Q(s, a, x_n^{-i}) \right\|.$$

If we then have that

$$\sup_{(s,a)} \left\| \tilde{Q}(s, a; \theta_n) - \tilde{Q}(s, a; \theta^*(x_n^{-i})) \right\| \to_p 0,$$

as $n \to \infty$, (iii) follows:
DQN involves nonlinear transformations from (s, a, θ) to outputs. Under mild regularity conditions on $\mathcal{DQ}$ and $Q(s, a, x_n^{-i})$, it can then be seen that the difference $\tilde{Q}(s, a; \theta^*(x_n^{-i})) - Q(s, a, x_n^{-i}) \in \mathcal{B}^2_{\gamma_Q}$, in which case point (iii) is satisfied.

Lemma 1. *Suppose Assumption 2 holds. Then conv*$[\mathcal{T}\tilde{Q}(s,a;\theta_n)]$ *satisfies Definition 2.*

Proof Sketch of Lemma 1
The proof makes use of the assumed good convergencde behavior of $\tilde{Q}$ (Assumption 2) as well as the upper hemicontinuity of the argmax function with respect to the supremum norm in bounded function spaces. Using these two tools, we show how the argmax of the random function approximator must approach the argmax of the best parametric approximation of the true Q-function.

To show that Assumption 2 is satisfied by a nonempty set of algorithms, we refer to Appendix B for a fully developed example of an actor-critic learning algorithm that satisfies it.

6. Equilibrium Selection

Now that we are equipped with a good intuition of the class of algorithms defined in 3, we can consider results on their limiting behavior more concretely.

Definition 7. *A profile x is an ε-equilibrium if for all players i all individual profiles $x' \in \bar{A}$ and states $s \in S$*

$$W^i(x,s) \geq W^i(x', x_{-i}, s) - \varepsilon.$$

Corollary 1. *In the case where $F = F_B$, let $x^* \in E_s$ be asymptotically stable for F_B (3). Then for all γ small enough and all $g \in \mathcal{B}^1_\gamma$ there is a $\bar{\varepsilon} > 0$ and a profile x^g such that*

(1) x^g is an ε-equilibrium for all $\varepsilon \geq \bar{\varepsilon}$
(2) $\sup_{g \in \mathcal{B}^1_\gamma} |x^g - x^*| \to 0$ as $\gamma \to 0$.
(3) $P[L_g = \{x^g\}] > 0$.

If $x^* \in E_s$ is unstable for F_B, for all γ small enough and all $g \in \mathcal{B}^1_\gamma$ there is an open neighborhood U_γ with $x^* \in U_\gamma$ such that

$$P[L_g \in U_\gamma] = 0.$$

6.1. Discussion

Corollary 1 shows the full potential of our framework. It allows to interpret algorithms in our class 3 as equilibrium-selection mechanism. Asymptotically stable equilibria are equilibria that can be limiting points of the RL learning game, while unstable equilibria are not. The intuition is related to how RL learn to play: since such agents make errors

by construction and also to explore their action space, opponent's strategy profile are constantly perturbed. In other words, out of the view of a fixed agent i, the other agents are frequently deviating to policies nearby in the policy space. Now suppose the current profile x_n is close to an equilibrium x^*. Since i's updating rule tracks F_B, their policy will only stay close to x^* if the dynamics of F_B are somehow robust to deviations. This robustness is implied by asymptotic stability, and broken by unstable equilibria.
There is a caveat here however: Corollary 1 does not state that all limiting points in L_g will be equilibria of the game. Depending on details of the game, we may or may not be able to rule out the case where algorithm updates get trapped in a cycle, or other more complex behavior not involving rest points (see Papadimitriou and Piliouras 2018). We do not include cycles in the above definition, however it is straightforward to extend Proposition 2 to the case of attracting cycles as in Faure and Roth 2010, and there exist results considering linearly unstable cycles (Benaım and Faure 2012) that suggest one may extend Proposition 3 to such linearly unstable cycles also.

For now, I have not extended my results to include cycles as I believe it of second order importance to the understanding of stability properties of equilibrium rest points, which is the main focus of this paper. I consider an extension of the result as interesting avenue for further research.

Now let us restrict attention to the equilibrium limiting points of the algorithm learning process. Importantly, recall that asymptotic stability of rest points is equivalent to an eigenvalue condition as defined in 5. This gives rise to the possibility of an interesting comparative statics exercise: how does the stability of a given set of equilibria change as we change parameters of the game?
The question boils down to the perturbation theory of eigenvalues of the linearization of F_B at an equilibrium of interest. For example, given a fixed state space S, one can characterize best equilibrium (with respect to payoffs) under that state space, or the most collusive equilibrium (with respect to average quantities). One can then observe how the stability of these equilibria changes as for example the elasticity of demand changes. This analysis is a main focus of further research of the author.

7. Conclusion

This paper considers the limiting behavior of a broad class of RL algorithms and shows that one can interpret these algorithms as equilibrium selection mechanism. By ways of

the example of collusion in repeated games, I observe the usefulness of this framework: it allows one to consider comparative statics exercises with respect to details of the game played by the RL agents. These comparative statics will allow to understand the change in potential limiting behavior or RL algorithms when their game environment changes. Potential applications include the prevalence of collusive limiting behavior when changing demand elasticities, stochastic components of the firm's payoffs, or firm's cost functions.

References

Assad, Stephanie et al. (2020). "Algorithmic pricing and competition: Empirical evidence from the German retail gasoline market". In:

Benaım, Michel and Mathieu Faure (2012). "Stochastic approximation, cooperative dynamics and supermodular games". In: *The Annals of Applied Probability* 22.5, pp. 2133–2164.

Benaım, Michel, Josef Hofbauer, and Sylvain Sorin (2005). "Stochastic approximations and differential inclusions". In: *SIAM Journal on Control and Optimization* 44.1, pp. 328–348.

Borkar, Vivek S (2009). *Stochastic approximation: a dynamical systems viewpoint*. Vol. 48. Springer.

Busoniu, Lucian et al. (2017). *Reinforcement learning and dynamic programming using function approximators*. CRC press.

Calvano, Emilio, Giacomo Calzolari, Vincenzo Denicolo, et al. (2020). "Artificial intelligence, algorithmic pricing, and collusion". In: *American Economic Review* 110.10, pp. 3267–97.

Calvano, Emilio, Giacomo Calzolari, Vincenzo Denicoló, et al. (2021). "Algorithmic collusion with imperfect monitoring". In: *International journal of industrial organization* 79, p. 102712.

Chicone, Carmen (2006). *Ordinary differential equations with applications*. Vol. 34. Springer Science & Business Media.

Faure, Mathieu and Gregory Roth (2010). "Stochastic approximations of set-valued dynamical systems: Convergence with positive probability to an attractor". In: *Mathematics of Operations Research* 35.3, pp. 624–640.

François-Lavet, Vincent et al. (2018). "An introduction to deep reinforcement learning". In: *arXiv preprint arXiv:1811.12560*.

Fudenberg, Drew and David K Levine (2009). "Learning and equilibrium". In: *Annu. Rev. Econ.* 1.1, pp. 385–420.

Hansen, Bruce E (2010). *Econometrics. University of Wisconsin.*

Hart, Sergiu and Andreu Mas-Colell (2003). "Uncoupled dynamics do not lead to Nash equilibrium". In: *American Economic Review* 93.5, pp. 1830–1836.

Klein, Timo (2021). "Autonomous algorithmic collusion: Q-learning under sequential pricing". In: *The RAND Journal of Economics* 52.3, pp. 538–558.

Leslie, David S, Steven Perkins, and Zibo Xu (2020). "Best-response dynamics in zero-sum stochastic games". In: *Journal of Economic Theory* 189, p. 105095.

Mazumdar, Eric, Lillian J Ratliff, and S Shankar Sastry (2020). "On gradient-based learning in continuous games". In: *SIAM Journal on Mathematics of Data Science* 2.1, pp. 103–131.

Mnih, Volodymyr et al. (2015). "Human-level control through deep reinforcement learning". In: *nature* 518.7540, pp. 529–533.

Newey, Whitney K (1991). "Uniform convergence in probability and stochastic equicontinuity". In: *Econometrica: Journal of the Econometric Society*, pp. 1161–1167.

Palis Jr, J, W de Melo, et al. (1982). "Geometric Theory of Dynamical Systems". In:

Papadimitriou, Christos and Georgios Piliouras (2018). "From nash equilibria to chain recurrent sets: An algorithmic solution concept for game theory". In: *Entropy* 20.10, p. 782.

Plappert, Matthias et al. (2017). "Parameter space noise for exploration". In: *arXiv preprint arXiv:1706.01905*.

Ramaswamy, Arunselvan and Eyke Hullermeier (2021). "Deep Q-Learning: Theoretical Insights from an Asymptotic Analysis". In: *IEEE Transactions on Artificial Intelligence*.

Sayin, Muhammed et al. (2021). "Decentralized Q-learning in zero-sum Markov games". In: *Advances in Neural Information Processing Systems* 34.

Appendix A. Proofs

A.1. Proof of Lemma 1

Consider Definition 2, *(i)*.

Recall that Berge's Theorem of the maximum shows upper hemicontinuity of the argmax correspondence with respect to parameters. It allows for parameters living in function spaces equipped with the sup norm. To apply Berge, we rewrite $\tilde{Q}(s, a; \theta_n)$. First, define

$$\mathcal{Q} \equiv \Big\{ Q : S \times E \mapsto \mathbb{R} \Big| \text{ Q is twice differentiable in } E \text{ and } \sup_{(s,x)} \|Q(s,x)\| < \infty \Big\},$$

as the space of $Q-$ functions that can be generated by our game as defined in 6. Then define

$$\mathcal{B}_\sigma^0 = \left\{ \varepsilon : S \times A \mapsto \mathbb{R} \middle| \ \varepsilon \text{ is continuous in } E \times \Theta \text{ and } \sup_{(s,a)} \|\varepsilon(s,a)\| < \sigma \right\},$$

as the space of error functions resulting from the DQN estimation. Given some $\gamma_Q, \sigma > 0$ we can now define

$$F : \mathcal{Q} \times \mathcal{B}_{\gamma_Q}^2 \times \mathcal{B}_\sigma^0 \mapsto \mathcal{DQ},$$

such that

$$F(Q, g_q, \varepsilon) \equiv \tilde{Q}(s,a;\theta) = Q(s,a,x) + g_Q(s,a,x,\theta) + \varepsilon(s,a),$$

where we can then treat g_q, ε as parameters in Berge's Theorem. Thus, $\mathcal{T}F(Q, g_q, \varepsilon)$ is upper hemnicontinuous in g_q, ε. To connect back to our DQN estimation, write

$$F(Q(\cdot,\cdot,x_n^{-i}), g_q, \varepsilon_n) \equiv \tilde{Q}(s,a;\theta_n).$$

It follows that under Assumption 2 *(iii)*, for all $\gamma > 0$ there exists $\gamma_Q > 0$, $N > 0$ such that for all $g_Q \in \mathcal{B}_{\gamma_Q}^2$ and almost surely for all $n \geq N$,

$$H(\mathcal{T}Q(s,a,x_n), \mathcal{T}\tilde{Q}(s,a;\theta_n)) < \gamma,$$

and thus Definition 2, *(i)* holds.

Now for Definition 2, *(ii)*:

Fix n large and take $x_n \in \mathcal{U}$. By our assumptions, x_n is interior for each i. Fix an agent i. Since interior, x_n must solve the FOCs

$$D_a Q(s, x_n^i(s), x_n^{-i}) = 0 \ \forall \, s.$$

Now consider $\mathcal{T}[Q(s,s,x_n^{-i}) + g_Q(s,a,x_n^{-i},\theta_n)]$. Since $g_Q(s,a,x_n,\theta_n) \in \mathcal{B}_{\gamma_Q}^2$ we can apply arguments analogous to the proof of Proposition 2 to show that there is γ_Q small enough s.t. the perturbed argmax a^* must also be interior and solve

$$D_a Q(s, a^*, x_n^{-i}) + D_a g_Q(s, a^*, x_n^{-i}, \theta_n) = 0 \ \forall \, s.$$

Next, by Assumption 2 *(iv)*, we can apply the implicit function theorem to show that $a^*(x_n^{-i})$ is differentiable in x_n^{-i} in a neighborhood $U_{x_n^{-i}}$ of x_n^{-i}. Using the terminology of Definition 2, we can write

$$a^*(x_n^{-i}, \theta_n) = G(x_n^{-i}) + g(x_n^{-i}, \theta_n),$$

where $g(x_n^{-i}, \theta_n)$ is differentiable in x_n^{-i} and can be made to vanish as γ_Q vanishes, again by arguments analogous to the proof of Proposition 2. Finally, write

$$conv\left[\mathcal{T}\tilde{Q}(s,a;\theta_n)\right] = G(x_n^{-i}) + g(x_n^{-i}, \theta_n) + R_n^{B_n}(x_n^{-i}),$$

where $R_n^{B_n}(x_n^{-i})$ is a convex set resulting from the error term $\varepsilon_n(s,a)$ in the definition of $\tilde{Q}(s,a;\theta_n)$:

$$R_n^{B_n}(x_n^{-i}) = \left\{ z - \left[G(x_n^{-i}) + g(x_n^{-i}, \theta_n) \right] : \; z \in conv\left[\mathcal{T}\tilde{Q}(s,a;\theta_n) \right] \right\}.$$

Note that by upper hemicontinuity of the argmax and our assumption on vanishing $\varepsilon_n(s,a)$,

$$\sup_{\delta_n \in R_n(x_n^{-i})} \|\delta_n\| \to_p 0$$

as required.

One may wonder about the content of this as x_n may move in and out of the neighborhood $\mathcal{U}$. This statement then means that we can always find N large enough such that if $x_n \in \mathcal{U}$ for $n > N$, $R_n(x_n)$ can be made negligible.

■

A.2. Proof of Proposition 1

Write $\bar{M}_{n+1} = M_{n+1} + \varepsilon_n - g(x_n)$, then the algorithm 2 can be written as

$$x_{n+1} = x_n + \alpha_n \left[F_g(x_n) + \bar{M}_{n+1} + \delta_n \right],$$

were all assumption for the set valued convergence Theorem 3.6 in Benaım, Hofbauer, and Sorin 2005 hold. ■

A.3. Proof of Proposition 2

Since payoffs are differentiable around x^*, point 1 follows as long as x^g and x^* are close. For point 2, we will prove something more general: as long as x^* is hyperbolic, point 2 holds.

This follows because when x^* is hyperbolic, there is a neighborhood U around 0 such that F has a differentiable inverse on U. Next, note that x^g solves

$$F(x^g) + g(x^g) = 0.$$

Since $\|g\|_1 \leq \gamma$, for γ small enough, $F(x^g) \in U$ must hold. Then there is some $L_{F^{-1}} > 0$ such that

$$\begin{aligned} \|x^g - x^*\| &= \|F^{-1}(F(x^g)) - F^{-1}(0)\| \\ &\leq L_{F^{-1}} \|F(x^g)\| \leq L_{F^{-1}} \gamma, \end{aligned}$$

where the first inequality follows because F^{-1} is differentiable and $F(x^*) = 0$, and the second by the definition of $F(x^g)$. Since the right hand side is independent of g, the bound

is uniform.
For point 3, we first need to verify that all x^g close enough to x^* must also be asymptotically stable. The next Lemma gives a more general result:

Lemma 2. *Suppose x^* is hyperbolic. Then the eigenvalues of $DF_g(x^g)$ converge to the eigenvalues of $DF(x^*)$ uniformly over $g \in \mathcal{B}^1_\gamma$ as $\gamma \to 0$. Thus, for small enough γ, x^g has the same stability properties as x^*.*

Proof. We will show that eigenvalues of a hyperbolic matrix $DF(x^*)$ vary continuously in C^1 perturbations g to F.
Proposition 2.18 in Palis Jr, Melo, et al. 1982 shows that eigenvalues vary continuously for any matrix A. Thus, if $\|DF(x^*) - DF_g(x^g)\|$ is small enough, the eigenvalues of the two matrices must be close to each other. Now write

$$\begin{aligned}\|DF(x^*) - DF_g(x^g)\| &= \|DF(x^*) - DF(x^g)\| + \|Dg(x^g)\| \\ &\leq \|DF(x^*) - DF(x^g)\| + \gamma,\end{aligned}$$

where the equality follows from the definition of F_g. Since DF is continuous, and $x^g \to x^*$ uniformly for $g \in \mathcal{B}^1_\gamma$ as $\gamma \to 0$ (see above proof of point 2), we get that

$$\sup_{g \in \mathcal{B}^1_\gamma} \|DF(x^*) - DF_g(x^g)\| \to 0$$

as $\gamma \to 0$. Then applying Proposition 2.18 in Palis Jr, Melo, et al. 1982 finishes the result. □

Now that we know that all x^g must be asymptotically stable for γ small enough, we can apply Faure and Roth 2010 (Thm 2.8).
We only need to verify that our game satisfies their attainability condition:

Definition 8. *A point p is attainable if, for any $n > 0$ and any neighborhood U of p*

$$P[\exists s \geq n : x_s \in U] > 0.$$

We let $Att(X)$ be the set of attainable points for algorithm 2. Then we need that the basin of attraction of an attractor has nonempty intersection with $Att(X)$. This should be true given our support condition on M_{n+1} and the assumption that equilibria must be interior:

Lemma 3. *Let B be a basin of attraction of an attractor A for F_g. Suppose $x_n \in E \setminus B$. Then there exists $s > n$ such that $x_s \in B$ with positive probability.*

Proof. Since t is finite, to show existence we construct $s = n + 1$: For any $z \in B$, we can pin down the necessary shock M_z to reach it:

$$M_z = \frac{z - x_n}{\alpha_n} - F_g(x_n).$$

Since $z \in int(E)$ by definition, M_z is in the support of M_{n+1} for every n. For any ball B_z around z, we can define

$$\mathbf{M}_z = \{M_{x'} : x' \in B_z\}.$$

$\mathbf{M}_z$ must have positive measure for all finite n, since it is in the support of M_{n+1}. (if we allow $s > n + 1$, we may be able to increase the measure but we only need it to be positive.) □

All other conditions that are sufficient for the model-algorithm to converge to the attractor hold by definition 3.

■

A.4. Proof of Proposition 3

Notice first that the following analysis is local to the rest points in E_S, which by assumption on $\mathcal{U}$ is also where F, F_g are single valued. Solution curves are unique whenever they intersect $\mathcal{U}$.

The proof will use the Hartman-Grobman Theorem (c.f.Chicone 2006, Thm 4.8) , which connects the flow of a nonlinear ODE in the neighborhood of a hyperbolic rest point to the flow of a linearized ODE. Since it works fully locally, our analysis only requires that $F(x)$ be single valued and C^1 in U_{x^*}, and we can allow $F(x)$ to be multivalued otherwise.

First, we define invariant sets for given differential equations:

Definition 9. *Let $z(t, z_0)$ be the solution so some given differential equation $\dot{z} = f(z)$ with initial value z_0. Then a set S*

- *is invariant for f, if $z(t, z_0) \in S$ holds for all $t \in \mathbb{R}$ and all $z_0 \in S$.*
- *isolated invariant for f if there is an open set N such that $S \subset N$ and*

$$S = \{z' : z(t, z') \in N \, \forall t \in \mathbb{R}\}.$$

Given a $g \in \mathcal{B}^1_\gamma$, we know from Proposition 1 that only ICT sets subset of a neighborhood of x^g are candidates to being limiting points of the algorithm 2. The singleton $\{x^g\}$ is an ICT set, and we show first that this cannot be a limiting set of the algorithm. Then we go on to show that for small engough γ, no other ICT sets can exists in a neighborhood around x^*, which finishes the proof.

1) $\{x^g\}$ cannot be a limiting set.

Note that by Lemma 2, there are $\gamma > 0$ small enough such that all x^g are linearly unstable just as x^*. We can thus apply Benaım and Faure 2012, Thm 3.12 to prove $P[L_g = x^g] = 0$ first:
We can show that the sufficient conditions for this hold by definition of our algorithm 3. According to Faure and Roth 2010 Proposition 2.16, we have that the bounding function required in Benaım and Faure 2012, Hypothesis 2.2 exists given our assumptions on ε_n, M_n. Benaım and Faure 2012's Hypothesis 3.6 is then also satisfied, at least in a neighborhood of the rest point. As noted by their Remark 3.7, all conditions only need to hold in a neighborhood of the unstable point, so set-valued gradients outside the neighborhood are allowed.

2) No other ICT sets exist in a neighborhood of x^* and x^g.

We will prove that there are no other invariant sets in such a neighborhood. Since ICT sets are subsets of invariant sets, this will complete the proof.
We can use Hartman-Grobman to show that there are open neighborhoods N_g, N_0 with $x^* \in N_0, x^g \in N_g$ such that x^*, x^g are isolated invariant sets in their respective neighborhoods. These neighborhoods are nontrivial for all γ small enough, which follows from both x^*, x^g being hyperbolic:
By Hartman-Grobman and hyperbolicity there exists a homeomorphism H on a neighborhood $N \subseteq U_{x^*}$ of x^* with $H(x^*) = x^*$ such that

$$H(\phi(t, x)) = \psi(t, H(x)),$$

where $\phi(t, \cdot)$ is a solution (flow) to the differential inclusion $\dot{x} \in conv[F(x)]$, and $\psi(t, \cdot)$ is the solution to the ODE $\dot{y} = DF(x^*)(y - x^*)$. Given a neighborhood $U \subseteq N$ of x^*, define

$$inv(U) = \{x \in U : \phi(t, x) \in U \, \forall t \in \mathbb{R}\}.$$

We will show that $x^* = inv(U)$, and therefore it is isolated invariant.
Notice that $inv(U)$ can be rewritten as

$$inv(U) = \{y \in H(U) : H^{-1}(\psi(t, y)) \in U \, \forall t \in \mathbb{R}\} = \{y \in H(U) : \psi(t, y) \in H(U) \, \forall t \in \mathbb{R}\},$$

since H is bijective. We know that x^* is an isolated invariant set for the linear ODE solution $\psi(t, y) = Ce^{tDF(x^*)}y + x^*$. Thus, we must also have that

$$inv(U) = x^*,$$

and x^* is isolated invariant set for $\phi(t, x)$.
Since x^g are hyperbolic for γ small enough, an analogous argument gives us that x^g are isolated invariant also. Let N_g be the neighborhood on which the homeomorphism is defined

that connects flows of F_g to flows of the linearized system $DF_g(x^g)$. By definition, $x^g \in N_g$, and we know that x^g is isolated invariant in N_g. We are left to show that for γ small enough, for all $g \in \mathcal{B}^1_\gamma$, $x^* \in N_g$:
To prove this, we will argue that each N_g contains a ball $B^g_z(x^g)$, for which the radius $z > 0$ can be lower bounded by a number that depends only on the eigenvalues of $DF(x^*)$ and γ. First we need an auxiliary Lemma to show how eigenvalues of $DF_g(x^g)$ vary continuously in γ. First some more notation:
For small enough γ, all x^g are hyperbolic when $g \in \mathcal{B}^1_\gamma$. Fix such a g. Define $\rho_l > 0$ to be the smallest positive eigenvalue of $DF_g(x^g)$, and $\rho_u < 0$ be the largest negative eigenvalue of $DF_g(x^g)$. Now let $a_g \in (0,1)$ be any number such that

$$\max\{e^{\rho_u}, e^{-\rho_l}\} < a_g < 1.$$

For the original system $DF(x^*)$, let $a_0 \in (0,1)$ be any such number.

Lemma 4. *For any $\delta > 0$ with $a_0 < 1 - \delta$ there exists $\overline{\gamma} > 0$ such that for all $\gamma \in (0, \overline{\gamma}]$, there is a set of $\{a_g\}_{g \in \mathcal{B}^1_\gamma}$ as defined above with*

$$\sup_{g \in \mathcal{B}^1_\gamma} |a_g - a_0| < \delta.$$

Proof. Apply Lemma 2. Since there is a one-to-one mapping between eigenvalues and $\{e^{\rho_u}, e^{-\rho_l}\}$, we can find numbers a_g. The result follows. □

Given this continuity in eigenvalues, we can prove the following Lemma to finish our result:

Lemma 5. *Suppose x^* is hyperbolic for F. Fix a small $\underline{z} > 0$. Then there is $\bar{\gamma}$ such that for all $\gamma \leq \bar{\gamma}$, and all $g \in \mathcal{B}^1_\gamma$, there is $B^g_z(x^g) \subseteq N_g$ with $z \geq \underline{z}$.*

Proof. For small enough γ, all x^g are hyperbolic when $g \in \mathcal{B}^1_\gamma$. Fix such a g. Given some $\varepsilon > 0$, let r_ε be defined as

$$\sup\{r > 0 : \|x - x^g\| < r;\ \|DF_g(x) - DF_g(x^g)\| < \varepsilon\}.$$

Since DF_g is continuous, $r_\varepsilon > 0$ must hold. Pick $a_g \in (0,1)$ as defined previously.
Then define

$$\overline{\varepsilon}_g = \frac{1 - a_g}{a_g} > 0.$$

By Lemmas 4.3 and 4.4 of Palis Jr, Melo, et al. 1982, $B_{r_\varepsilon}(x^g) \subseteq N_g$, if $\varepsilon < \overline{\varepsilon}_g$.
We are left to show that r_ε can be made to depend only on the eigenvalues of $DF(x^*)$ and γ.

Notice that small enough $\underline{z} > 0$ pins down the $\delta > 0$ referred to in Lemma 4: Let

$$\hat{z}(\overline{\gamma}) = \inf_{\gamma \in (0,\overline{\gamma}]} \inf_{g \in \mathcal{B}^1_\gamma} \overline{\varepsilon}_g.$$

For $\delta > 0$ small enough, choose $\overline{\gamma} > 0$ such that Lemma 4 holds. It follows from the Lemma that $\hat{z}(\overline{\gamma}) > 0$. Then any $\underline{z} < \hat{z}(\overline{\gamma})$ satisfies our conditions and the conclusion follows. □

Now recall that by the proof of Proposition 2 point 2, $x^g \to x^*$ uniformly over $g \in \mathcal{B}^1_\gamma$ as $\gamma \to 0$. Thus there is γ small enough for which $\sup_{g \in \mathcal{B}^1_\gamma} |x^g - x^*| < \underline{z}$ and therefore $x^* \in N_g$ for all $g \in \mathcal{B}^1_\gamma$. Let $U_\gamma = \cap_{g \in \mathcal{B}^1_\gamma} N_g$. Since x^g for $g \in \mathcal{B}^1_\gamma$ are isolated invariant in U_γ by construction, the result follows.

■

Appendix B. Example of a C^1 Approximation Algorithm

We show here an example of a synchronous actor-critic Q learning scheme that achieves our sufficient conditions 2. Since our sufficient conditions concern the asymptotic behavior, we will consider a simple, naive scheme with the required asymptotic properties, that may otherwise be inefficient and not too desirable. We argue that if this simple scheme satisfies the sufficient conditions, since the conditions are desirable for an algorithm designer it is likely that more realistic schemes will do better at achieving our bounds.

Take a finite set of agents I with $|I| = N$, fix a finite state space S with $|S| = K$ and continuous action set A as defined in 6. As in the previous section, we let $Q^i(s, a, x^{-i})$ be the unique Q value function given an opponent policy profile x^{-i}.

Recall that, for $a \in A$, we write $u^i(a, x^{-i})$ as the stage game payoff given an opponent profile, and

$$P_{ss'}(x) = Pr[s'|s, x]$$

as the transition probability function from state s to s' given a profile x. We will assume that all agents use parametric function approximation to estimate their payoff and transition functions. Take compact, finite dimensional parameter space Θ and $0 < j < h < 1$ and define

$$\mathcal{F}_u = \left\{ f_u(a; \theta) \mapsto \mathbb{R} \right\} \subset \mathcal{C}^2[A \times \Theta, \mathbb{R}]; \quad \mathcal{F}_p = \left\{ f_p(a; \eta) \mapsto [j, h] \right\} \subset \mathcal{C}^2[A \times \Theta, [j, h]],$$

as the space of functions used by all agents to approximate their respective payoffs and transition functions. Note that we impose transition function approximators to map into numbers strictly between $0, 1$. This will simplify the convergence proof and not make us loose generality (for small enough j and large enough h) by the assumption of irreducible state distributions (see Definition 6).

Before speaking about the function approximation in detail, we will describe how agents update their policies, choose actions and make observations. From now on, to save notation we drop the i superscripts.

- **Parameter estimates**
  Each agent i keeps track of their current period's parameter estimates θ_t, η_t that pin down their payoff and transition function estimators respectively.
- **$\tilde{Q}$ as a fixed point**
  Agents determine their Q estimates as the fixed point of

$$\tilde{Q}^i(s,a;\theta,\eta) = f_u(a;\theta_{s,t}) + \delta \sum_{s' \in S} f_p(a;\eta_{ss',t}) \mathcal{T} \tilde{Q}^i(s',a';\theta,\eta). \tag{5}$$

  Recall that $\mathcal{T} = \max_{a' \in A}$. Here we write $\theta_{s,t}, \eta_{ss',t}$ to make clear the dependence of the approximating parameters on the given states.
  Since for each s, a, $\sum_{s' \in S} f_p(a;\eta_{ss',t}) = 1$, it is quick to check that 5 is a contraction mapping and therefore has a unique fixed point.

- **Policy updates**
  We assume agents update their policies $x_t \in \bar{A}$ according to

$$x_{t+1}(s) = x_t(s) + \alpha_t \left[\mathcal{T} \tilde{Q}(s,a;\theta_t,\eta_t) - x_t + M_{t+1} \right], \tag{6}$$

  for all $s \in S$, where we take our assumptions in Definition 3 to hold for all agents, replacing the algorithm operator in Definition 3 by our parametric estimator $\mathcal{T} \tilde{Q}(s,a;\theta_t,\eta_t)$.

- **Action sampling**
  To simplify our analysis, we assume that given current state s_t, agents sample their actions at every period with an ε- greedy policy:

$$a_t \sim \bar{x}_t(s_t),$$

  where $\bar{x}_t$ is a mixture such that with probability $1-\varepsilon$, x_t is chosen, and with ε, actions are sampled from A according to a continuous full support distribution with fixed mean μ and density function $g(a) > 0 \forall a \in A$. Note that $\varepsilon, g(a)$ can vary by individuals without loss, but for ease of notation we will assume them symmetric here. What is important is that over time, only the mean x_t changes, while the exploration mixture g is held fixed. For such mixtures and for any θ, we write

$$U(\bar{x}_t(s)) = \mathbb{E}[u(a_t) | \, \bar{x}_t, s]; \quad F_u(\bar{x}^i_t(s);\theta) = \mathbb{E}[f_u(a^i_t;\theta) | \, \bar{x}^i_t, s]$$

The following describes the period-by-period behavior of an agent:

(1) Agents observe s_t.
(2) Agents sample $a_t^i \sim \bar{x}_t^i(s_t)$, observing their own action only.
(3) Agents observe their own stage payoff u_t^i and next state s_{t+1}.
(4) Agents update their parameter vectors θ_t, η_t.
(5) Agents compute new $\tilde{Q}$ fixed points based on those parameters and 5.
(6) Agents use 6 to update policies.

Thus, the new data that is observed every period by an agent is (s_t, a_t, u_t, s_{t+1}).
Now we are ready to define the parametric function estimation procedure.
For each $s \in S$, let the state-count function be

$$n_T(s) = \sum_{t=1}^{T} \mathbf{1}\{s_t = s\}.$$

By our assumptions on the irreducibility of the state-Markov chain conditional on any action profile $a \in A$ (see Definition 6), we have that there is $c \in (0,1)$ and $\bar{T}$ such that for all s,

$$\frac{n_T(s)}{T} > c \tag{7}$$

almost surely for all $T > \bar{T}$.
For sequences of natural numbers $\bar{B}_T(s) < T$, $\bar{B}_T(ss') < T$ that diverge,

$$W_T(s) = \left\{T - \bar{B}_T(s) + 1 \leq t \leq T |\, s_t = s\right\};$$
$$W_T(ss') = \left\{T - B_T(ss') + 1 \leq t \leq T |\, s_t = s, s_{t+1} = s'\right\},$$

as moving windows of time indices at which a certain state was observed, or a certain state transition was observed. These will be the sets of time indices used to estimate state-dependent payoffs and transitions for each agent.
Let $B_T(s) = |W_T(s)|$ be the random size of $W_T(s)$. We will use a carefully constructed rate of divergence for $B_T(s)$ to ensure convergence of our estimations. If $B_T(s)$ is a strictly monotone function of $n_T(s)$, this is indeed feasible by 7.
Define

$$\hat{L}_T^s(\theta) = \frac{1}{B_T(s)} \sum_{t \in W_T(s)} \left(u_t - f_u(a_t; \theta)\right)^2, \tag{8}$$

as the sample criterion function used to estimate θ_t for all s:

$$\theta_{s,T} = argmin_{\theta \in \Theta} \hat{L}_T^s(\theta).$$

Thus, $\theta_{s,t}$ is an M-estimator as in Hansen 2010, Chapter 6. Given a fixed profile $x \in E$ and a sequence of profiles $\mathbf{x}(W_T(s)) = \{x_t\}_{t \in W_T(s)}$, define the population criterion functions

$$L^s(\theta, x) = \mathbb{E}\big[\big(u_t - f_u(a_t;\theta)\big)^2 \big|\, \bar{x}, s\big];$$
$$L^s(\theta, \mathbf{x}(W_T(s))) = \frac{1}{B_T(s)} \sum_{t \in W_T(s)} \mathbb{E}\big[\big(u_t - f_u(a_t;\theta)\big)^2 \big|\, \bar{x}_t, s\big].$$

We assume:

Assumption 3 (θ-ID)**.**
For all $x \in E$ and all $s \in S$ there exists a unique $\theta^*(x)$ that minimizes $L^s(\theta, x)$, in the sense that for all $\rho > 0$,

$$\inf_{\theta \notin B_\rho(\theta^*(x))} L^s(\theta, x) > L^s(\theta^*(x), x).$$

With this assumption at hand we are ready to prove consistency in θ_t:

Lemma 6. *Suppose assumption 3 holds and consider the algorithm 6. Then θ_T as the minimizer of 8 satisfies*

$$\|\theta_T - \theta^*(x_T)\| \to_p 0,$$

as $T \to \infty$.

Proof. We first show that for all $\theta \in \Theta$,

$$\|\hat{L}^s_T(\theta) - L^s(\theta, x_T)\| \to_p 0, \tag{9}$$

as $T \to \infty$. We can write

$$\|\hat{L}^s_T(\theta) - L^s(\theta, x_T)\| \leq \|\hat{L}^s_T(\theta) - L^s(\theta, \mathbf{x}(W_T(s)))\| + \|L^s(\theta, \mathbf{x}(W_T(s)) - L^s(\theta, x_T)\|.$$

First, letting $Y_t = \big(u_t - f_u(a_t;\theta)\big)^2$ we can write

$$\hat{L}^s_T(\theta) - L^s(\theta, \mathbf{x}(W_T(s))) = \frac{1}{B_T(s)} \sum_{t \in W_T(s)} Y_t - \mathbb{E}\big[Y_t \big|\, x_t, s\big].$$

Now note that conditional on x_t, s, actions are sampled independently across individuals and time. In fact, states s follow a controlled markov process, where in this case the control profile x_t is markov with respect to states also, only that it is changing over time. Thus over t, Y_t conditional on x_t, s are not identically distributed, but independent. We are left to show that

$$\mathbb{E}[Y_t^2 |\, \bar{x}, s] < \infty$$

holds for all t and uniformly over θ, x, s. This follows promptly from the uniform boundedness of u, f_u. Now we can use Chebyshev's inequality, and it follows that

$$\|\hat{L}^s_T(\theta) - L^s(\theta, \mathbf{x}(W_T(s)))\| \to_p 0$$

as $T \to \infty$. Then write

$$\|L^s(\theta, \mathbf{x}(W_T(s)) - L^s(\theta, x_T)\| \leq \frac{1}{B_T(s)} \sum_{t \in W_T(s)} \|\mathbb{E}\big[Y_t\big|\, \bar{x}_t, s\big] - \mathbb{E}\big[Y_t\big|\, \bar{x}_T, s\big]\|$$
$$\leq C_0 R_{1,T} + C_0 R_{2,T},$$

with

$$R_{1,T} = \frac{1}{B_T(s)} \sum_{t \in W_T(s)} \|U(\bar{x}_t(s)) - U(\bar{x}_T(s))\|,$$

and

$$R_{2,T} = \frac{1}{B_T(s)} \sum_{t \in W_T(s)} \|F_u(\bar{x}^i_T(s); \theta) - F_u(\bar{x}^i_t(s); \theta)\big]\|. \tag{10}$$

The last inequality follows from bounded payoffs, so that the expectation of the squared difference has Lipschitz constant $C_0 < \infty$.

Recall that given some profile $\bar{x}$, we can equivalently state each agent's action sampling the following way: first each agent tosses a biased coin $C_i \in \{0, 1\}$, such that $Pr[C_i = 0] = 1-\varepsilon$. In other words, $C_i = 0$ means the agent plays x_i, while otherwise they sample actions from A according to pdf $g(a)$. To find $U(\bar{x}_t)$, we can thus define all possible outcomes of all agents draws from C_i:

$$\Sigma = \{\{c_i\}_{i=1}^N |\, c_i \in \{0, 1\}\}.$$

Then for any $\sigma \in \Sigma$

$$Pr[\sigma] = \Pi_i \varepsilon^{(1-c_i)}(1-\varepsilon)^{c_i}.$$

Given a draw of σ, let $a \in A$ be an action profile in the support of $\bar{x}$. We write $a(\sigma(x))$ as the subvector of a that collects all actions taken by agents whose coin toss landed on $\{0\}$:

$$a(\sigma(x)) = (x_i)_{i:\, c_i = 0}.$$

Analogously we define $a(\sigma(g))$ as the remaining coordinates of a - precisely those for which the draw σ specifies exploration. We let $I(\sigma(g))$ $k(\sigma) = |I(\sigma(g))|$ be the set and number of agents drawing $c_i = 1$ in profile σ. Let z be the function that maps the reordered profile $(a(\sigma(x)), a(\sigma(g)))$ back to a. We can thus write $u(a) = u(z(a(\sigma(x)), a(\sigma(g))))$.

Using this notation, we can write the expected payoff as

$$U(\bar{x}) = \sum_{\sigma \in \Sigma} Pr[\sigma] \int_{A_i | i \in I(\sigma(g))} u(z(a(\sigma(x)), a(\sigma(g))))g(a(\sigma(g)))^k,$$

where we make use of our assumption that all agents use the same exploration density g. If that were not the case, the proof is analogous but more notation heavy. Now we are able to use the Lipschitz continuity of u, f_u to finish the proof:

$$R_{1,T} \leq C_1 \frac{1}{B_T(s)} \sum_{t \in W_T(s)} \|x_t(s) - x_T(s)\| \leq C_1 C_2 \frac{1}{B_T(s)} \sum_{t \in W_T(s)} \sum_{l=t}^{T} \alpha_l,$$

where the first inequality uses the $C_1 < \infty$, the Lipschitz constant of u and the fact that between $\bar{x}_t, \bar{x}_T$, the only difference is the underlying policy x_t, x_T, such that there is no expectation involved anylonger. The second inequality takes $C_2 < \infty$ such that

$$\sup_{a \in A} \|a - \underline{a} + M\| < C_2,$$

where $\underline{a} = \inf_{a' \in A} a'$ and M is the random variable as in 6, which is assumed to be bounded almost surely (see Definition 3). In other words, C_2 is an upper bound to the updating factor for a policy at each period. Since α_t is decreasing, we have

$$\frac{1}{B_T(s)} \sum_{t \in W_T(s)} \sum_{l=t}^{T} \alpha_l \leq B_T(s) \alpha_{T - B_T(s)}.$$

We will show that for some α_t that satisfy the classical Robbins-Monro condition stated in Definition 3, the right hand side can be made to vanish. Take $b \in (\frac{1}{2}, 1]$, and let $\alpha_t = t^{-b}$. Then for any $v \in (0, b)$, the result follows if we let $B_T(s) = n_T(s)^v$:
By 7,

$$B_T(s) \alpha_{T - B_T(s)} = n_T(s)^v \big(T - n_T(s)^v\big)^{-b} = n_T(s)^{v-b} \Big(\frac{T}{n_T(s)^b} - n_T(s)^{v-b}\Big)^{-b}$$
$$n_T(s)^{v-b} \Big(T^{1-b} \big(\frac{T}{n_T(s)}\big)^b - n_T(s)^{v-b}\Big)^{-b} \to 0$$

almost surely as $T \to \infty$, since $n_T(s)^{v-b} \to 0$ and by 7 $\frac{T}{n_T(s)} > 0$ almost surely. An analogous argument holds for $R_{2,T}$. The claim follows: 9 holds.
To conclude we will apply well known results from econometric theory for M- estimators. We assume compact Θ, and by the above proof of pointwise convergence, we have that Assumptions 1 and 2 in Newey 1991 are satisfied. To conclude, is is sufficient to show that

$$H_T^s(\theta) = \hat{L}_T^s(\theta) - L^s(\theta, \mathbf{x}(W_T(s))) \tag{11}$$

is stochastically equicontinuous. This holds if $H_T^s(\theta)$ has a uniformly bounded derivative in θ for all T large enough (see the remark after Corollary 2.2 in Newey 1991). In our case this is satisfied since we assume u, f_u to be bounded and twice differentiable. Corollary 2.2 in Newey 1991 then gives that $H_T^s(\theta)$ converges to zero uniformly over θ. The result now follows from Theorem 22.1 in Hansen 2010. □

For η_T we proceed similarly, but using maximum likelihood estimation. For any parameter η, given a state transition s let the parametrized likelihood of a the random next state

$\bar{s}$ conditional on action profile a and state s be defined as

$$\Lambda(\bar{s}|s,a,\eta) = \Pi_{s' \in S} f_p(a;\eta_{ss'})^{\mathbf{1}\{s'=\bar{s}\}}. \tag{12}$$

From here we can define the observed loglikelihood function

$$\hat{l}_T^s(\eta) = -\frac{1}{B_T(s)} \sum_{t \in W_T(s)} \sum_{s' \in S} \mathbf{1}\{s_{t+1} = s'\} log f_p(a_t;\eta_{ss'}). \tag{13}$$

Similarly, and as in the case of θ, we have the population versions:

$$l^s(\eta,x) = -\sum_{s' \in S} \mathbb{E}\big[\mathbf{1}\{s_{t+1} = s'\} log f_p(a_t;\eta_{ss'}) \big| \bar{x}, s\big];$$

$$l_T^s(\eta, \mathbf{x}(W_T(s))) = -\frac{1}{B_T(s)} \sum_{t \in W_T(s)} \sum_{s' \in S} \mathbb{E}\big[\mathbf{1}\{s_{t+1} = s'\} log f_p(a_t;\eta_{ss'}) \big| \bar{x}_t, s\big].$$

Then, define

$$\eta_T = argmin_{\eta \in \Theta} \hat{l}_T^s(\eta); \quad \eta^*(x) = argmin_{\eta \in \Theta} l^s(\eta, x).$$

We assume:

Assumption 4 (η-ID)**.**
For all $x \in E$ and all $s \in S$ there exists a unique $\eta^*(x)$ that minimizes $l^s(\eta, x)$, in the sense that for all $\rho > 0$,

$$\inf_{\eta \notin B_\rho(\eta^*(x))} l^s(\eta, x) > l^s(\eta^*(x), x).$$

Then,

Lemma 7. *Suppose assumption 4 holds and consider the algorithm 6. Then η_T as the minimizer of 13 satisfies*

$$\|\eta_T - \eta^*(x_T)\| \to_p 0,$$

as $T \to \infty$.

Proof. The proof continues analogously to the proof of Lemma 6. We will need to show the pointwise consistency of $\hat{l}_T^s(\eta)$, and the rest will follow from our condition on the speed of $B_T(s)$ and x_T. We first show that for all $\eta \in \Theta$,

$$\|\hat{l}_T^s(\eta) - l^s(\eta, x_T)\| \to_p 0, \tag{14}$$

as $T \to \infty$. We can write

$$\|\hat{l}_T^s(\eta) - l^s(\eta, x_T)\| \leq \|\hat{l}_T^s(\eta) - l^s(\eta, \mathbf{x}(W_T(s)))\| + \|l^s(\eta, \mathbf{x}(W_T(s)) - l^s(\eta, x_T)\|.$$

By analogous arguments to Lemma 6, we get that the first term converges to 0 in probability as $T \to \infty$. As for the second term,

$$\|l^s(\eta, \mathbf{x}(W_T(s)) - l^s(\eta, x_T)\|$$
$$\leq \frac{1}{B_T(s)} \sum_{t \in W_T(s)} \sum_{s' \in S} \left\| \mathbb{E}\big[\mathbf{1}\{s_{t+1} = s'\} log f_p(a_t; \eta_{ss'}) \big| \bar{x}_t, s\big] - \mathbb{E}\big[\mathbf{1}\{s_{t+1} = s'\} log f_p(a_t; \eta_{ss'}) \big| \bar{x}_T, s\big] \right\|$$
$$= \frac{1}{B_T(s)} \sum_{t \in W_T(s)} \sum_{s' \in S} \left\| \mathbb{E}\big[P_{ss'}(a_t) log f_p(a_t; \eta_{ss'}) \big| \bar{x}_t, s\big] - \mathbb{E}\big[P_{ss'}(a_t) log f_p(a_t; \eta_{ss'}) \big| \bar{x}_T, s\big] \right\|,$$

where the equality comes from an application of the law of iterated expectations (by conditioning on $a_t \bar{x}_t, s$). Having this, we can use analogous arguments to Lemma 6 to write out the expectation operators and then bound their differences, by using differentiability and boundedness of $P_{ss'}(a_t) log f_p(a_t; \eta_{ss'})$ for all ss', η. The conclusion then follows as in the previous Lemma, and 14 holds. Finally we can again refer to Lemma 6 for an analogous verification that all assumptions for uniform weak convergence hold, and the conclusion follows. □

We have shown in Lemma 7 that η_T converges to the unique minimizer $\eta^*(x_T)$. One may wonder about the interpretation of this result in the case where the true transition function $P_{ss'}(a) \notin \mathcal{F}_p$. It is a classical result in econometric theory that $\eta^*(x)$ can be seen as minimizer of the Kulback-Leibler Information Criterion, which measures distances between two measures P, f_p. In other words, η^* minimizes the distance between the true transition function $P_{ss'}$ and the parametric family $\mathcal{F}_p$. For details, see for example Hansen 2010, Chapter 28.

Given these results, we can write $\tilde{Q}(s, a; \theta_t, \eta_t)$ as we wanted:

Proposition 4. *Under assumptions 3, 4 and the assumptions specified for algorithm 6, we can write*

$$\tilde{Q}(s, a; \theta_t, \eta_t) = \tilde{Q}(s, a; \theta^*(x_t), \eta^*(x_t)) + \zeta_t,$$

where $\zeta_t \to_p 0$ *as* $t \to \infty$. *Furthermore,*

$$\bar{g}(s, a, x_t) = Q(s, a, x_t) - \tilde{Q}(s, a; \theta^*(x_t), \eta^*(x_t))$$

is the asymptotic bias term, and is twice differentiable in a, x_t.

Proof. We can write

$$\sup_{s,a} \|\tilde{Q}(s,a;\theta_t,\eta_t) - \tilde{Q}(s,a;\theta^*(x_t),\eta^*(x_t))\|$$

$$\leq \sup_{s,a} \|f_u(a;\theta_{s,t}) - f_u(a;\theta^*_s(x_t))\|$$

$$+\delta \sum_{s'\in S} \sup_{s,a} \left\| f_p(a;\eta_{ss',t})\mathcal{T}\tilde{Q}(s',a';\theta_t,\eta_t) - f_p(a;\eta^*_{ss'}(x_t))\mathcal{T}\tilde{Q}(s',a';\theta^*(x_t),\eta^*(x_t))\right\|$$

$$\leq D_1\|\theta_t - \theta^*(x_t)\| + \delta D_2 \sup_{s,a} \sum_{s'\in S} \left\| f_p(a;\eta_{ss',t}) - f_p(a;\eta^*_{ss'}(x_t))\right\|$$

$$+\delta \sup_{s,a} \sum_{s'\in S} f_p(a;\eta^*_{ss'}(x_t))\left\|\mathcal{T}\tilde{Q}(s',a';\theta_t,\eta_t) - \mathcal{T}\tilde{Q}(s',a';\theta^*(x_t),\eta^*(x_t))\right\|$$

$$\leq D_1\|\theta_t - \theta^*(x_t)\| + \delta D_2 D_3 K\left\|\eta_t - \eta^*(x_t)\right\| + \delta \sup_{s,a} \|\tilde{Q}(s,a;\theta_t,\eta_t) - \tilde{Q}(s,a;\theta^*(x_t),\eta^*(x_t))\|,$$

where $D_1 < \infty$ is the upper bound on the Lipschitz constant of f_u, $D_2 < \infty$ is the upper bound on $\sup_{s,a,\theta,\eta} \|\tilde{Q}\|$, $D_3 < \infty$ is the upper bound on the Lipschitz constant of f_p, and the last inequality follows because $\sum_{s'\in S} f_p = 1$ by construction, and the maximum of a difference dominates the difference of maxima. Putting the last line and the first line together, we get

$$\sup_{s,a} \|\tilde{Q}(s,a;\theta_t,\eta_t) - \tilde{Q}(s,a;\theta^*(x_t),\eta^*(x_t))\|$$

$$\leq \frac{1}{1-\delta}\left[D_1\|\theta_t - \theta^*(x_t)\| + \delta D_2 D_3 K\left\|\eta_t - \eta^*(x_t)\right\|\right],$$

and the first assertion follows by Lemmas 6, 7.
The second assertion follows once prove that the minimizers θ^*, η^* are differentiable in x_t. This follows by strengthening assumptions 3, 4 slightly, by requiring that the Hessian of the minimization problem at the solution be negative definite, for all profiles x_t. □

Proposition 4 shows that for the algorithm (6) here developed, our sufficient condition outlined in Assumption 2 holds.