STRATEGY SYNTHESIS FOR ZERO-SUM NEURO-SYMBOLIC CONCURRENT STOCHASTIC GAMES[*]

RUI YAN[†], GABRIEL SANTOS[†], GETHIN NORMAN[‡], DAVID PARKER[§], AND MARTA KWIATKOWSKA[†]

Abstract. Neuro-symbolic approaches to artificial intelligence, which combine neural networks with classical symbolic techniques, are growing in prominence, necessitating formal approaches to reason about their correctness. We propose a novel modelling formalism called neuro-symbolic concurrent stochastic games (NS-CSGs), which comprise probabilistic finite-state agents interacting in a shared continuous-state environment, observed through perception mechanisms implemented as neural networks. Since the environment state space is continuous, we focus on the class of NS-CSGs with Borel state spaces. We consider the problem of zero-sum discounted cumulative rewards and prove the existence of the value of NS-CSGs under Borel measurability and piecewise-constant restrictions on the components of the model. From an algorithmic perspective, existing methods to compute values and optimal strategies for CSGs focus on finite state spaces. We present, for the first time, implementable value iteration and policy iteration algorithms to solve a class of uncountable state space CSGs, namely NS-CSGs, and prove their convergence. Our approach works by exploiting the underlying game structures and then formulating piecewise linear or constant representations of the value functions and strategies of NS-CSGs. We illustrate our approach by applying a prototype implementation of value iteration to a dynamic vehicle parking case study.

Key words. Stochastic games, neuro-symbolic systems, value iteration, policy iteration, Borel state spaces

MSC codes. 91A15, 92B20, 60J05, 93E20

1. Introduction.

Game theory offers an attractive framework for analysing strategic interactions among agents in machine learning, with application to, for instance, the game of Go [30], autonomous driving [28] and robotics [10]. An important class of dynamic games is *stochastic games* [29], which move between states according to transition probabilities controlled jointly by multiple agents (players). Extending both strategic-form games to dynamic environments and Markov decision processes to multiple players, stochastic games have long been used to model sequential decision-making problems with more than one agent, ranging from multi-agent reinforcement learning [34], to quantitative verification and synthesis for equilibria [16].

Recent years have witnessed encouraging advances in the use of neural networks (NNs) to approximate either value functions or strategies [17] for stochastic games that model large, complex environments. Such *end-to-end* NNs directly map environment states to Q-values or actions. This means that they have a relatively complex structure and a large number of weights and biases, since they interweave multiple tasks (e.g., object detection and recognition, decision making) within a single NN. An emerging trend in autonomous and robotic systems is *neuro-symbolic* approaches, where some components that are synthesized from data (e.g., perception modules) are implemented as NNs, while others (e.g., nonlinear controllers) are formulated us-

[*] Submitted to the editors DATE.

Funding: This project was funded by the ERC under the European Union's Horizon 2020 research and innovation programme (FUN2MODEL, grant agreement No. 834115).

[†] Department of Computer Science, University of Oxford, Oxford, OX1 2JD, UK (rui.yan@cs.ox.ac.uk, gabriel.santos@cs.ox.ac.uk, marta.kwiatkowska@cs.ox.ac.uk).

[‡] School of Computing Science, University of Glasgow, University Avenue, Glasgow, G12 8QQ, UK (gethin.norman@glasgow.ac.uk)

[§] School of Computer Science, University of Birmingham, Edgbaston, Birmingham, B15 2TT, UK (d.a.parker@cs.bham.ac.uk)

ing traditional symbolic methods. This can greatly simplify the design and training process, and yield smaller NNs.

However, there remains a lack of modelling and verification frameworks which can reason formally about the correctness of neuro-symbolic systems. Progress has been made on techniques for both multi-agent verification [1, 24] and safe reinforcement learning [2] in this context, but without the ability to reason formally about stochasticity, which is crucial for modelling uncertainty. Elsewhere, concurrent stochastic games (CSGs) have been widely studied [32, 31, 9, 22], but primarily in the context of finite state spaces, which are insufficient for many real-life systems; for instance, uncountable real vector spaces are usually supplied as inputs to NNs.

Stochastic games were introduced by Shapley [29], who assumed a finite state space. Since then, many researchers, have considered CSGs with uncountable state spaces, e.g., [15, 18, 20]. Maitra and Parthasarathy [18] were the first to study discounted zero-sum CSGs in this setting, assuming that the state space is a compact metric space. Following this, more general results for discounted zero-sum CSGs with Borel state spaces have been derived, e.g., [15, 19, 20, 11]. These aim at providing sufficient conditions for the *existence* of either values or optimal strategies for players.

Another important and practical problem for zero-sum CSGs with uncountable state spaces is the *computation* of values and optimal strategies. Since the seminal policy iteration (PI) methods were introduced by Hoffman and Karp [12], and by Pollatschek and Avi-Itzhak [23], a wide range of fixed-point algorithms have been developed for zero-sum CSGs with finite state spaces [32, 31, 9, 22]. Recent work by Bertsekas [5] proposed a distributed optimistic abstract PI algorithm, which inherits the attractive structure of the Pollatschek and Avi-Itzhak algorithm while resolving the convergence difficulties suffered by the latter. All these assume finite state spaces. However, to the best of our knowledge, there are no existing value iteration (VI) or PI algorithms for CSGs with uncountable, or more specifically Borel, state spaces. VI and PI algorithms for stochastic control (i.e., the one player case) with Borel state spaces can be found in [37, 36]. Other problems for zero-sum CSGs with uncountable state spaces have been studied and include information structure [13], specialized strategy spaces [4], continuous time setup [8] and payoff criteria [11].

As discussed later, this paper assumes a *fully observable* game setting; a natural extension would be partially observable stochastic games (POSGs). A variant of POSGs, called factored-observation stochastic games (FOSGs), was recently proposed [14] that distinguishes between private and public observations in a similar fashion to our model but for finite-state models without NNs. Partial observability in FOSGs is dealt with via a mechanism that converts imperfect-information games into continuous-state (public belief state) perfect-information games [7, 14], such that many techniques for perfect-information games can also be applied. The fully-observable model can arguably serve as a vehicle to later solve the more complex case with imperfect information.

Contributions. First, we propose a new modelling formalism called *neuro-symbolic concurrent stochastic games (NS-CSGs)*, which comprise multiple finite-state agents endowed with perception mechanisms implemented via NNs and conventional, symbolic decision-making mechanisms. The agents proceed concurrently in a continuous, shared environment, since in many applications inputs supplied to NNs are real-valued vectors. Second, we tackle the problem of optimising zero-sum discounted cumulative rewards for NS-CSGs, under the assumption that agents have full state observability. Working with Borel state spaces, we establish restrictions on the modelling formalism which ensure that the NS-CSGs belong to a class of uncountable state-space CSGs [15] that are determined, and therefore prove the existence of the value of NS-CSGs.

Third, we present two algorithms to compute values and synthesise optimal strategies for these games, one using value iteration and the other using policy iteration. Our approach is based on formulating piecewise constant representations of the value functions and strategies for NS-CSGs by exploiting the underlying game structure. To the best of our knowledge, these are the first implementable algorithms for solving zero-sum CSGs over Borel state spaces with a convergence guarantee. The policy iteration approach is inspired by recent work for finite state spaces [5], which we generalise by employing piecewise constant or piecewise linear functions to ensure finite representability. This allows us to overcome the main issue that arises when solving Borel state space CSGs with policy iteration, namely that the value function may change from a Borel measurable function to a non-Borel measurable function across iterations. Finally, we illustrate our approach by modelling a dynamic vehicle parking case study with NS-CSGs, and then synthesising strategies for it using a prototype implementation of value iteration.

2. Background.

In this section we summarise the notation used in this paper.

2.1. Borel measurable functions.

Given a non-empty set X, we denote its Borel σ-algebra by $\mathcal{B}(X)$, and the sets in $\mathcal{B}(X)$ are called *Borel sets* of X. The pair $(X, \mathcal{B}(X))$ is a (standard) *Borel space* if there exists a metric on X that makes it a complete separable metric space (unless required for clarity, $\mathcal{B}(X)$ will be omitted). For convenience we will work with real vector spaces; however, this is not essential and any complete separable metric spaces could be used. For Borel spaces X and Y, a function $f : X \to Y$ is *Borel measurable* if $f^{-1}(B) \in \mathcal{B}(X)$ for all $B \in \mathcal{B}(Y)$ and *bimeasurable* if it is Borel measurable and $f(B) \in \mathcal{B}(Y)$ for all $B \in \mathcal{B}(X)$.

We denote by $\mathbb{F}(X)$ the space of all bounded, Borel measurable real-valued functions on a Borel space X, with respect to the unweighted sup-norm $\|J\| = \sup_{x \in X} |J(x)|$ for $J \in \mathbb{F}(X)$. For functions $J, K \in \mathbb{F}(X)$, we use $\max[J, K]$ and $\min[J, K]$ to denote their respective pointwise maximum and minimum functions, i.e., for each $x \in X$, we have $\mathrm{opt}[J, K](x) \coloneqq \mathrm{opt}\{J(x), K(x)\}$ for $\mathrm{opt} \in \{\min, \max\}$.

We now introduce notation and definitions for concepts that are fundamental to the abstraction on which our algorithms are performed. The abstraction is based on a decomposition of the uncountable state space into finitely many abstract regions such that all concrete states in a region have the same behaviour. In the definitions below, let $X \subset \mathbb{R}^{n_1}$ and $Y \subset \mathbb{R}^{n_2}$ for $n_1, n_2 > 0$.

Definition 2.1 (FCP and Borel FCP). *A* finite connected partition (FCP) *of X, denoted Φ, is a finite collection of disjoint connected subsets* (regions) *that cover X. Furthermore, Φ is a* Borel FCP *(BFCP) if each region $\phi \in \Phi$ is a Borel set of X.*

Definition 2.2 (PWC Borel measurable function). *A function $f : X \to Y$ is* piecewise constant (PWC) Borel measurable *if there exists a BFCP Φ of X such that $f : \phi \to Y$ is constant for all $\phi \in \Phi$ and is called a constant-BFCP of X for f.*

Definition 2.3 (PWL Borel measurable function). *A function $f : X \to Y$ is* piecewise linear (PWL) Borel measurable *if there exists a BFCP Φ of X such that $f : \phi \to Y$ is bounded and linear for all $\phi \in \Phi$.*

Definition 2.4 (BFCP invertible function). *A function $f : X \to Y$ is* BFCP invertible *if, for any BFCP Φ of Y, there exists a BFCP Φ' of X, called a consistent-*

BFCP of X for f to Φ, such that for each $\phi' \in \Phi'$ there exists $\phi \in \Phi$ such that $\{f(x) \mid x \in \phi'\} \subseteq \phi$.

For BFCPs Φ_1 and Φ_2 of X, we denote by $\Phi_1 + \Phi_2$ the smallest BFCP of X such that $\Phi_1 + \Phi_2$ is a refinement of both Φ_1 and Φ_2, which can be obtained by taking all the intersections between regions of Φ_1 and regions of Φ_2.

2.2. Probability measures. Let X be a Borel space. A function $f : \mathcal{B}(X) \to [0, 1]$ is a probability measure on X if $f(X) = 1$ and $\sum_{i\in I} f(B_i) = f(\cup_{i\in I} B_i)$ for any countable disjoint family of Borel sets $(B_i)_{i\in I}$. We denote the space of all probability measures on a Borel space X by $\mathbb{P}(X)$. For Borel spaces X and Y, a Borel measurable function $\sigma : Y \to \mathbb{P}(X)$ is called a *stochastic kernel* on X given Y (also known as a transition probability function from Y to X), and we denote by $\mathbb{P}(X \mid Y)$ the set of all stochastic kernels on X given Y. If $\sigma \in \mathbb{P}(X \mid Y)$, $y \in Y$ and $B \in \mathcal{B}(X)$, then we write $\sigma(B \mid y)$ for $\sigma(y)(B)$. It follows that $\sigma \in \mathbb{P}(X \mid Y)$ if and only if $\sigma(\cdot \mid y) \in \mathbb{P}(X)$ for all $y \in Y$ and $\sigma(B \mid \cdot\,)$ is Borel measurable for all $B \in \mathcal{B}(X)$.

2.3. Neural networks. A *neural network (NN)* is a real vector-valued function $f : \mathbb{R}^m \to \mathbb{R}^c$, where $m, c \in \mathbb{N}$, composed of a sequence of *layers* $h_1, \dots, h_k$, where $h_i : \mathbb{R}^{m_i} \to \mathbb{R}^{c_i}$ for $1 \leq i \leq k$, $m_1 = m$, $c_i = m_{i+1}$ for $1 \leq i \leq k-1$ and $c_k = c$. Each layer h_i is a data-processing module explicitly formulated as $h_i(x_i) = act_i(W_i x_i + b_i)$, where x_i is the input to the ith layer given by the output $h_{i-1}(x_{i-1})$ of the $(i-1)$th layer, act_i is an activation function, and $W_i x_i + b_i$ is a weighted sum of x_i for a weight matrix W_i and a bias vector b_i. An NN f is continuous for all popular activation functions, e.g., Rectified Linear Unit (ReLU), Sigmoid and Softmax. An NN f is said to be a *classifier* for a set of classes C of size c if, for any input $x \in \mathbb{R}^m$, the output $f(x) \in \mathbb{R}^c$ is a probability vector where, for any $1 \leq i \leq c$, the ith element of $f(x)$ represents the probability of the ith class of C, i.e., a classifier is a function $f : \mathbb{R}^m \to \mathbb{P}(C)$.

2.4. Concurrent stochastic games. We recall the model of two-player *concurrent stochastic games.*

Definition 2.5. *A (two-player)* concurrent stochastic game *(CSG) is a tuple $\mathsf{G} = (N, S, A, \Delta, \delta)$ where:*

- *$N = \{1, 2\}$ is the set of two players;*
- *S is a finite set of states;*
- *$A = (A_1 \cup \{\perp\}) \times (A_2 \cup \{\perp\})$ where A_i is a finite set of actions available to player $i \in N$ and $\perp$ is an idle action disjoint from the set $A_1 \cup A_2$;*
- *$\Delta \colon S \to 2^{(A_1 \cup A_2)}$ is an action assignment function;*
- *$\delta \colon (S \times A) \to \mathbb{P}(S)$ is a probabilistic transition function.*

In a state s of a CSG G, each player $i \in N$ selects an action from its available actions, i.e.. from the set $\Delta(s) \cap A_i$, if this set is non-empty, and selects the idle action $\perp$ otherwise. We denote the action choices for each player i in state s by $A_i(s)$, i.e. $A_i(s)$ equals $\Delta(s) \cap A_i$ if $\Delta(s) \cap A_i \neq \varnothing$ and equals $\{\perp\}$ otherwise and by $A(s)$ the possible joint actions in a state, i.e., $A(s) = A_1(s) \times A_2(s)$. Supposing each player i chooses action a_i, then with probability $\delta(s, (a_1, a_2))(s')$ there is a transition to state $s' \in S$. A *path* π of G is a sequence $\pi = s_0 \xrightarrow{\alpha_0} s_1 \xrightarrow{\alpha_1} \cdots$ such that $s_k \in S$, $\alpha_k \in A(s_k)$ and $\delta(s_k, \alpha_k)(s_{k+1}) > 0$ for all $k \geq 0$. For a path π, we denote by $\pi(k)$ the $(k+1)$th state, and $\pi[k]$ the action for the transition from $\pi(k)$ to $\pi(k+1)$.

3. Zero-sum neuro-symbolic concurrent stochastic games. In this section we introduce our model of *neuro-symbolic concurrent stochastic games* (NS-CSGs).

We restrict attention to two-agent (player) games as we are concerned with zero-sum games in which there are two agents with directly opposing objectives. However, the approach extends to multi-agent games, by allowing the agents to form two coalitions with directly opposing objectives. An NS-CSG comprises two interacting neuro-symbolic *agents* acting in a shared, continuous-state environment. Each agent has finitely many local states and actions, and is endowed with a perception mechanism implemented as an NN through which it can observe the state of the environment, storing the observations locally in *percepts*.

DEFINITION 3.1. *A (two-agent) neuro-symbolic concurrent stochastic game (NS-CSG)* C *comprises agents* $(\mathsf{Ag}_i)_{i \in N}$*, for* $N = \{1, 2\}$*, and an environment* E *where:*

$$\mathsf{Ag}_i = (S_i, A_i, \Delta_i, obs_i, \delta_i) \textit{ for } i \in N, \quad E = (S_E, \delta_E)$$

and we have:

- $S_i = Loc_i \times Per_i$ *is a set of states for* Ag_i*, where* $Loc_i \subseteq \mathbb{R}^{b_i}$ *and* $Per_i \subseteq \mathbb{R}^{d_i}$ *are finite sets of local states and percepts, respectively;*
- $S_E \subseteq \mathbb{R}^e$ *is a closed infinite set of environment states;*
- A_i *is a nonempty finite set of actions for* Ag_i*, and* $A \coloneqq (A_1 \cup \{\bot\}) \times (A_2 \cup \{\bot\})$ *is the set of joint actions, where* $\bot$ *is an idle action disjoint from* $A_1 \cup A_2$*;*
- $\Delta_i : S_i \to 2^{A_i}$ *is an available action function, defining the actions* Ag_i *can take in each state;*
- $obs_i : (Loc_1 \times Loc_2 \times S_E) \to Per_i$ *is an observation function for* Ag_i*, mapping the local states of all agents and the environment to a percept of the agent, implemented via an NN classifier for the set* Per_i*;*
- $\delta_i : (S_i \times A) \to \mathbb{P}(Loc_i)$ *is a partial probabilistic transition function for* Ag_i *determining the distribution over the agent's local states given its current state and joint action;*
- $\delta_E : (S_E \times A) \to S_E$ *is a partial deterministic transition function for the environment determining its next state given its current state and joint action.*

In an NS-CSG C the agents and environment execute concurrently and agents move between their local states probabilistically. For simplicity, we consider deterministic environments, but all the results extend directly to probabilistic environments with finite branching.

A (global) state for an NS-CSG comprises a state $s_i = (loc_i, per_i)$ for each agent Ag_i (a pair of a local state and percept, finite sets of real vectors) and an environment state s_E drawn from a closed set of real vectors. In state $s = (s_1, s_2, s_E)$, each Ag_i simultaneously chooses one of the actions available in its state s_i (if no action is available, i.e., $\Delta_i(s_i) = \varnothing$, then Ag_i chooses the idle action $\bot$). This results in a joint action $\alpha = (a_1, a_2) \in A$. Next, each Ag_i updates its local state to $loc'_i \in Loc_i$, according to its probabilistic local transition function δ_i, applied to the current agent state s_i and the joint action α. The environment updates its state to $s'_E \in S_E$ according to its deterministic transition function δ_E, based on its state s_E and the actions α. Finally, each Ag_i observes the new state of the other agents and the environment using the observation function obs_i to generate a new percept $per'_i = obs_i(loc'_1, loc'_2, s'_E)$. Thus, the game reaches the state $s' = (s'_1, s'_2, s'_E)$, where $s'_i = (loc'_i, per'_i)$ for $1 \leq i \leq 2$. A state $s = (s_1, s_2, s_E)$ with $s_i = (loc_i, per_i)$ is *percept compatible* if $per_i = obs_i(loc_1, loc_2, s_E)$ for $1 \leq i \leq 2$.

We assume that each observation function obs_i is implemented via an NN $f_i : \mathbb{R}^m \to \mathbb{P}(Per_i)$, where $m = b_1 + b_2 + e$, yielding a belief (normalised score) over the percepts of the agent, to which a rule is then applied that selects the percept with the maximum probability. However, observation functions can be defined by any function, including other types of machine learning models. For convenience, in our formalisation the percepts (and the local states) of agents are drawn from finite sets of real-valued vectors, but any finite set could be used instead.

3.1. Semantics of an NS-CSG. Formally, the semantics of an NS-CSG C is a concurrent stochastic game $[\![\mathsf{C}]\!]$ over the product of the states of the agents and the environment.

Definition 3.2 (Semantics of an NS-CSG). *Given an NS-CSG C consisting of two agents and an environment, the semantics of C is the CSG $[\![\mathsf{C}]\!] = (N, S, A, \Delta, \delta)$ where:*

- *$S \subseteq S_1 \times S_2 \times S_E$ is the set of global (percept compatible) states;*
- *$A = (A_1 \cup \{\bot\}) \times (A_2 \cup \{\bot\})$;*
- *$\Delta(s_1, s_2, s_E) = \Delta_1(s_i) \cup \Delta_2(s_2)$;*
- *$\delta : (S \times ((A_1 \cup \{\bot\}) \times (A_2 \cup \{\bot\}))) \to \mathbb{P}(S)$ is the partial probabilistic transition function, where for states $s = (s_1, s_2, s_E), s' = (s'_1, s'_2, s'_E) \in S$ and joint action $\alpha = (a_1, a_2) \in A$, if $a_i \in \Delta_i(s_i)$ when $\Delta_i(s_i) \neq \varnothing$ and $a_i = \bot$ otherwise for $1 \leq i \leq 2$, then $\delta(s, \alpha)$ is defined and, if $s'_i = (loc'_i, per'_i)$, $per'_i = obs_i(loc'_1, loc'_2, s'_E)$ for $1 \leq i \leq 2$ and $s'_E = \delta_E(s_E, \alpha)$, then*

$$\delta(s, \alpha)(s') = \delta_1(s_1, \alpha)(loc'_1)\delta_2(s_2, \alpha)(loc'_2)$$

and otherwise $\delta(s, \alpha)(s') = 0$.

Notice that the CSG $[\![\mathsf{C}]\!]$ is over percept compatible states and that the underlying transition relation δ is closed with respect to percept compatible states, by definition of δ_i for each agent Ag_i. Since δ_E is deterministic and Loc_i is a finite set, the branching set $\Theta(s, \alpha) = \{s' \mid \delta(s, \alpha)(s') > 0\}$ is finite for all $s \in S$ and $\alpha \in A(s)$. While the semantics of an NS-CSG is an instance of the general class of uncountable state space CSGs, its particular structure (see Definition 3.1) will be important in order to establish measurability and define our algorithms.

3.2. Strategies. The semantics of an NS-CSG is a CSG where the players are the agents, and therefore the behaviour of an NS-CSG is controlled by *strategies* for the agents. Since the state space S is uncountable due to the continuous environment state space, we follow the approach of [15] and require Borel measurable conditions on the choices that the strategies can make to ensure the measurability of the induced sets of paths.

The semantics of any NS-CSG will turn out to be an instance of the class of CSGs from [15], for which *stationary strategies* achieve optimal values [15, Theorem 2(ii), Theorem 3], and therefore, to simplify the presentation, we restrict our attention to stationary strategies and refer to them simply as strategies. Before we give the formal definition, since we work with real vector spaces we require the following lemma.

Lemma 3.3 (Borel spaces). *The sets S, S_i, S_E and A_i for $1 \leq i \leq 2$ are Borel spaces.*

Proof. By Theorem 27 [27, Chapter 9.6], S_i and S_E are separable metric spaces. By Theorem 12 [27, Chapter 9.4], S_i and S_E are complete metric spaces. Thus, S_i and S_E are Borel spaces. According to Theorem 1.10 [21, Chapter 1], the state space $S_1 \times S_2 \times S_E$, their Cartesian product space, is a Borel space. Since obs_i is Borel measurable (stated at the end of this section) for $1 \leq i \leq 2$, then for $(loc_i, per_i) \in S_i$

and $1 \leq i \leq 2$, the set

$$\{((loc_1, per_1), (loc_2, per_2), s_E) \in S \mid obs_i(loc_1, loc_2, s_E) = per_i \text{ for } 1 \leq i \leq 2\}$$

is a Borel set of $S_1 \times S_2 \times S_E$. Then, S is a Borel space as S_1 and S_2 are finite. Similarly, A_i are Borel spaces, as they are finite sets. □

Definition 3.4 (Strategy). *A* (stationary) strategy *for* Ag_i *of an NS-CSG* C *is a stochastic kernel* $\sigma_i : S \to \mathbb{P}(A_i)$*, i.e.,* $\sigma_i \in \mathbb{P}(A_i \mid S)$*, such that* $\sigma_i(A_i(s) \mid s) = 1$ *for all* $s \in S$*. A* (strategy) profile $\sigma = (\sigma_1, \sigma_2)$ *is a tuple of strategies for each agent. We denote by* Σ_i *the set of all strategies of* Ag_i *and by* $\Sigma = \Sigma_1 \times \Sigma_2$ *the set of profiles.*

For a state $s \in S$, we use $\mathbb{P}(A_i(s))$ to denote the largest subset of $\mathbb{P}(A_i)$ such that $u_i(A_i(s)) = 1$ for all probability measures $u_i \in \mathbb{P}(A_i(s))$.

3.3. Observability.

NS-CSGs are designed to model neuro-symbolic agents, whose operation depends on particular perception functions, which may result in imperfect information. However, in this paper we consider *full observability*, i.e., where agents' decisions can depend on the full state space. It is straightforward to extend the semantics above to *partially observable* CSGs (POSGs) [25, 26] where, for any state, each agent's observation function returns the agent's percept component of the state, and by restricting to observationally-equivalent strategies. However, our initial results on NS-CSGs focus on the simpler (but still challenging) case of full observability for the following reasons. First, the fully observable case represents an important baseline for this new model class, against which partially observable scenarios can later be evaluated. Second, the solution presented here can be directly used when converting imperfect-information games to perfect information, such as the mechanism described in [7, 14].

3.4. Zero-sum NS-CSGs.

We are concerned with *zero-sum* games, namely, two-agent games in which the agents have directly opposing objectives. The objectives we consider are *discounted accumulated rewards* and we assume the first agent tries to maximise the value of this objective and the second tries to minimise it. More precisely, for a reward structure of the form $r = (r_A, r_S)$, where $r_A : S \times A \to \mathbb{R}$ is an action reward function and $r_S : S \to \mathbb{R}$ is a state reward function, the accumulated discounted reward for a path π over the infinite-horizon is defined by:

$$Y(\pi) = \textstyle\sum_{k=0}^{\infty} \beta^k \big(r_A(\pi(k), \pi[k]) + r_S(\pi(k))\big) \tag{3.1}$$

where $\beta \in (0, 1)$ is the discount factor.

3.5. Assumptions.

We conclude our introduction to NS-CSGs by clarifying the assumptions that are required for the results presented in this paper.

Assumption 3.5. *For any NS-CSG* C *and reward structure* $r = (r_A, r_S)$*:*

(i) $\delta_E(\cdot, \alpha) : S_E \to S_E$ *is a bimeasurable, BFCP invertible function for all* $\alpha \in A$*;*

(ii) $obs_i(loc_1, loc_2, \cdot\,) : S_E \to Per_i$ *is a PWC Borel measurable function for all* $loc_i \in Loc_i$ *and* $1 \leq i \leq 2$*;*

(iii) $r_A(\cdot, \alpha), r_S : S \to \mathbb{R}$ *are bounded, PWC Borel measurable real-valued functions for all* $\alpha \in A$*.*

Our assumptions for NS-CSGs differ from existing stochastic games with Borel state spaces [15, 19, 11] in that the states have both discrete and continuous elements, while the observation and reward functions are required to be PWC Borel measurable. The PWC requirements in Assumption 3.5(ii) and (iii) and BFCP invertibility in Assumption 3.5(i) are needed to achieve PWC consistency, and hence ensure finitely many abstract state regions (and are used in Lemmas 4.2, 4.3, and 6.1 below). Bimeasurability in Assumption 3.5(ii) ensures the existence of the value of a NS-CSG with respect to a reward structure (and is used in Proposition 5.3). For the remainder of this paper, unless otherwise stated we will assume Assumption 3.5 holds.

The observation function obs_i for $1 \leq i \leq 2$ is implemented through an NN classifier, i.e., a continuous mapping from the finite real vector space $\mathbb{R}^m$ to the probability space over the finite percept classes Per_i, with the rule of selecting the class with the highest probability. Since f_i is continuous it follows that it is also Borel measurable. To ensure that obs_i is Borel measurable, we also need to consider the situation where the class with the highest probability is not unique. We assume such cases are resolved using a *tie-breaking rule* defined by a function $\kappa_i : 2^{Per_i} \to Per_i$ which, given a set of percepts, i.e., those with the highest probability, returns the selected percept. It follows that, if κ_i is a Borel measurable function, then Assumption 3.5(ii) holds. The case when observation functions are implemented using ReLU networks are discussed in Section 6, but we remark that Assumption 3.5(ii) allows a wider range of functions than just NNs for implementing observations.

4. Game structures.

NS-CSGs are a class of stochastic games with Borel state spaces in which the state space, observation function and reward function have structured form. Before designing our algorithms, we present several characteristics of these games, which distinguish them from the general case and which will be exploited later. For the remainder of this section we fix an NS-CSG C and reward structure r.

Since by Assumption 3.5 the perception function obs_i and the reward functions r_A and r_S are all PWC, there exist two FCPs of the state space, respectively denoting the abstract state spaces for perception and reward computations.

Definition 4.1 (Model-induced FCPs of the state space). *There exists:*

(i) a smallest FCP of S, called the perception FCP *of S and denoted Φ_P, such that all states in any $\phi \in \Phi_P$ have the same agents' states, i.e., if $(s_1, s_2, s_E), (s'_1, s'_2, s'_E) \in \phi$, then $s_i = s'_i$ for $1 \leq i \leq 2$;*

(ii) for each joint action $\alpha \in A$, a smallest FCP of S, called the reward FCP *of S under α and denoted Φ_R^α, such that all states in any $\phi \in \Phi_R^\alpha$ have the same state reward and reward when α is chosen, i.e., if $s, s' \in \phi$, then $r_A(s, \alpha) = r_A(s', \alpha)$ and $r_S(s) = r_S(s')$.*

The FCPs given in Definition 4.1 allow us to abstract an uncountable state space into a finite set of regions for percept and reward computations. Using Assumption 3.5, the FCPs Φ_P and Φ_R^α have the following properties, which will be used for the existence of the value of C and in our algorithms.

Lemma 4.2 (BFCPs). *For each $\alpha \in A$, Φ_P and Φ_R^α are BFCPs of S.*

Proof. We consider a region $\phi \in \Phi_P$. By Definition 4.1 all states in ϕ have the same agents' states, say $s_1 = (loc_1, per_1)$ and $s_2 = (loc_2, per_2)$. According to Assumption 3.5, $obs_i(loc_1, loc_2, \cdot\,) : S_E \to Per_i$ for $1 \leq i \leq 2$ is a PWC Borel measurable function. The pre-image of (per_1, per_2) under obs_1 and obs_2 over S given $s_1 = (loc_1, per_1)$ and $s_2 = (loc_2, per_2)$, denoted $obs^{-1}(per_1, per_2 \mid s_1, s_2)$, equals:

$$\{(s_1, s_2, s_E) \in S \mid obs_1(loc_1, loc_2, s_E) = per_1 \wedge obs_2(loc_1, loc_2, s_E) = per_2\}$$

and therefore is a Borel set of S. Since Φ_P is the smallest partition of S satisfying Definition 4.1(i), the regions in Φ_P, which lead to the percept (per_1, per_2) given s_1

and s_2, have no common boundary. Thus, $obs^{-1}(per_1, per_2 \mid s_1, s_2)$ is a finite union of disjoint regions in Φ_P which include the agents' states s_1 and s_2. Thus, each such region is a Borel set of S, meaning that $\phi \in \mathcal{B}(S)$. Thus, Φ_P is a BFCP of S. Since $r_A(\,\cdot\,,\alpha) + r_S(\,\cdot\,)$ is a PWC Borel measurable function on S by Assumption 3.5, we can similarly show that Φ_R^α is a BFCP of S for all $\alpha \in A$. ◻

Recall that $\Theta(s,\alpha) = \{s' \mid \delta(s,\alpha)(s') > 0\}$ is the finite branching set from $s \in S$ under $\alpha \in A(s)$.

Lemma 4.3 (Reachability consistency). *For each $\alpha \in A$, there exists a refinement BFCP Φ of Φ_P such that, for each $\phi \in \Phi$ and $\phi' \in \Phi_P$, if $\delta(s,\alpha)$ is defined for $s \in \phi$, then there exists $p_\alpha(\phi,\phi') \in [0,1]$ such that:*

1. *either $\delta(s,\alpha)(s') = p_\alpha(\phi,\phi') = 0$ for all $s \in \phi$ and $s' \in \phi'$;*
2. *or (i) if $s, \tilde{s} \in \phi$, then there exist unique states $s', \tilde{s}' \in S$ such that $s' = \Theta(s,\alpha) \cap \phi'$, $\tilde{s}' = \Theta(\tilde{s},\alpha) \cap \phi'$ and $\delta(s,\alpha)(s') = \delta(\tilde{s},\alpha)(\tilde{s}') = p_\alpha(\phi,\phi') > 0$, and (ii) there exists a bimeasurable, BFCP invertible function $q_\alpha : \phi \to \phi'$ such that $q_\alpha(s) = \Theta(s,\alpha) \cap \phi'$ for all $s \in \phi$.*

Proof. We compute the refinement Φ of Φ_P by dividing each ϕ of Φ_P such that the required reachability consistency holds. Now for any $\alpha \in A$ and $\phi \in \Phi_P$. By Definition 4.1, all states in ϕ have the same agents' states, say s_1 and s_2. To aid the proof, for each $\phi' \in \Phi_P$, we will construct a BFCP of ϕ based on ϕ', denoted $\Phi'(\phi,\phi')$, such that the reachability consistency to the region ϕ' holds in each region of $\Phi'(\phi,\phi')$. If $\delta(s,\alpha)$ is not defined for $s \in \phi$, we do not divide ϕ and let $\Phi'(\phi,\phi') = \{\phi\}$ for all $\phi' \in \Phi_P$ and the reachability consistency to ϕ' is preserved.

It remains to consider the case when $\delta(s,\alpha)$ is defined. Considering any $\phi' \in \Phi_P$, by Definition 4.1 there exists agent states $s_1' = (loc_1', per_1')$ and $s_2' = (loc_2', per_2')$ such that if $(s_1'', s_2'', s_E'') \in \phi'$ then $s_1 = s_1'$ and $s_2 = s_2'$. We have the following two cases to consider.

- If $\{(s_1', s_2', \delta_E(s_E,\alpha)) \in S \mid (s_1, s_2, s_E) \in \phi\} \cap \phi' = \varnothing$, $\delta_1(s_1,\alpha)(loc_1') = 0$ or $\delta_2(s_2,\alpha)(loc_2') = 0$, then we do not divide ϕ and let $\Phi'(\phi,\phi') = \{\phi\}$ and we have $\delta(s,\alpha)(s') = p_\alpha(\phi,\phi') = 0$ for all $s \in \phi$ and $s' \in \phi'$.
- If $(\cup_{s\in\phi}\Theta(s,\alpha)) \cap \phi'$ is non-empty, then since $\delta_E(\,\cdot\,,\alpha) : S_E \to S_E$ is BFCP invertible using Assumption 3.5 and ϕ' is a Borel measurable region there exists a BFCP $\Phi'(\phi,\phi')$ of ϕ such that for each $\phi_1 \in \Phi'(\phi,\phi')$:
  - either $\delta(s,\alpha)(s') = p_\alpha(\phi_1,\phi') = 0$ for all $s \in \phi_1$ and $s' \in \phi'$;
  - or for $s, \tilde{s} \in \phi_1$ there exist unique states $s', \tilde{s}' \in S$ such that $s' = \Theta(s,\alpha) \cap \phi'$, $\tilde{s}' = \Theta(\tilde{s},\alpha) \cap \phi'$ and $\delta(s,\alpha)(s') = \delta(\tilde{s},\alpha)(\tilde{s}') = p_\alpha(\phi_1,\phi') > 0$.

  It remains to show that the bimeasurable, BFCP invertible function q_α of (ii) exists, which follows from the the fact that $\delta_E(\,\cdot\,,\alpha) : S_E \to S_E$ is BFCP invertible.

Finally, we divide ϕ into a BFCP $\sum_{\phi'\in\Phi_P} \Phi'(\phi,\phi')$, and therefore each region of this BFCP has the required reachability consistency. ◻

5. Value of zero-sum NS-CSGs.

We now proceed by establishing the value of an NS-CSG C for an objective Y, i.e. for a reward structure r and discount factor β. We prove the existence of this value, which is a fixed point of a minimax operator. Using Banach's fixed-point theorem, a sequence of bounded, Borel measurable functions converging to this value is constructed.

Given a state s and profile $\sigma = (\sigma_1, \sigma_2)$, we denote by $\mathbb{E}_s^\sigma[Y]$ the expected value of the objective Y when starting from state s, defined in (3.1). The functions, for any

$s \in S$, given by:

$$\underline{V}(s) \coloneqq \sup_{\sigma_1 \in \Sigma_1} \inf_{\sigma_2 \in \Sigma_2} \mathbb{E}_s^{\sigma_1,\sigma_2}[Y] \quad \text{and} \quad \overline{V}(s) \coloneqq \inf_{\sigma_2 \in \Sigma_2} \sup_{\sigma_1 \in \Sigma_1} \mathbb{E}_s^{\sigma_1,\sigma_2}[Y]$$

are called the *lower value* and the *upper value*, respectively, of Y.

Definition 5.1 (Value function). *If $\underline{V}(s) = \overline{V}(s)$ for all $s \in S$, then C is determined with respect to the objective Y and the common function is called the* value *of C, denoted by $V^\star(\,\cdot\,)$, with respect to Y.*

We first introduce the spaces of feasible state-action and state-action-distribution pairs and properties of these spaces. More precisely, for $1 \leq i \leq 2$, we define:

$$\begin{aligned}
\Xi_i &\coloneqq \{(s, a_i) \in (S \times A_i) \mid a_i \in A_i(s)\} \\
\Lambda_i &\coloneqq \{(s, u_i) \in (S \times \mathbb{P}(A_i)) \mid u_i \in \mathbb{P}(A_i(s))\} \\
\Xi_{12} &\coloneqq \{(s, (a_1, a_2)) \in (S \times (A_1 \times A_2)) \mid a_1 \in A_1(s) \wedge a_2 \in A_2(s)\} \\
\Lambda_{12} &\coloneqq \{(s, (u_1, u_2)) \in (S \times (\mathbb{P}(A_1) \times \mathbb{P}(A_2))) \mid u_1 \in \mathbb{P}(A_1(s)) \wedge u_2 \in \mathbb{P}(A_2(s))\}\,.
\end{aligned}$$

Lemma 5.2 (Borel spaces). *For $1 \leq i \leq 2$, the spaces Ξ_i and Λ_i are Borel sets of $S \times A_i$ and $S \times \mathbb{P}(A_i)$, respectively. Furthermore, the spaces Ξ_{12} and Λ_{12} are Borel sets of $S \times (A_1 \times A_2)$ and $S \times (\mathbb{P}(A_1) \times \mathbb{P}(A_2))$, respectively.*

Proof. We first consider Ξ_i and Λ_i in the case when $i = 1$, and the case for $i = 2$ follows similarly. Since A_1 is finite, the sets Ξ_1 and Λ_1 can be rearranged as:

$$\begin{aligned}
\Xi_1 &= \textstyle\bigcup_{\hat{A}_1 \subseteq A_1} \big(\{s_1 \mid \Delta_1(s_1) = \hat{A}_1\} \times S_2 \times S_E \times \hat{A}_1\big) \cap (S \times A_1) \\
\Lambda_1 &= \textstyle\bigcup_{\hat{A}_1 \subseteq A_1} \big(\{s_1 \mid \Delta_1(s_1) = \hat{A}_1\} \times S_2 \times S_E \times \mathbb{P}(\hat{A}_1)\big) \cap (S \times \mathbb{P}(A_1))
\end{aligned}$$

respectively. Since $\hat{A}_1$ is a subset of the finite set A_1, then the sets $\hat{A}_1$ and $\mathbb{P}(\hat{A}_1)$ are Borel sets of A_1 and $\mathbb{P}(A_1)$, respectively. Since S_1 is a finite set, for any $\hat{A}_1 \subseteq A_1$, the set $\{s_1 \mid \Delta_1(s_1) = \hat{A}_1\}$ is a Borel set of S_1. Since S_2 and S_E are both Borel sets by Lemma 3.3, then the result follows by Theorem 1.10 [21, Chapter 1]. By the same argument, for Ξ_{12} and Λ_{12} we have:

$$\begin{aligned}
&\textstyle\bigcup_{\hat{A}_1 \subseteq A_1, \hat{A}_2 \subseteq A_2} \Big(\big(\{s_1 \mid \Delta_1(s_1) = \hat{A}_1\} \times \{s_2 \mid \Delta_2(s_2) = \hat{A}_2\} \times S_E\big) \times (\hat{A}_1 \times \hat{A}_2)\Big) \\
&\textstyle\bigcup_{\hat{A}_1 \subseteq A_1, \hat{A}_2 \subseteq A_2} \Big(\big(\{s_1 \mid \Delta_1(s_1) = \hat{A}_1\} \times \{s_2 \mid \Delta_2(s_2) = \hat{A}_2\} \times S_E\big) \times (\mathbb{P}(\hat{A}_1) \times \mathbb{P}(\hat{A}_2))\Big)
\end{aligned}$$

intersecting with $S \times A_1 \times A_2$ and $S \times \mathbb{P}(A_1) \times \mathbb{P}(A_2)$, and thus Ξ_{12} and Λ_{12} are also Borel sets of the respective spaces. □

Using Lemmas 4.2, 4.3, and 5.2, we have the following result.

Proposition 5.3 (Stochastic kernel transition function). *The probabilistic transition function δ is a stochastic kernel.*

Proof. For every $(s, \alpha) \in \Xi_{12}$, we have $\delta(s, \alpha)(\,\cdot\,) \in \mathbb{P}(S)$. We show that if $B \in \mathcal{B}(S)$, then $\delta(\,\cdot\,,\,\cdot\,)(B) : (S \times A) \to \mathbb{R}$ is Borel measurable on Ξ_{12}. More precisely we show that, for any $c \in \mathbb{R}$, pre-image of the Borel set $[c, \infty)$ of $\mathbb{R}$ under $\delta(\,\cdot\,,\,\cdot\,)(B)$ which is given by:

$$\delta^{-1}([c, \infty))(B) = \{(s, \alpha) \in \Xi_{12} \mid \delta(s, \alpha)(B) \geq c\}$$

is an element of $\mathcal{B}(\Xi_{12})$. If $c > 1$, then $\delta^{-1}([c, \infty))(B) = \varnothing \in \mathcal{B}(\Xi_{12})$, and if $c \leq 0$, then $\delta^{-1}([c, \infty))(B) = \Xi_{12} \in \mathcal{B}(\Xi_{12})$.

Therefore it remains to consider the case when $0 < c \leq 1$. For $\alpha \in A$, let Φ be the refinement of Φ_P meeting the conditions of Lemma 4.3. For each $\phi \in \Phi$ and $\phi' \in \Phi_P$ such that $p_\alpha(\phi, \phi') > 0$ using Lemma 4.3, let $q_\alpha : \phi \to \phi'$ be the associated bimeasurable, BFCP invertible function from Lemma 4.3. The image of ϕ under q_α into ϕ' for a fixed α is given by:

$$\hat{q}_\alpha(\phi, \phi') = \{s' \in \phi' \mid s' = q_\alpha(s) \wedge s \in \phi\}\,.$$

By Lemmas 4.2 and 4.3 both ϕ and ϕ' are Borel sets and q_α is bimeasurable, and therefore $\hat{q}_\alpha(\phi, \phi')$ is a Borel set. Next, the pre-image of the Borel set $\hat{q}_\alpha(\phi, \phi') \cap B$ under q_α over the region ϕ is given by:

$$\hat{q}_\alpha^{-1}(\phi, \hat{q}_\alpha(\phi, \phi') \cap B) = \{s \in \phi \mid q_\alpha(s) \in \hat{q}_\alpha(\phi, \phi') \cap B\}$$

and hence, as q_α is Borel measurable, is a Borel set. By combining this result with Lemma 4.3, all states in $\hat{q}_\alpha^{-1}(\phi, \hat{q}_\alpha(\phi, \phi') \cap B)$ are transferred to B with the same probability $p_\alpha(\phi, \phi')$. We denote the set of all transition probabilities from ϕ for a fixed α by $P_\alpha(\phi) = \{p_\alpha(\phi, \phi') > 0 \mid \phi' \in \Phi_P\}$. Then, the collection of the subsets of $P_\alpha(\phi)$ whose all elements' sum is greater or equal to c is defined as:

$$P_\alpha^{\geq c}(\phi) \coloneqq \left\{P' \subseteq P_\alpha(\phi) \mid \textstyle\sum_{p' \in P'} p' \geq c\right\}$$

which is finite. For every set $P' \in P_\alpha^{\geq c}(\phi)$, all states in the set:

$$O_\alpha(\phi, P') = \textstyle\bigcap_{p_\alpha(\phi,\phi') \in P'} \hat{q}_\alpha^{-1}(\phi, \hat{q}_\alpha(\phi, \phi') \cap B)$$

which reaches B under α with probability greater or equal to c, where $O_\alpha(\phi, P')$ is a Borel set as P' is a finite set. Thus, under a fixed α, the states in ϕ reaching B with probability greater or equal to c are given by:

$$O_\alpha(\phi) = \textstyle\bigcup_{P' \in P_\alpha^{\geq c}(\phi)} O_\alpha(\phi, P')$$

which is a Borel set since $P_\alpha^{\geq c}(\phi)$ is a finite set. Finally, we have:

$$\delta^{-1}([c, \infty))(B) = \textstyle\bigcup_{\phi \in \Phi} \bigcup_{\alpha \in A} \{(s, \alpha) \in \Xi_{12} \mid s \in O_\alpha(\phi)\}$$

from which it follows that $\delta^{-1}([c, \infty))(B) \in \mathcal{B}(\Xi_{12})$ by combining Lemmas 4.3 and 5.2 as required. □

Before discussing the value function, we first introduce an operator based on the classical Bellman equation.

Definition 5.4 (Minimax operator). *Given a bounded, Borel measurable real-valued function $V \in \mathbb{F}(S)$, the minimax operator $T : \mathbb{F}(S) \to \mathbb{F}(S)$ is defined, for any $s \in S$, by:*

$$[TV](s) \coloneqq \max_{u_1 \in \mathbb{P}(A_1(s))} \min_{u_2 \in \mathbb{P}(A_2(s))} \sum_{a_1 \in A_1(s)} \sum_{a_2 \in A_2(s)} Q(s, (a_1, a_2), V) u_1(a_1) u_2(a_2)$$

where for any $\alpha \in A(s)$:

$$Q(s, \alpha, V) \coloneqq r_A(s, \alpha) + r_S(s) + \beta \sum_{s' \in \Theta(s,\alpha)} \delta(s, \alpha)(s') V(s')\,.$$

The following theorem holds for the value function and minimax operator, which relies on Lemmas 3.3 and 5.2 and Proposition 5.3.

THEOREM 5.5 (Value function). *If C is an NS-CSG and Y is a discounted zero-sum objective, then*

(i) C is determined with respected to Y, i.e., $V^\star$ exists;
(ii) $V^\star$ is the unique fixed point of the operator T;
(iii) $V^\star$ is a bounded, Borel measurable function.

Proof. The proof is through showing that the NS-CSG C is an instance of a zero-sum stochastic game with Borel model as presented in [15].

From Lemma 3.3, we have that A_1, A_2 and S are complete and separable metric spaces. By Lemma 5.2, the spaces Ξ_i and Λ_i are Borel sets of $S \times A_i$ and $S \times \mathbb{P}(A_i)$ for $1 \leq i \leq 2$, respectively. By Proposition 5.3, δ is a Borel stochastic kernel.

Furthermore, since $r_A + r_S$ is bounded on $S \times A$, it follows that C with respect to the zero-sum objective Y is an instance of a zero-sum stochastic game with Borel model and discounted payoffs introduced in [15]. Therefore (i) follows from [15, Theorems 2 and 3], and (ii) from the discounted case of [15, Theorem 1]. Finally, for (iii), since $\beta \in (0, 1)$, we have that $V^\star$ is bounded, and therefore $V^\star$ is Borel measurable using [15, Lemma 3]. □

Since zero-sum NS-CSGs are determined games, as the following result demonstrates, optimal strategies exist.

LEMMA 5.6 (Optimal strategies). *There is a strategy $\sigma_1^\star$ for Ag_1 (and a dual strategy $\sigma_2^\star$ for Ag_2), called an optimal strategy of Ag_1, such that under $\sigma_1^\star$ in any state s the expected discounted reward is at least $V^\star(s)$ regardless of the strategy of Ag_2; more precisely, we have $\inf_{\sigma_2 \in \Sigma_2} \mathbb{E}_s^{\sigma_1^\star, \sigma_2}[Y] = V^\star(s)$ for all $s \in S$.*

Proof. The result follows from [15, Theorems 2 and 3]. □

Furthermore, the following lemma guarantees that value iteration (VI) can be used to find the value function.

LEMMA 5.7 (Convergence sequence). *For any $V^0 \in \mathbb{F}(S)$, the sequence $\langle V^t \rangle_{t \in \mathbb{N}}$, where $V^{t+1} = TV^t$, converges to $V^\star$. Moreover, each V^t is bounded, Borel measurable.*

Proof. Since $r_A + r_S$ is bounded, using [15, Lemma 2] we have that, if V^t is bounded, Borel measurable, then so is TV^t. The result then follows from the fact that $V^\star(s) = \lim_{t \to \infty} V^t(s)$ for all $s \in S$ if $V^{t+1} = TV^t$ for all $t \in \mathbb{N}$ [15]. □

6. Value iteration.

Despite the convergence result of Lemma 5.7, in practice there may not exist effective representations of the bounded Borel measurable functions V^t, due to the uncountable state space. We now show how VI can be used to *approximate* the values of C with respect to a discounted accumulated reward objective Y, and the corresponding optimal profile, based on a sequence of bounded, PWC Borel measurable functions.

LEMMA 6.1 (PWC and measurable consistency). *If $V \in \mathbb{F}(S)$ is bounded, PWC Borel measurable, then*

(i) $Q(\,\cdot\,, \alpha, V)$ is bounded, PWC Borel measurable for $\alpha \in A$;
(ii) TV is bounded, PWC Borel measurable.

Proof. We first proof that (i) holds. Considering any bounded, PWC Borel measurable V and joint action $\alpha \in A$, since $r_A(\,\cdot\,, \alpha) + r_S(\,\cdot\,)$ is bounded, PWC Borel measurable by Assumption 3.5, the fact that $Q(\,\cdot\,, \alpha, V)$ is bounded, PWC Borel measurable follows if we can show that the function $\overline{Q}(\,\cdot\,, \alpha, V)$ where:

$$\overline{Q}(\,\cdot\,, \alpha, V) \coloneqq \textstyle\sum_{s' \in \Theta(\cdot, \alpha)} \delta(\,\cdot\,, \alpha)(s') V(s')$$

is bounded, PWC Borel measurable. Boundedness follows because V is bounded. The indicator function of a subset $S' \subseteq S$ is the function $\chi_{S'} : S \to \mathbb{R}$ such that $\chi_{S'}(s) = 1$ if $s \in S'$ and 0 otherwise. Now $\chi_{S'}$ is Borel measurable if and only if S' is a Borel set of S [27]. For clarity, we use $q_\alpha(s;\phi,\phi')$ to refer to q_α from Lemma 4.3 for $\alpha \in A$, $s \in \phi$, $\phi \in \Phi$ and $\phi' \in \Phi_P$ (where again Φ is from Lemma 4.3). For any $s \in S$ such that $\delta(s,\alpha)$ is defined, we have:

$$
\begin{aligned}
\overline{Q}(s,\alpha,V) &= \sum_{\phi\in\Phi} \chi_\phi(s) \sum_{s'\in\Theta(s,\alpha)} \delta(s,\alpha)(s')V(s') \\
&= \sum_{\phi\in\Phi} \chi_\phi(s) \sum_{\phi'\in\Phi_P} p_\alpha(\phi,\phi')V(q_\alpha(s;\phi,\phi')) && \text{by Lemma 4.3} \\
&= \sum_{\phi\in\Phi}\sum_{\phi'\in\Phi_P} p_\alpha(\phi,\phi')\chi_\phi(s)V(q_\alpha(s;\phi,\phi')) && \text{rearranging.}
\end{aligned}
$$

Since ϕ is a Borel set of S, we have that χ_ϕ is Borel measurable. Next, we show that $V(q_\alpha(\,\cdot\,;\phi,\phi'))$ is Borel measurable on ϕ. Let Φ_V be a constant-BFCP of S for V. Given $c \in \mathbb{R}$, we denote by $\Phi_V^{\geq c}$ the set of regions in Φ_V on which $V \geq c$ holds. The pre-image of $[c,\infty)$ under $V(q_\alpha(\,\cdot\,;\phi,\phi'))$ defined on ϕ is given by:

$$
\begin{aligned}
V^{-1}(q_\alpha([c,\infty);\phi,\phi')) &= \{s\in\phi \mid V(q_\alpha(s;\phi,\phi')) \geq c\} \\
&= \textstyle\bigcup_{\phi_V\in\Phi_V^{\geq c}} \{s\in\phi \mid q_\alpha(s;\phi,\phi') \in \phi_V\}\,.
\end{aligned}
$$

Since $q_\alpha(s;\phi,\phi')$ is Borel measurable in $s \in \phi$ (q_α in Lemma 4.3) and ϕ_V is a Borel set of S, then $\{s\in\phi \mid q_\alpha(s;\phi,\phi') \in \phi_V\}$ is a Borel set of ϕ. Since $V^{-1}(q_\alpha([c,\infty);\phi,\phi'))$ is also a Borel set of ϕ by noting that $\Phi_V^{\geq c}$ is finite, it follows that $V(q_\alpha(\,\cdot\,;\phi,\phi'))$ is Borel measurable on ϕ. Therefore $\overline{Q}(\,\cdot\,,\alpha,V)$ is Borel measurable.

Next, since $q_\alpha(\,\cdot\,;\phi,\phi')$ is BFCP invertible on ϕ by Lemma 4.3, there exists a BFCP Φ_q of ϕ such that all states in each region of Φ_q are mapped into the same region of Φ_V under $q_\alpha(\,\cdot\,;\phi,\phi')$. Following this, $V(q_\alpha(\,\cdot\,;\phi,\phi'))$ is constant on each region of Φ_q. Therefore, using the fact that χ_ϕ is PWC, it follows that $\overline{Q}(\,\cdot\,,\alpha,V)$ is PWC, which completes the proof of (i).

Regarding (ii), from Lemma 5.7 we have that TV is bounded, Borel measurable. Since $Q(\,\cdot\,,\alpha,V)$ is PWC for any joint action $\alpha \in A$, $A(s)$ is PWC and A is finite, it follows that TV is PWC using the fact that the value of a zero-sum normal-formal game induced at every $s \in S$ is unique. □

We use the above PWC and measurable consistencies to iteratively construct a sequence of representation-friendly functions $\langle V^t\rangle_{t\in\mathbf{N}}$, where V^0 is an arbitrary PWC, Borel measurable function and $V^{t+1} = TV^t$. The following theorem is then a direct consequence of Lemmas 5.7 and 6.1.

Theorem 6.2 (PWC convergence sequence). *From an arbitrary bounded, PWC Borel measurable function $V^0 \in \mathbb{F}(S)$, the sequence $\langle V^t\rangle_{t\in\mathbb{N}}$ such that $V^{t+1} = TV^t$, converges to $V^\star$. Moreover, each V^t is bounded, PWC Borel measurable.*

Theorem 6.2 ensures convergence of bounded, PWC measurable functions. Using this, Algorithm 6.1 presents our VI scheme, in which the initial value function V^0 is a 0-valued PWC Borel measurable function defined over the BFCP $\Phi_P + \sum_{\alpha\in A}\Phi_R^\alpha$ of S. The steps of a single iteration of our VI algorithm are illustrated in Figure 1. These steps make use of the function *Preimage_BFCP*$(\Phi_{V^t}, \Phi_P, \langle\Phi_R^\alpha\rangle_{\alpha\in A})$ in Algorithm 6.2, to compute a refinement of $\Phi_P + \sum_{\alpha\in A}\Phi_R^\alpha$ that is a consistent-BFCP of S for δ to Φ_{V^t}. Then, in order to compute the value V'_ϕ over each region $\phi \in \Phi$, we take one

ALGORITHM 6.1 PWC-function VI

```
Input: NS-CSG $\mathsf{C}$, discounted zero-sum objective $Y$, perception FCP $\Phi_P$, reward FCPs $\langle \Phi_R^\alpha \rangle_{\alpha \in A}$, threshold $\varepsilon$
Output: Approximate value function $V$
$\Phi_{V^0} \leftarrow \Phi_P + \sum_{\alpha \in A} \Phi_R^\alpha$,  $V_\phi^0 \leftarrow 0$ for all $\phi \in \Phi_{V^0}$,  $V^0 \leftarrow \langle V_\phi^0 \rangle_{\phi \in \Phi_{V^0}}$,
$\bar{\varepsilon} \leftarrow 2\varepsilon$,  $t \leftarrow 0$
while $\bar{\varepsilon} > \varepsilon$ do
    $\Phi \leftarrow \mathit{Preimage\_BFCP}(\Phi_{V^t}, \Phi_P, \langle \Phi_R^\alpha \rangle_{\alpha \in A})$  (Algorithm 6.2)
    for $\phi \in \Phi$ do
        Take one state $s \in \phi$,  $V'_\phi \leftarrow (TV^t)(s)$
    $(\Phi_{V^{t+1}}, V^{t+1}) \leftarrow (\Phi, \langle V'_\phi \rangle_{\phi \in \Phi})$,
    $\bar{\varepsilon} \leftarrow \mathit{Dist}(V^{t+1}, V^t)$,
    $t \leftarrow t + 1$
return $V \leftarrow V^t$
```

ALGORITHM 6.2 BFCP iteration for PWC-function VI

```
Input: A BFCP $\Phi$ of $S$, perception FCP $\Phi_P$, reward FCPs $\langle \Phi_R^\alpha \rangle_{\alpha \in A}$
Output: A BFCP of $S$
procedure $\mathit{Preimage\_BFCP}(\Phi, \Phi_P, \langle \Phi_R^\alpha \rangle_{\alpha \in A})$
    $\Phi_{\text{pre}} \leftarrow \varnothing$
    for $\phi \in \Phi_P + \sum_{\alpha \in A} \Phi_R^\alpha$ do
        $B \leftarrow \varnothing$
        for $\alpha \in A$, $\phi' \in \{\phi' \in \Phi \mid (\cup_{s \in \phi} \Theta(s, \alpha)) \cap \phi' \neq \varnothing\}$ do
            $B \leftarrow B \cup \big\{\{s \in \phi \mid \Theta(s, \alpha) \cap \phi' \neq \varnothing\}\big\}$
        $\Phi_{\text{pre}} \leftarrow \Phi_{\text{pre}} \cup \{\phi_1 \in \mathit{Intersect}(B)\}$
    return $\Phi_{\text{pre}}$
```

state $s \in \phi$ and then find the value of a zero-sum normal form game [33] at s induced by Definition 5.4.

As a convergence criterion for VI, we detect when the difference between successive value approximations falls below a pre-specified threshold ε (as usual for VI, this does *not* guarantee an ε-optimal approximation). The function $\mathit{Dist}(V^{t+1}, V^t)$ computes the difference between V^{t+1} and V^t, which may be of different sizes due to the possible inconsistency of $\Phi_{V^{t+1}}$ and Φ_{V^t}. An intuitive method is to first evaluate V^{t+1} and V^t at a finite set of points, and then compute the maximum pointwise difference. In the usual manner for VI, an approximately optimal strategy can be extracted from the final step of the computation.

In Algorithm 6.2, the function $\mathit{Intersect}(B)$ returns a refinement of ϕ by computing all pairwise intersections for regions in B such that the refinement is a consistent-BFCP of ϕ for δ to Φ. The following corollary then follows from Lemmas 4.3 and 6.1.

COROLLARY 6.3 (BFCP iteration for VI). *In Algorithm 6.2, Φ_{pre} is a refinement of $\Phi_P + \sum_{\alpha \in A} \Phi_R^\alpha$ and a consistent-BFCP of S for δ to Φ.*

6.1. Polytope regions. The VI algorithm above assumes that each region in a BFCP is finitely representable. We now briefly discuss the use of BFCPs defined by *polytopes*, which work for the ReLU NNs. A polytope $\phi \subseteq \mathbb{R}^m$ is an intersection of ℓ halfspaces $\{x \in \mathbb{R}^m \mid g_k(x) \geq 0 \text{ for } 1 \leq k \leq \ell\}$, where $g_k(x) = W_k^\top x + b_k$ is a

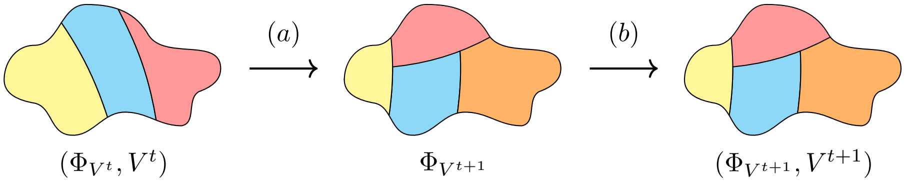

FIG. 1. *PWC-function VI. (a) Find new BFCP $\Phi_{V^{t+1}}$: refine $\Phi_P + \sum_{\alpha \in A} \Phi_R^\alpha$ to be a consistent-BFCP of S for δ to Φ_{V^t}; (b) compute a value for each $\phi \in \Phi_{V^{t+1}}$: take one state $s \in \phi$ and compute V^{t+1} by assigning each region ϕ with value $(TV^t)(s)$.*

linear function, i.e. $W_k \in \mathbb{R}^m$ and $b_k \in \mathbb{R}$, for $1 \leq k \leq \ell$. If ϕ_1 and ϕ_2 are polytopes, represented by $\{(W_k, b_k) \mid 1 \leq k \leq \ell'\}$ and $\{(W_k, b_k) \mid \ell' + 1 \leq k \leq \ell\}$, respectively, then the *sum* (refinement polytope) of ϕ_1 and ϕ_2, denoted $\phi_1 + \phi_2$, is the intersection of ℓ halfspaces $\{x \in \mathbb{R}^m \mid g_k(x) \geq 0 \text{ for } 1 \leq k \leq \ell\}$ and can be represented as $\{(W_k, b_k) \mid 1 \leq k \leq \ell\}$. Therefore the sum $\Phi_1 + \Phi_2$ of two BFCPs Φ_1 and Φ_2 can be computed by considering the sum $\phi_1 + \phi_2$ of all combinations of regions $\phi_1 \in \Phi_1$ and $\phi_2 \in \Phi_2$.

The *image* of a polytope $\phi = \{x \in \mathbb{R}^m \mid g_k(x) \geq 0 \text{ for } 1 \leq k \leq \ell\}$ under a linear function $f : \mathbb{R}^m \to \mathbb{R}^m$, where $f(x) = Dx + b$, $D \in \mathbb{R}^{m \times m}$ is non-singular and $b \in \mathbb{R}^m$, is the polytope $f(\phi) = \{x \in \mathbb{R}^m \mid W_k^\top D^{-1}x + b_k - W_k^\top D^{-1}b \geq 0, 1 \leq k \leq \ell\}$ and can be represented as $\{(D^{-\top} W_k, b_k - W_k^\top D^{-1}b) \mid 1 \leq k \leq \ell\}$. The *preimage* of ϕ under f is the polytope $f^{-1}(\phi) = \{x \in \mathbb{R}^m \mid W_k^\top Dx + b_k + W_k^\top b \geq 0, 1 \leq k \leq \ell\}$ and can be represented as $\{(D^\top W_k, b_k + W_k^\top b) \mid 1 \leq k \leq \ell\}$. Checking the feasibility of a set constrained by a set of linear inequalities can be solved by a linear program solver.

6.2. ReLU networks. If each observation function obs_i is implemented via a ReLU NN and there exist polytope constant-BFCPs for $r_A(\,\cdot\,, \alpha)$ and r_S for all $\alpha \in A$, then all regions in Φ_P and Φ_R^α for $\alpha \in A$ are polytopes. If the inverse function of $\delta_E(\,\cdot\,, \alpha)$ is linear and ϕ' is a polytope (line 8 in Algorithm 6.2), then $\{s \in \phi \mid \Theta(s, \alpha) \cap \phi' \neq \varnothing\}$ is a polytope. Therefore each region in Φ_{pre} is a polytope after every iteration and the operations over polytopes, including intersections, image and preimgae computations directly follow from the computation above.

7. Policy iteration. In this section, we show how policy iteration (PI) can be used to approximate the values and optimal strategies of an NS-CSGs C with respect to a discounted accumulated reward objective Y. Our algorithm takes ideas from recent work [5], which proposed a new PI method to solve zero-sum stochastic games with finite state spaces. Our algorithm is the first PI algorithm for CSGs with Borel state spaces and with a convergence guarantee. The authors of [37] presented a mixed value and policy iteration method for stochastic optimal control models with Borel state space, but not for games.

7.1. Minimax-action-free PI. Let $\gamma \in \mathbb{R}$ be a constant such that $\gamma > 1$ and $\gamma\beta < 1$, which will be used to distribute the discount factor β between policy evaluation and policy improvement of the two agents. Before introducing the two operators, we require the notion of a stationary Stackelberg (follower) strategy for Ag_2, which is a stochastic kernel $\overline{\sigma}_2 : \Lambda_1 \to \mathbb{P}(A_2)$, i.e., $\overline{\sigma}_2 \in \mathbb{P}(A_2 \mid \Lambda_1)$ such that $\overline{\sigma}_2(A_2(s) \mid (s, u_1)) = 1$ for all $(s, u_1) \in \Lambda_1$. This strategy is introduced only for the PI algorithm and implies that Ag_2 makes decisions conditioned on the current state s and the current choice, i.e. action distribution u_1, of Ag_1. We denote by $\overline{\Sigma}_2$ the set of all stationary Stackelberg strategies for Ag_2.

Definition 7.1 (Operator for the Max-Min value). *For each strategy $\sigma_1 \in \Sigma_1$ of Ag_1 and bounded, Borel measurable real-valued function $V_2 \in \mathbb{F}(\Lambda_1)$, we define the operator $H^1_{\sigma_1,V_2} : \mathbb{F}(\Lambda_1) \to \mathbb{F}(S)$ such that for $J_2 \in \mathbb{F}(\Lambda_1)$ and $s \in S$:*

$$H^1_{\sigma_1,V_2}(J_2)(s) = \frac{1}{\gamma}\min\{J_2(s,\sigma_1(s)), V_2(s,\sigma_1(s))\} = \frac{1}{\gamma}\min\{J_2(s,u_1), V_2(s,u_1)\}$$

where $\sigma_1(s) = u_1 \in \mathbb{P}(A_1(s))$.

Definition 7.2 (Operator for the Min-Max value). *For each Stackelberg (follower) strategy $\overline{\sigma}_2 \in \overline{\Sigma}_2$ of Ag_2 and each $V_1 \in \mathbb{F}(S)$, we define the operator $H^2_{\overline{\sigma}_2,V_1} : \mathbb{F}(S) \to \mathbb{F}(\Lambda_1)$ such that for $J_1 \in \mathbb{F}(S)$ and $(s,u_1) \in \Lambda_1$:*

$$H^2_{\overline{\sigma}_2,V_1}(J_1)(s,u_1) = \; = \sum_{a_1 \in A_1(s)} \sum_{a_2 \in A_2(s)} Q(s,(a_1,a_2),\gamma \max[J_1,V_1]) u_1(a_1) u_2(a_2)$$

where $\overline{\sigma}_2(\cdot \mid (s,u_1)) = u_2 \in \mathbb{P}(A_2(s))$.

Unlike the classical PI algorithms by Hoffman and Karp [12] and Pollatschek and Avi-Itzhak [23], following [5], our PI algorithm separates the policy evaluation and policy improvement of the maximiser (Ag_1) and the minimiser (Ag_2) through the use of the operators from Definition 7.1 and Definition 7.2, respectively. To track the value functions after performing policy evaluation of Ag_1 and Ag_2, our PI algorithm introduces value functions J_1 and J_2. In addition, the value functions V_1 and V_2 are introduced to avoid the oscillatory behavior of Pollatschek and Avi-Itzhak PI algorithm [23], thus ensuring convergence, and are updated only during policy improvement. The role of γ is to split the discount factor β such that all the operators corresponding to policy evaluation and policy improvement of the two agents are contraction mappings, which will ensure convergence.

Before presenting our PI algorithm, we define two classes of functions, which play a key role in characterizing the functions and strategies generated during each iteration of our PI algorithm.

Definition 7.3 (Bounded, CON-PWL Borel measurable function). *A function $f \in \mathbb{F}(\Lambda_1)$ is a* bounded, constant-piecewise-linear (CON-PWL) Borel measurable function *if there exists a BFCP Φ of S such that for each $\phi \in \Phi$, $A_1(s) = A_1(s')$ for all $s, s' \in \phi$ and it generates $\Theta = \{\theta(\phi) \mid \phi \in \Phi\}$ where $\theta(\phi) = \{(s,u_1) \in \Lambda_1 \mid s \in \phi\}$, a BFCP of Λ_1, such that for each region $\theta(\phi) \in \Theta$:*

(i) $f(\,\cdot\,,u_1) : \phi \to \mathbb{R}$ is constant for all $u_1 \in \mathbb{P}(A_1(s))$ where $s \in \phi$;

(ii) $f(s,\,\cdot\,) : \mathbb{P}(A_1(s)) \to \mathbb{R}$ is bounded, PWL for all $s \in \phi$.

Definition 7.4 (CON-PWC stochastic kernel). *A function $f \in \overline{\Sigma}_2$ is a* constant, piecewise-constant (CON-PWC) stochastic kernel *if there exists a BFCP Φ of S such that for each $\phi \in \Phi$, $A(s) = A(s')$ for all $s, s' \in \phi$ and it generates $\Theta = \{\theta(\phi) \mid \phi \in \Phi\}$ where $\theta(\phi) = \{(s,u_1) \in \Lambda_1 \mid s \in \phi\}$, a BFCP of Λ_1, such that for each region $\theta(\phi) \in \Theta$:*

(i) $f(\,\cdot\,,u_1) : \phi \to \mathbb{P}(A_2(s))$ is constant for all $u_1 \in \mathbb{P}(A_1(s))$ where $s \in \phi$;

(ii) $f(s,\,\cdot\,) : \mathbb{P}(A_1(s)) \to \mathbb{P}(A_2(s))$ is PWC for all $s \in \phi$.

Figure 2 presents an examples of a bounded, CON-PWL Borel measurable function and CON-PWC stochastic kernel over a region. Each bounded, CON-PWL Borel measurable function f can be represented by a finite set of vectors $\{(D_{\phi,\phi'}, b_{\phi,\phi'}) \in \mathbb{R}^{|A_1|} \times \mathbb{R} \mid \phi \in \Phi \wedge \phi' \in \Phi'(\phi)\}$ such that $f(s,u_1) = D^{\top}_{\phi,\phi'} u_1 + b_{\phi,\phi'}$ for all $s \in \phi$ and $u_1 \in \phi'$, where Φ is a BFCP of S for f using Definition 7.3 and $\Phi'(\phi)$ is a BFCP

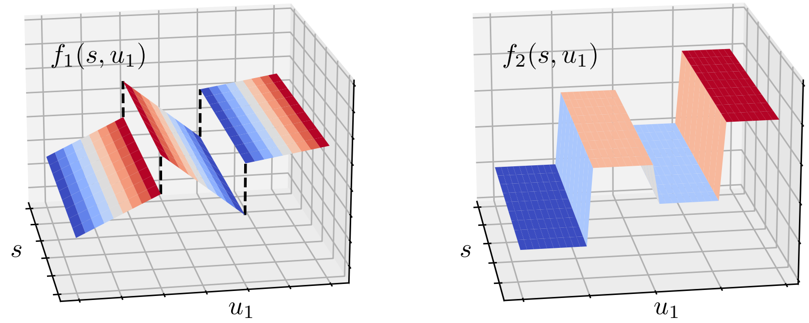

FIG. 2. *Two functions over one region in a BFCP of Λ_1. Bounded, constant-piecewise-linear (CON-PWL) Borel measurable function (left): given u_1, $f_1(s, u_1)$ is constant in s, and given s, $f_1(s, u_1)$ is PWL in u_1. Constant-piecewise-constant (CON-PWC) stochastic kernel (right): given u_1, $f_2(s, u_1)$ is constant in s, and given s, $f_2(s, u_1)$ is PWC in u_1.*

of $\{u_1 \in \mathbb{P}(A_1) \mid (s, u_1) \in \theta(\phi)\}$, and $\theta(\phi) \in \Theta$ again using Definition 7.3 such that, over each region $\phi' \in \Phi'(\phi)$, $f(s, u_1)$ is linear in u_1 given $s \in \phi$. Similarly using Definition 7.4, each CON-PWC stochastic kernel f can be represented by a finite set of vectors $\{D_{\phi,\phi'} \in \mathbb{P}(A_2) \mid \phi \in \Phi \wedge \phi' \in \Phi(\phi)\}$ such that $f(s, u_1) = D_{\phi,\phi'}$ for all $s \in \phi$ and $u_1 \in \phi'$, where over each region $\phi' \in \Phi'(\phi)$, $f(s, u_1)$ is constant in u_1 given $s \in \phi$.

We introduce a criterion for selecting the maximum or minimum solution over a region, by which the strategies from policy improvement are finitely representable.

DEFINITION 7.5 (CON-1 solution). *Let $f \in \mathbb{F}(\Lambda_1)$ be a bounded, CON-PWL Borel measurable function. Using Definition 7.3 there exists a BFCP Φ of S for f. Now, for each $\phi \in \Phi$, if there exists $u_1^\phi \in \mathbb{P}(A_1(s))$ such that:*

$$f(s, u_1^\phi) = \max\nolimits_{u_1 \in \mathbb{P}(A_1(s))} f(s, u_1)$$

for all $s \in \phi$ and if σ_1 is a strategy of Ag_1 such that $\sigma_1(s) = u_1^\phi$ for all $s \in \phi$, then σ_1 is a constant-1 (CON-1) solution *of f over ϕ.*

DEFINITION 7.6 (CON-2 solution). *Let $f \in \mathbb{F}(\Lambda_{12})$ be a bounded, Borel measurable function. If there exists a BFCP Θ of Λ_1 where, for each $\theta \in \Theta$, $A_2(s)$ is constant for all $(s, u_1) \in \theta$ and there exists $u_2^\theta \in \mathbb{P}(A_2(s))$ such that:*

$$f(s, u_1, u_2^\theta) = \min\nolimits_{u_2 \in \mathbb{P}(A_2(s))} f(s, u_1, u_2)$$

for all $(s, u_1) \in \theta$, and if $\overline{\sigma}_2$ is a Stackelberg strategy for Ag_2 such that $\overline{\sigma}_2(s, u_1) = u_2^\theta$ for all $(s, u_1) \in \theta$, then $\overline{\sigma}_2$ is a constant-2 (CON-2) solution *of f over θ.*

We now use the operators of Definitions 7.1 and 7.2, together with the functions and solutions from Definitions 7.3 to 7.6 specialised to Borel state spaces with game structures described in Sections 3 and 4, to derive a PI algorithm called *Minimax-action-free PI* (Algorithm 7.1) for strategy synthesis of NS-CSGs with Borel state spaces. Our algorithm closely follows the steps of the PI method of [5] for finite state spaces, but has to resolve a number of issues due to the uncountability of the underlying state space and the need to ensure Borel measurability at each iteration. To overcome these issues we (i) introduce bounded, CON-PWL Borel measurable functions and CON-PWC Borel measurable strategies to ensure measurability and finite representability; (ii) work with CON-1 and CON-2 solutions for policy improvement to ensure that the strategies generated are finitely representable and consistent; and

Algorithm 7.1 Iteration t of Minimax-action-free PI

```
Input: NS-CSG $\mathsf{C}$, PWC $\sigma_1^t \in \Sigma_1$, CON-PWC $\overline{\sigma}_2^t \in \overline{\Sigma}_2$, PWC $J_1^t, V_1^t \in \mathbb{F}(S)$,
CON-PWL $J_2^t, V_2^t \in \mathbb{F}(\Lambda_1)$
Perform one of the following four iterations.
  Policy evaluation of $\mathsf{Ag}_1$:
    $J_1^{t+1} \leftarrow H^1_{\sigma_1^t, V_2^t}(J_2^t)$ via *PE1*, $\sigma_1^{t+1} \leftarrow \sigma_1^t$,
    $V_1^{t+1} \leftarrow V_1^t$, $\overline{\sigma}_2^{t+1} \leftarrow \overline{\sigma}_2^t$, $J_2^{t+1} \leftarrow J_2^t$, $V_2^{t+1} \leftarrow V_2^t$
  Policy improvement of $\mathsf{Ag}_1$ by CON-1 solution:
    $\sigma_1^{t+1}(s) \in \operatorname{argmax}_{u_1 \in \mathbb{P}(A_1(s))} H^1_{u_1, V_2^t}(J_2^t)(s)$,
    $V_1^{t+1} \leftarrow H^1_{\sigma_1^{t+1}, V_2^t}(J_2^t)$ via *PI1*,
    $J_1^{t+1} \leftarrow J_1^t$, $\overline{\sigma}_2^{t+1} \leftarrow \overline{\sigma}_2^t$, $J_2^{t+1} \leftarrow J_2^t$, $V_2^{t+1} \leftarrow V_2^t$
  Policy evaluation of $\mathsf{Ag}_2$:
    $J_2^{t+1} \leftarrow H^2_{\overline{\sigma}_2^t, V_1^t}(J_1^t)$ via *PE2*, $\sigma_1^{t+1} \leftarrow \sigma_1^t$,
    $J_1^{t+1} \leftarrow J_1^t$, $V_1^{t+1} \leftarrow V_1^t$, $\overline{\sigma}_2^{t+1} \leftarrow \overline{\sigma}_2^t$, $V_2^{t+1} \leftarrow V_2^t$
  Policy improvement of $\mathsf{Ag}_2$ by CON-2 solution:
    $\overline{\sigma}_2^{t+1}(s, u_1) \in \operatorname{argmin}_{u_2 \in \mathbb{P}(A_2(s))} H^2_{u_2, V_1^t}(J_1^t)(s, u_1)$,
    $V_2^{t+1} \leftarrow H^2_{\overline{\sigma}_2^{t+1}, V_1^t}(J_1^t)$ via *PI2*,
    $\sigma_1^{t+1} \leftarrow \sigma_1^t$, $J_1^{t+1} \leftarrow J_1^t$, $V_1^{t+1} \leftarrow V_1^t$, $J_2^{t+1} \leftarrow J_2^t$
$t \leftarrow t + 1$
```

(iii) propose a BFCP iteration algorithm (Algorithm 7.2) and a BFCP-based computation algorithm (Algorithm 7.3) to compute a new BFCP of the state space and the values or strategies over this BFCP. We also provide a simpler proof than that presented in [5], which does not require the introduction of any new concepts except those used in the algorithm.

The Minimax-action-free PI algorithm is initialized with a strategy for each agent, σ_1^0 and $\overline{\sigma}_2^0$, and four functions, $J_1^0, V_1^0, J_2^0, V_2^0$, where Algorithm 7.2 gives one BFCP for each initial strategy and function. An iteration of the Minimax-action-free PI is given in Algorithm 7.1. As shown later, the order and frequency by which the possible four iterations of Algorithm 7.1 are run do not affect the convergence, as long as each of them is performed infinitely often. Although we have not discussed the pros and cons of an asynchronous implementation, the Minimax-action-free PI algorithm indeed allows such an implementation, as discussed in [5] and for its single-agent counterparts in [6].

For each of the four iterations, Algorithm 7.2 provides a way to compute new BFCPs and the results below demonstrate that, over each region of these BFCPs, the corresponding computed strategies and value functions are either constant, PWC or PWL. Therefore, we can follow similar steps to our VI algorithm (see Algorithm 6.1) to compute the value functions of these new strategies and value functions (see Algorithm 7.3). The idea is to first compute the BFCPs $\Phi_{J_1^{t+1}}$, $\Phi_{V_1^{t+1}}$, $\Phi_{\sigma_1^{t+1}}$, $\Theta_{J_2^{t+1}}$, $\Theta_{V_2^{t+1}}$ and $\Theta_{\overline{\sigma}_2^{t+1}}$ via Algorithm 7.2 and then use these BFCPs to compute strategies and value functions using Algorithm 7.3. For instance, if the policy improvement of Ag_2 is chosen at iteration t then we proceed as follows. First, new BFCPs are computed via Algorithm 7.2. Second, *PI2* of Algorithm 7.3 is performed. In this second step we take each region $\theta \in \Theta_{\overline{\sigma}_2^{t+1}}$, let $\phi = \{s \mid (s, u_1) \in \theta\}$, then take one state $s' \in \phi$, and compute a BFCP Φ_u of $\mathbb{P}(A_1(s'))$ such that $\min_{u_2 \in \mathbb{P}(A_2(s'))} H^2_{u_2, V_1^t}(J_1^t)(s', u_1)$ is con-

ALGORITHM 7.2 BFCP iteration t for PI

```
**Input:** Perception FCP $\Phi_P$, reward FCPs $\langle \Phi_R^\alpha \rangle_{\alpha \in A}$
**Output:** BFCPs $\langle \Phi_{J_1^t}, \Phi_{V_1^t}, \Phi_{\sigma_1^t}, \Theta_{J_2^t}, \Theta_{V_2^t}, \Theta_{\overline{\sigma}_2^t} \rangle_{t \in \mathbb{N}}$ for $\langle J_1^t, V_1^t, \sigma_1^t, J_2^t, V_2^t, \overline{\sigma}_2^t \rangle_{t \in \mathbb{N}}$
$\Phi_{J_1^0}, \Phi_{V_1^0}, \Phi_{\sigma_1^0} \leftarrow \{S\}, \ \Theta_{J_2^0}, \Theta_{V_2^0}, \Theta_{\overline{\sigma}_2^0} \leftarrow \{\Lambda_1\}$
**while** Algorithm 7.1 performs iteration $t$ **do**
    **if** policy evaluation of $\mathsf{Ag}_1$ is chosen **then**
        *Preprocess_maximiser*(),
        $\Phi_{J_1^{t+1}} \leftarrow \Phi_{\sigma_1^t} + \Phi_{J_2^t} + \Phi_{V_2^t}, \ \Phi_{V_1^{t+1}} \leftarrow \Phi_{V_1^t}, \ \Phi_{\sigma_1^{t+1}} \leftarrow \Phi_{\sigma_1^t}$
    **if** policy improvement of $\mathsf{Ag}_1$ is chosen **then**
        *Preprocess_maximiser*(),
        $\Phi_{\sigma_1^{t+1}} \leftarrow \Phi_{J_2^t} + \Phi_{V_2^t}, \ \Phi_{V_1^{t+1}} \leftarrow \Phi_{\sigma_1^{t+1}}, \ \Phi_{J_1^{t+1}} \leftarrow \Phi_{J_1^t}$
    **if** policy evaluation of $\mathsf{Ag}_2$ is chosen **then**
        *Preprocess_minimiser*(),
        $\Phi_{\overline{\sigma}_2^t} \leftarrow \big\{\{s \mid (s, u_1) \in \theta\} \mid \theta \in \Theta_{\overline{\sigma}_2^t}\big\}$,
        $\Theta_{J_2^{t+1}} \leftarrow \big\{\{(s, u_1) \in \Lambda_1 \mid s \in \phi\} \mid \phi \in \Phi_{\hat{Q}^{t+1}} + \Phi_{\overline{\sigma}_2^t}\big\}$,
        $\Theta_{V_2^{t+1}} \leftarrow \Theta_{V_2^t}, \ \Theta_{\overline{\sigma}_2^{t+1}} \leftarrow \Theta_{\overline{\sigma}_2^t}$
    **if** policy improvement of $\mathsf{Ag}_2$ is chosen **then**
        *Preprocess_minimiser*(),
        $\Theta_{\overline{\sigma}_2^{t+1}} \leftarrow \big\{\{(s, u_1) \in \Lambda_1 \mid s \in \phi\} \mid \phi \in \Phi_{\hat{Q}^{t+1}}\big\}$,
        $\Theta_{V_2^{t+1}} \leftarrow \Theta_{\overline{\sigma}_2^{t+1}}, \ \Theta_{J_2^{t+1}} \leftarrow \Theta_{J_2^t}$
**return** $\langle \Phi_{J_1^t}, \Phi_{V_1^t}, \Phi_{\sigma_1^t}, \Theta_{J_2^t}, \Theta_{V_2^t}, \Theta_{\overline{\sigma}_2^t} \rangle_{t \in \mathbb{N}}$

**procedure** *Preprocess_maximiser*()
    $\Theta_{J_2^{t+1}} \leftarrow \Theta_{J_2^t}, \ \Theta_{V_2^{t+1}} \leftarrow \Theta_{V_2^t}, \ \Theta_{\overline{\sigma}_2^{t+1}} \leftarrow \Theta_{\overline{\sigma}_2^t}$,
    $\Phi_{J_2^t} \leftarrow \big\{\{s \mid (s, u_1) \in \theta\} \mid \theta \in \Theta_{J_2^t}\big\}, \ \Phi_{V_2^t} \leftarrow \big\{\{s \mid (s, u_1) \in \theta\} \mid \theta \in \Theta_{V_2^t}\big\}$
**procedure** *Preprocess_minimiser*()
    $\Phi_{J_1^{t+1}} \leftarrow \Phi_{J_1^t}, \ \Phi_{V_1^{t+1}} \leftarrow \Phi_{V_1^t}, \ \Phi_{\sigma_1^{t+1}} \leftarrow \Phi_{\sigma_1^t}$,
    $\Phi_{\hat{Q}^{t+1}} \leftarrow$ *Preimage_BFCP*$(\Phi_{J_1^t} + \Phi_{V_1^t}, \Phi_P, \langle \Phi_R^\alpha \rangle_{\alpha \in A})$
```

stant over each region $\phi_u \in \Phi_u$ and for all $u_1 \in \phi_u$. Third, take one $u_1' \in \phi_u$ and find $u_2' \in \mathbb{P}(A_2(s'))$ that minimises $H^2_{u_2, V_1^t}(J_1^t)(s', u_1')$. Fourth, we let $\overline{\sigma}_2^{t+1}(s, u_1) = u_2'$ for all $s \in \phi$ and $u_1 \in \phi_u$, which is a CON-2 solution of $H^2_{u_2, V_1^t}(J_1^t)(s, u_1)$ over $\{(s, u_1) \mid s \in \phi, u_1 \in \phi_u\}$ by Lemma 7.10 and $V_2^{t+1}(s, u_1)$ is CON-linear in $s \in \phi$ and $u_1 \in \phi_u$. Finally, we copy the other strategies and value functions for the next iteration.

We next introduce the following lemmas in order to prove that the strategies and value functions generated during each iteration of the Minimax-action-free PI algorithm are finitely representable.

LEMMA 7.7 (Evaluation consistency for Ag_1). *If $\sigma_1^t \in \Sigma_1$ is a PWC stochastic kernel and $J_2^t, V_2^t \in \mathbb{F}(\Lambda_1)$ are bounded, CON-PWL Borel measurable and an iteration of policy evaluation of Ag_1 is performed (procedure PE1), then $J_1^{t+1} = H^1_{\sigma_1^t, V_2^t}(J_2^t)$ is bounded, PWC Borel measurable.*

Proof. Suppose $\sigma_1^t \in \Sigma_1$ is a PWC stochastic kernel and $J_2^t, V_2^t \in \mathbb{F}(\Lambda_1)$ are bounded, CON-PWL Borel measurable. Since σ_1^t is a PWC stochastic kernel, there

Algorithm 7.3 BFCP based computation for PI

```
Input: $J_1^t, V_1^t, \sigma_1^t, J_2^t, V_2^t, \overline{\sigma}_2^t, \Phi_{J_1^{t+1}}, \Phi_{\sigma_1^{t+1}}, \Theta_{J_2^{t+1}}, \Theta_{\overline{\sigma}_2^{t+1}}$
procedure PE1
    for $\phi \in \Phi_{J_1^{t+1}}$ do
        Take one state $s \in \phi$, and then $J_{1,\phi}^{t+1} \leftarrow H^1_{\sigma_1^t, V_2^t}(J_2^t)(s)$
    return $J_1^{t+1} \leftarrow \langle J_{1,\phi}^{t+1} \rangle_{\phi \in \Phi_{J_1^{t+1}}}$
procedure PI1
    for $\phi \in \Phi_{\sigma_1^{t+1}}$ do
        Take one state $s \in \phi$, and then $u_1 \in \mathrm{argmax}_{u_1 \in \mathbb{P}(A_1(s))} H^1_{u_1, V_2^t}(J_2^t)(s)$
        $\sigma_{1,\phi}^{t+1} \leftarrow u_1, \ \ V_{1,\phi}^{t+1} \leftarrow \max_{u_1 \in \mathbb{P}(A_1(s))} H^1_{u_1, V_2^t}(J_2^t)(s)$
    return $\sigma_1^{t+1} \leftarrow \langle \sigma_{1,\phi}^{t+1} \rangle_{\phi \in \Phi_{\sigma_1^{t+1}}}, \ \ V_1^{t+1} \leftarrow \langle V_{1,\phi}^{t+1} \rangle_{\phi \in \Phi_{\sigma_1^{t+1}}}$
procedure PE2
    for $\theta \in \Theta_{J_2^{t+1}}$ do
        $\phi \leftarrow \{s \mid (s, u_1) \in \theta\}$
        Take one state $s \in \phi$, and then compute a BFCP $\Phi_u$ of $\mathbb{P}(A_1(s))$ such that
          over $\phi_u \in \Phi_u$, $H^2_{\overline{\sigma}_2^t, V_1^t}(J_1^t)(s, u_1)$ is linear in $u_1$
        $J_{2,\phi,\phi_u}^{t+1} \leftarrow H^2_{\overline{\sigma}_2^t, V_1^t}(J_1^t)(s, u_1)$ is linear in $u_1$
    return $J_2^{t+1} \leftarrow \langle J_{2,\phi,\phi_u}^{t+1} \rangle_{\theta \in \Theta_{J_2^{t+1}}}$
procedure PI2
    for $\theta \in \Theta_{\overline{\sigma}_2^{t+1}}$ do
        $\phi \leftarrow \{s \mid (s, u_1) \in \theta\}$
        Take one state $s' \in \phi$, and then compute a BFCP $\Phi_u$ of $\mathbb{P}(A_1(s'))$ such that
          over $\phi_u \in \Phi_u$, $\min_{u_2 \in \mathbb{P}(A_2(s'))} H^2_{u_2, V_1^t}(J_1^t)(s', u_1)$ is constant for all $u_1 \in \phi_u$
        Take $u_1' \in \phi_u$, and $u_2' \in \mathrm{argmin}_{u_2 \in \mathbb{P}(A_2(s'))} H^2_{u_2, V_1^t}(J_1^t)(s', u_1')$ for $\phi_u \in \Phi_u$
        $\overline{\sigma}_{2,\phi,\phi_u}^{t+1} \leftarrow u_2', \ \ V_{2,\phi,\phi_u}^{t+1} \leftarrow H^2_{u_2', V_1^t}(J_1^t)(s', u_1)$ is linear in $u_1$
    return $\overline{\sigma}_2^{t+1} \leftarrow \langle \overline{\sigma}_{2,\phi,\phi_u}^{t+1} \rangle_{\theta \in \Theta_{\overline{\sigma}_2^{t+1}}}, V_2^{t+1} \leftarrow \langle V_{2,\phi,\phi_u}^{t+1} \rangle_{\theta \in \Theta_{\overline{\sigma}_2^{t+1}}}$
```

exists a constant-BFCP $\Phi_{\sigma_1^t}$ of S for σ_1^t. Since J_2^t is a CON-PWL Borel measurable function, there exists a BFCP $\Phi_{J_2^t}$ of S satisfying the properties of Definition 7.3 for J_2^t. Therefore $J_2^t(s, \sigma_1^t(s))$ is constant on each region in the BFCP $\Phi_{\sigma_1^t} + \Phi_{J_2^t}$. We can similarly show that $V_2^t(s, \sigma_1^t(s))$ is constant on each region in the BFCP $\Phi_{\sigma_1^t} + \Phi_{V_2^t}$, where $\Phi_{V_2^t}$ is a BFCP of S from Definition 7.3 for V_2^t. Consider the policy evaluation of Ag_1 (procedure *PE1*). Using Definition 7.1 we have that $J_1^{t+1} = H^1_{\sigma_1^t, V_2^t}(J_2^t)$ is constant on each region in the BFCP $\Phi_{\sigma_1^t} + \Phi_{J_2^t} + \Phi_{V_2^t}$, which also implies that J_1^{t+1} is Borel measurable. Since J_2^t and V_2^t are bounded, then J_1^{t+1} is also bounded. ∎

Lemma 7.8 (Improvement consistency for Ag_1). *If $J_2^t, V_2^t \in \mathbb{F}(\Lambda_1)$ are bounded, CON-PWL Borel measurable and an iteration of policy improvement of Ag_1 is performed (procedure PI1), then $\sigma_1^{t+1}(s) \in \mathrm{argmax}_{u_1 \in \mathbb{P}(A_1(s))} H^1_{u_1, V_2^t}(J_2^t)(s)$ is a PWC stochastic kernel, and $V_1^{t+1} = H^1_{\sigma_1^{t+1}, V_2^t}(J_2^t)$ is bounded, PWC Borel measurable.*

Proof. Suppose $J_2^t, V_2^t \in \mathbb{F}(\Lambda_1)$ are bounded, CON-PWL Borel measurable functions. Using [27, Chapter 18.1] and Definition 7.3 it follows that the function $K^t \in$

$\mathbb{F}(\Lambda_1)$ where $K^t(s,u_1) := \min\{J_2^t(s,u_1), V_2^t(s,u_1)\}$ is also a bounded, Borel measurable function for $(s,u_1) \in \Lambda_1$. Note that, over each region in $\Phi_{J_2^t} + \Phi_{V_2^t}$, $K^t(s,u_1)$ is constant in s given u_1, and PWL in u_1 given s ($\Phi_{J_2^t}$ and $\Phi_{V_2^t}$ from Lemma 7.7), and therefore K^t is CON-PWL.

Let $\Phi_{K^t} = \Phi_{J_2^t} + \Phi_{V_2^t}$ be a BFCP of S and Θ_{K^t} a BFCP of Λ_1 satisfying the properties of Definition 7.3 for K^t. Every state in each region of the BFCP Φ_{K^t} has the same set of available actions for Ag_1 and same strategy u_1 that maximises $K^t(s,u_1)$ on a region of Θ_{K^t}. Therefore, using the CON-1 solution in Definition 7.5, the strategy of Ag_1:

$$\sigma_1^{t+1}(s) \in \operatorname{argmax}_{u_1 \in \mathbb{P}(A_1(s))} H^1_{u_1, V_2^t}(J_2^t)(s)$$

is constant on each region in Φ_{K^t}, which also implies that σ_1^{t+1} is Borel measurable.

Since σ_1^{t+1} is a PWC stochastic kernel, then Lemma 7.7 implies that V_1^{t+1} is bounded, PWC Borel measurable. □

Lemma 7.9 (Evaluation consistency for Ag_2). *If $J_1^t, V_1^t \in \mathbb{F}(S)$ are bounded, PWC Borel measurable and $\overline{\sigma}_2^t \in \overline{\Sigma}_2$ is a CON-PWC stochastic kernel and an iteration of policy evaluation of Ag_2 is performed (procedure PE1), then $J_2^{t+1} = H^2_{\overline{\sigma}_2^t, V_1^t}(J_1^t)$ is bounded, CON-PWL Borel measurable.*

Proof. The proof follows similarly to Lemma 7.7. □

Lemma 7.10 (Improvement consistency for Ag_2). *If $J_1^t, V_1^t \in \mathbb{F}(S)$ are bounded, PWC Borel measurable and an iteration of policy improvement of Ag_2 is performed (procedure PI2), then $\overline{\sigma}_2^{t+1}(s,u_1) \in \operatorname{argmin}_{u_2 \in \mathbb{P}(A_2(s))} H^2_{u_2, V_1^t}(J_1^t)(s,u_1)$ is a CON-PWC stochastic kernel, and $V_2^{t+1} = H^2_{\overline{\sigma}_2^{t+1}, V_1^t}(J_1^t)$ is bounded, CON-PWL Borel measurable.*

Proof. The proof follows similarly to Lemma 7.8. □

By fusing Lemmas 7.7 to 7.10 we can prove that the strategies and value functions generated during each iteration of Algorithm 7.1 never leave a finitely representable class of functions. Formally, we have the following result.

Theorem 7.11 (Representation consistency). *In any iteration of the Minimax-action-free PI algorithm (see Algorithm 7.1), if*

(i) $J_1^t, V_1^t \in \mathbb{F}(S)$ are bounded, PWC Borel measurable and $\sigma_1^t \in \Sigma_1$ is a PWC stochastic kernel;

(ii) $J_2^t, V_2^t \in \mathbb{F}(\Lambda_1)$ are bounded, CON-PWL Borel measurable and $\overline{\sigma}_2^t \in \overline{\Sigma}_2$ is a CON-PWC stochastic kernel;

then so are J_1^{t+1}, V_1^{t+1}, σ_1^{t+1}, J_2^{t+1}, V_2^{t+1} and $\overline{\sigma}_2^{t+1}$ respectively, regardless of which one of the four iterations is performed.

Proof. The theorem directly follows from Lemmas 7.7 to 7.10. □

We next show that the optimization problem in the policy improvement of Ag_2 can be further simplified.

Corollary 7.12 (Pure strategy for Ag_2). *For the policy improvement of Ag_2 (procedure PI2), there exists a CON-PWC stochastic kernel:*

$$\overline{\sigma}_2^{t+1}(s,u_1) \in \operatorname{argmin}_{u_2 \in \mathbb{P}(A_2(s))} H^2_{u_2, V_1^t}(J_1^t)(s,u_1)$$

such that $\overline{\sigma}_2^{t+1}(a_2 \mid (s,u_1)) = 1$ for some $a_2 \in A_2$ over each region $\theta \in \Theta_{\overline{\sigma}_2^{t+1}}$, where $\Theta_{\overline{\sigma}_2^{t+1}}$ is a BFCP of Λ_1 for $\overline{\sigma}_2^{t+1}$ using Definition 7.4.

Proof. The result follows directly follows from the proof of Lemma 7.10. □

We next demonstrate that Algorithm 7.2 presents a way of constructing new BFCPs such that the strategies and value functions after one iteration of the Minimax-action-free PI algorithm remain constant, PWC, or PWL on each region of the constructed BFCPs.

Corollary 7.13 (BFCP iteration for PI). *After performing Algorithm 7.2 we have:*

(i) $\Phi_{J_1^{t+1}}$, $\Phi_{V_1^{t+1}}$ *and* $\Phi_{\sigma_1^{t+1}}$ *are constant-BFCPs of* S *for* $J_1^{t+1} = H^1_{\sigma_1^t, V_2^t}(J_2^t)$, $V_1^{t+1} = H^1_{\sigma_1^{t+1}, V_2^t}(J_2^t)$ *and* $\sigma_1^{t+1}(s) \in \operatorname{argmax}_{u_1 \in \mathbb{P}(A_1(s))} H^1_{u_1, V_2^t}(J_2^t)(s)$, *respectively;*

(ii) $\Theta_{J_2^{t+1}}$ *and* $\Theta_{V_2^{t+1}}$ *are BFCPs of* Λ_1 *for* $J_2^{t+1} = H^2_{\overline{\sigma}_2^t, V_1^t}(J_1^t)$ *and* $V_2^{t+1} = H^2_{\overline{\sigma}_2^{t+1}, V_1^t}(J_1^t)$ *meeting the conditions of Definition 7.3 respectively, and* $\Theta_{\overline{\sigma}_2^{t+1}}$ *is a BFCP of* Λ_1 *for* $\overline{\sigma}_2^{t+1}(s, u_1) \in \operatorname{argmin}_{u_2 \in \mathbb{P}(A_2(s))} H^2_{u_2, V_1^t}(J_1^t)(s, u_1)$ *meeting the conditions of Definition 7.4.*

Proof. The results follow directly from Lemmas 6.1 and 7.7 to 7.10. □

7.2. Convergence analysis and strategy computation. We next prove the convergence of the Minimax-action-free PI algorithm by showing that an operator related to the algorithm is a contraction mapping with a unique fixed point, one of whose components is the value function multiplied by a known constant. The proof closely follows the steps for finite state spaces given in [5], but is more complex due to the underlying infinite state space and the need to deal with the requirement of Borel measurable consistency and finite representation of strategies and value functions.

Given PWC $\sigma_1 \in \Sigma_1$ and CON-PWC $\overline{\sigma}_2 \in \overline{\Sigma}_2$ stochastic kernels, we define the operator $G_{\sigma_1, \overline{\sigma}_2} : (\mathbb{F}(S) \times \mathbb{F}(S) \times \mathbb{F}(\Lambda_1) \times \mathbb{F}(\Lambda_1)) \to (\mathbb{F}(S) \times \mathbb{F}(S) \times \mathbb{F}(\Lambda_1) \times \mathbb{F}(\Lambda_1))$ such that:

$$G_{\sigma_1, \overline{\sigma}_2}(J_1, V_1, J_2, V_2) := (M^1_{\sigma_1}(J_2, V_2), K^1(J_2, V_2), M^2_{\overline{\sigma}_2}(J_1, V_1), K^2(J_1, V_1)) \tag{7.1}$$

where $J_1, V_1 \in \mathbb{F}(S)$ are assumed to be PWC, $J_2, V_2 \in \mathbb{F}(\Lambda_1)$ are assumed to be CON-PWL, and the four operators $M^1_{\sigma_1}$, K^1, $M^2_{\overline{\sigma}_2}$ and K^2 represent the four iterations of the Minimax-action-free PI algorithm from lines 2 to 14, and are defined as follows.

- $M^1_{\sigma_1} : \mathbb{F}(\Lambda_1) \times \mathbb{F}(\Lambda_1) \to \mathbb{F}(S)$ corresponds to the policy evaluation of Ag_1 (procedure *PE1*) where for any $s \in S$:

  $$M^1_{\sigma_1}(J_2, V_2)(s) := H^1_{\sigma_1, V_2}(J_2)(s) \tag{7.2}$$

  and is bounded, PWC Borel measurable using Lemma 7.7.
- $K^1 : \mathbb{F}(\Lambda_1) \times \mathbb{F}(\Lambda_1) \to \mathbb{F}(S)$ corresponds to the policy improvement of Ag_1 (procedure *PI1*) and for any $s \in S$:

  $$K^1(J_2, V_2)(s) := \max\nolimits_{u_1 \in \mathbb{P}(A_1(s))} H^1_{u_1, V_2}(J_2)(s) \tag{7.3}$$

  and is bounded, PWC Borel measurable using Lemma 7.8.
- $M^2_{\overline{\sigma}_2} : \mathbb{F}(S) \times \mathbb{F}(S) \to \mathbb{F}(\Lambda_1)$ corresponds to the policy evaluation of Ag_2 (procedure *PE2*) and for any $(s, u_1) \in \Lambda_1$:

  $$M^2_{\overline{\sigma}_2}(J_1, V_1)(s, u_1) := H^2_{\overline{\sigma}_2, V_1}(J_1)(s, u_1) \tag{7.4}$$

  and is bounded, CON-PWL Borel measurable using Lemma 7.9.

- $K^2 : \mathbb{F}(S) \times \mathbb{F}(S) \to \mathbb{F}(\Lambda_1)$ corresponds to the policy improvement of Ag_2 (procedure *PI2*) and any $(s, u_1) \in \Lambda_1$:

$$K^2(J_1, V_1)(s, u_1) \coloneqq \min_{u_2 \in \mathbb{P}(A_2(s))} H^2_{u_2, V_1}(J_1)(s, u_1) \tag{7.5}$$

  and is bounded, CON-PWL Borel measurable using Lemma 7.10.

For both the spaces $\mathbb{F}(S) \times \mathbb{F}(S)$ and $\mathbb{F}(\Lambda_1) \times \mathbb{F}(\Lambda_1)$, we consider the norm:

$$\|(J, V)\| = \max\{\|J\|, \|V\|\}$$

and for the space $\mathbb{F}(S) \times \mathbb{F}(S) \times \mathbb{F}(\Lambda_1) \times \mathbb{F}(\Lambda_1)$ the norm:

$$\|(J_1, V_1, J_2, V_2)\| = \max\{\|J_1\|, \|V_1\|, \|J_2\|, \|V_2\|\}\,. \tag{7.6}$$

We next require the following properties of theses norms.

Lemma 7.14. *For any $J_1, V_1, J_1', V_1' \in \mathbb{F}(S)$ and $J_2, V_2, J_2', V_2' \in \mathbb{F}(\Lambda_1)$, we have:*

$$\begin{aligned} \|\max[J_1, V_1] - \max[J_1', V_1']\| &\leq \max\{\|J_1 - J_1'\|, \|V_1 - V_1'\|\} \\ \|\min[J_2, V_2] - \min[J_2', V_2']\| &\leq \max\{\|J_2 - J_2'\|, \|V_2 - V_2'\|\}\,. \end{aligned}$$

Proof. Consider any $J_1, V_1, J_1', V_1' \in \mathbb{F}(S)$. The sup-norm for the space $\mathbb{F}(S)$ implies that for every $s \in S$:

$$J_1(s) \leq J_1'(s) + \max\{\|J_1 - J_1'\|, \|V_1 - V_1'\|\} \tag{7.7}$$

$$V_1(s) \leq V_1'(s) + \max\{\|J_1 - J_1'\|, \|V_1 - V_1'\|\} \tag{7.8}$$

from which we have:

$$\max\{J_1(s), V_1(s)\} \leq \max\{J_1'(s), V_1'(s)\} + \max\{\|J_1 - J_1'\|, \|V_1 - V_1'\|\}\,. \tag{7.9}$$

Exchanging (J_1, V_1) with (J_1', V_1') in (7.7) and (7.8) derives an inequality similar to (7.9), and combining it with (7.9) leads to the inequality:

$$|\max\{J_1(s), V_1(s)\} - \max\{J_1'(s), V_1'(s)\}| \leq \max\{\|J_1 - J_1'\|, \|V_1 - V_1'\|\}$$

implying that:

$$\sup_{s \in S} |\max\{J_1(s), V_1(s)\} - \max\{J_1'(s), V_1'(s)\}| \leq \max\{\|J_1 - J_1'\|, \|V_1 - V_1'\|\}\,. \tag{7.10}$$

Since J_1, V_1, J_1' and V_1' are bounded, Borel measurable, so is $\max[J_1, V_1] - \max[J_1', V_1']$ by [27, Chapter 18.1], i.e., $\max[J_1, V_1] - \max[J_1', V_1'] \in \mathbb{F}(S)$. Thus, by (7.10):

$$\|\max[J_1, V_1] - \max[J_1', V_1']\| \leq \max\{\|J_1 - J_1'\|, \|V_1 - V_1'\|\}.$$

The second inequality of the lemma can be proved following the same steps for $J_2, V_2, J_2', V_2' \in \mathbb{F}(\Lambda_1)$. □

Using the above operators and results, we are now in position to prove the convergence of the Minimax-action-free PI algorithm.

Theorem 7.15 (Convergence guarantee). *Suppose that each of the four iterations of the Minimax-action-free PI algorithm from lines 2 to 14 is performed infinitely often. Then the sequence $\langle \gamma V_1^t \rangle_{t=0}^{\infty}$ generated by the algorithm converges to $V^\star$.*

Proof. We prove that each component of the operator $G_{\sigma_1,\overline{\sigma}_2}$ satisfies a contraction property. Suppose that $J_1, V_1, J_1', V_1' \in \mathbb{F}(S)$ are PWC and $J_2, V_2, J_2', V_2' \in \mathbb{F}(\Lambda_1)$ are CON-PWL.

- For $M^1_{\sigma_1}$, since $M^1_{\sigma_1}(J_2, V_2) - M^1_{\sigma_1}(J_2', V_2') \in \mathbb{F}(S)$ by [27, Chapter 18.1] and by Definition 7.1 and the definition of the sup-norm for $\mathbb{F}(S)$ we have:

$$
\begin{aligned}
&(7.11)\quad \|M^1_{\sigma_1}(J_2, V_2) - M^1_{\sigma_1}(J_2', V_2')\| \\
&= \sup_{s\in S}\left|\frac{1}{\gamma}\min\{J_2(s,\sigma_1(s)), V_2(s,\sigma_1(s))\} - \frac{1}{\gamma}\min\{J_2'(s,\sigma_1(s)), V_2'(s,\sigma_1(s))\}\right| \\
&\leq \frac{1}{\gamma}\sup_{(s,u_1)\in\Lambda_1}\big|\min\{J_2(s,u_1), V_2(s,u_1)\} - \min\{J_2'(s,u_1), V_2'(s,u_1)\}\big| \\
&\qquad\qquad\qquad \text{rearranging and since } \{(s,\sigma_1(s)) \mid s\in S\} \subseteq \Lambda_1 \\
&= \frac{1}{\gamma}\big\|\min[J_2, V_2] - \min[J_2', V_2']\big\| \\
&\qquad\qquad \text{since } \min[J_2, V_2] - \min[J_2', V_2'] \in \mathbb{F}(\Lambda_1) \text{ using [27, Chapter 18.1]} \\
&\leq \frac{1}{\gamma}\max\{\|J_2 - J_2'\|, \|V_2 - V_2'\|\} \qquad\qquad \text{by Lemma 7.14} \\
&\leq \frac{1}{\gamma}\max\{\|J_1 - J_1'\|, \|V_1 - V_1'\|, \|J_2 - J_2'\|, \|V_2 - V_2'\|\}\,.
\end{aligned}
$$

- For K^1, since $K^1(J_2, V_2) - K^1(J_2', V_2') \in \mathbb{F}(S)$ by Definition 7.1 and definition of the sup-norm for $\mathbb{F}(S)$:

$$
\begin{aligned}
&(7.12)\quad \|K^1(J_2, V_2) - K^1(J_2', V_2')\| \\
&= \sup_{s\in S}\left|\max_{u_1\in\mathbb{P}(A_1(s))}\frac{1}{\gamma}\min\{J_2(s,u_1), V_2(s,u_1)\}\right. \\
&\qquad\left. - \max_{u_1\in\mathbb{P}(A_1(s))}\frac{1}{\gamma}\min\{J_2'(s,u_1), V_2'(s,u_1)\}\right| \\
&\leq \frac{1}{\gamma}\sup_{(s,u_1)\in\Lambda_1}\big|\min\{J_2(s,u_1), V_2(s,u_1)\} - \min\{J_2'(s,u_1), V_2'(s,u_1)\}\big| \\
&\qquad\qquad \text{rearranging and since } \{(s,u_1) \mid u_1\in\mathbb{P}(A_1(s))\} \subseteq \Lambda_1 \\
&\leq \frac{1}{\gamma}\max\{\|J_1 - J_1'\|, \|V_1 - V_1'\|, \|J_2 - J_2'\|, \|V_2 - V_2'\|\}
\end{aligned}
$$

  where the final inequality follows from similar arguments used in (7.11).
- For $M^2_{\overline{\sigma}_2}$, since $M^2_{\overline{\sigma}_2}(J_1, V_1) - M^2_{\overline{\sigma}_2}(J_1', V_1') \in \mathbb{F}(\Lambda_1)$ by Definition 7.2 and the sup-norm for $\mathbb{F}(\Lambda_1)$ we have:

$$
\begin{aligned}
&(7.13)\quad \|M^2_{\overline{\sigma}_2}(J_1, V_1) - M^2_{\overline{\sigma}_2}(J_1', V_1')\| \\
&= \sup_{(s,u_1)\in\Lambda_1}\left|\sum_{a_1\in A_1(s)}\sum_{a_2\in A_2(s)}\big(Q(s,(a_1,a_2)),\right. \\
&\qquad\left.\gamma\max[J_1, V_1]) - Q(s,(a_1,a_2),\gamma\max[J_1', V_1'])\big)u_1(a_1)\overline{\sigma}_2(a_2 \mid (s,u_1))\right| \\
&= \sup_{(s,u_1)\in\Lambda_1}\left|\sum_{(a_1,a_2)\in A(s)}\gamma\beta\sum_{s'\in\Theta(s,(a_1,a_2))}\delta(s,(a_1,a_2))(s')\right.
\end{aligned}
$$

$$\begin{aligned}
&\qquad \left.\big(\max\{J_1(s'),V_1(s')\} - \max\{J_1'(s'),V_1'(s')\}\big)u_1(a_1)\overline{\sigma}_2(a_2 \mid (s,u_1))\right| \\
&\qquad\qquad \text{rearranging, by Definition 5.4 and the sup-norm for } \mathbb{F}(\Lambda_1) \\
\le\ & \gamma\beta \sup_{(s,u_1)\in\Lambda_1} \sum_{(a_1,a_2)\in A(s)} \sum_{s'\in\Theta(s,(a_1,a_2))} \delta(s,(a_1,a_2))(s') \\
& \big|\max\{J_1(s'),V_1(s')\} - \max\{J_1'(s'),V_1'(s')\}\big|\, u_1(a_1)\overline{\sigma}_2(a_2 \mid (s,u_1)) \\
&\qquad\qquad \text{rearranging and since } \delta,\ u_1 \text{ and } \overline{\sigma}_2 \text{ are non-negative} \\
\le\ & \gamma\beta \sup_{(s,u_1)\in\Lambda_1} \sum_{(a_1,a_2)\in A(s)} \sum_{s'\in\Theta(s,(a_1,a_2)} \delta(s,(a_1,a_2))(s') \\
& \sup_{s''\in S} \big|\max\{J_1(s''),V_1(s'')\} - \max\{J_1'(s''),V_1'(s'')\}\big|\, u_1(a_1)\overline{\sigma}_2(a_2 \mid (s,u_1)) \\
&\qquad\qquad \text{since } f(s') \le \textstyle\sup_{s''\in S} f(s'') \text{ for any } f \in \mathbb{F}(S) \\
=\ & \gamma\beta \sup_{s''\in S} \big|\max\{J_1(s''),V_1(s'')\} - \max\{J_1'(s''),V_1'(s'')\}\big| \\
& \sup_{(s,u_1)\in\Lambda_1} \sum_{(a_1,a_2)\in A(s)} \sum_{s'\in\Theta(s,(a_1,a_2))} \delta(s,(a_1,a_2))(s')u_1(a_1)\overline{\sigma}_2(a_2 \mid (s,u_1)) \\
&\qquad\qquad \text{rearranging} \\
=\ & \gamma\beta \sup_{s''\in S} \big|\max\{J_1(s''),V_1(s'')\} - \max\{J_1'(s''),V_1'(s'')\}\big| \\
&\qquad\qquad \text{since } \delta \in \mathbb{P}(S\times A),\ u_1 \in \mathbb{P}(A_1) \text{ and } \bar{\sigma}_2 \in \mathbb{P}(A_2 \mid \Lambda_1) \\
=\ & \gamma\beta \,\big\|\max[J_1,V_1] - \max[J_1',V_1']\big\| \\
&\qquad\qquad \text{since } \max[J_1,V_1] - \max[J_1',V_1'] \in \mathbb{F}(S) \\
\le\ & \gamma\beta \max\{\|J_1 - J_1'\|, \|V_1 - V_1'\|\} \qquad\qquad \text{by Lemma 7.14} \\
\le\ & \gamma\beta \max\{\|J_1 - J_1'\|, \|V_1 - V_1'\|, \|J_2 - J_2'\|, \|V_2 - V_2'\|\}
\end{aligned}$$

- For K^2, since $K^2(J_1,V_1) - K^2(J_1',V_1') \in \mathbb{F}(\Lambda_1)$, by definition of the sup-norm for $\mathbb{F}(\Lambda_1)$ we have:

$$\begin{aligned}
& \|K^2(J_1,V_1) - K^2(J_1',V_1')\| \\
=\ & \sup_{(s,u_1)\in\Lambda_1} \left| \min_{u_2\in\mathbb{P}(A_2(s))} \sum_{a_1\in A_1(s)} \sum_{a_2\in A_2(s)} \gamma\beta \sum_{s'\in\Theta(s,(a_1,a_2))} \delta(s,(a_1,a_2))(s') \right. \\
& \left. \big(\max\{J_1(s'),V_1(s')\} - \max\{J_1'(s'),V_1'(s')\}\big)\, u_1(a_1)u_2(a_2) \right| \\
\le\ & \gamma\beta \sup_{(s,u_1)\in\Lambda_1} \min_{u_2\in\mathbb{P}(A_2(s))} \sum_{(a_1,a_2)\in A(s)} \sum_{s'\in\Theta(s,(a_1,a_2))} \delta(s,(a_1,a_2))(s') \\
& \big|\max\{J_1(s'),V_1(s')\} - \max\{J_1'(s'),V_1'(s')\}\big|\, u_1(a_1)u_2(a_2) \qquad \text{rearranging} \\
\le\ & \gamma\beta \max\{\|J_1 - J_1'\|, \|V_1 - V_1'\|, \|J_2 - J_2'\|, \|V_2 - V_2'\|\}
\end{aligned} \tag{7.14}$$

where the final inequality follows from similar arguments used in (7.13).

Next we prove that $G_{\sigma_1,\overline{\sigma}_2}$ is a contraction mapping using the above inequalities. More precisely, by definition, see (7.1), we have:

$$\begin{aligned}
& \big\|G_{\sigma_1,\overline{\sigma}_2}(J_1,V_1,J_2,V_2) - G_{\sigma_1,\overline{\sigma}_2}(J_1',V_1',J_2',V_2')\big\| \\
=\ & \big\|(M^1_{\sigma_1}(J_2,V_2) - M^1_{\sigma_1}(J_2',V_2'), K^1(J_2,V_2) - K^1(J_2',V_2'), \\
& \quad M^2_{\overline{\sigma}_2}(J_1,V_1) - M^2_{\overline{\sigma}_2}(J_1',V_1'), K^2(J_1,V_1) - K^2(J_1',V_1'))\big\|
\end{aligned}$$

$$
\begin{aligned}
&= \max\Big\{\big\|M^1_{\sigma_1}(J_2,V_2)-M^1_{\sigma_1}(J_2',V_2')\big\|,\big\|K^1(J_2,V_2)-K^1(J_2',V_2')\big\|,\\
&\qquad \big\|M^2_{\overline{\sigma}_2}(J_1,V_1)-M^2_{\overline{\sigma}_2}(J_1',V_1')\big\|,\big\|K^2(J_1,V_1)-K^2(J_1',V_1')\big\|\Big\} \qquad \text{rearranging}\\
&\le \max\left\{\frac{1}{\gamma},\gamma\beta\right\}\max\{\|J_1-J_1'\|,\|V_1-V_1'\|,\|J_2-J_2'\|,\|V_2-V_2'\|\}
\end{aligned}
$$

where the final inequality follows from (7.11)–(7.14).

Therefore, since $\max\{\frac{1}{\gamma},\gamma\beta\}<1$ and assuming σ_1 is PWC and $\overline{\sigma}_2$ is CON-PWC, we have that $G_{\sigma_1,\overline{\sigma}_2}$ is a contraction mapping for all $(\sigma_1,\overline{\sigma}_2)\in\Sigma_1\times\overline{\Sigma}_2$. Now since $\mathbb{F}(S)\times\mathbb{F}(S)\times\mathbb{F}(\Lambda_1)\times\mathbb{F}(\Lambda_1)$ is a complete metric space with respect to the norm (7.6), we conclude that $G_{\sigma_1,\overline{\sigma}_2}$ has a unique fixed point $(J_1^\star,V_1^\star,J_2^\star,V_2^\star)$. In view of (7.2)–(7.5), this fixed point satisfies for each $(s,u_1)\in\Lambda_1$:

$$
\begin{aligned}
&(7.15)\qquad J_1^\star(s)=\frac{1}{\gamma}\min\{J_2^\star(s,\sigma_1(s)),V_2^\star(s,\sigma_1(s))\}\\
&(7.16)\qquad V_1^\star(s)=\max_{u_1\in\mathbb{P}(A_1(s))}\frac{1}{\gamma}\min\{J_2^\star(s,u_1),V_2^\star(s,u_1)\}\\
&(7.17)\quad J_2^\star(s,u_1)=\sum_{a_1\in A_1(s)}\sum_{a_2\in A_2(s)}Q(s,(a_1,a_2)\gamma\max[J_1^\star,V_1^\star])u_1(a_1)\overline{\sigma}_2(a_2\mid(s,u_1))\\
&(7.18)\quad V_2^\star(s,u_1)=\min_{u_2\in\mathbb{P}(A_2(s))}\sum_{a_1\in A_1(s)}\sum_{a_2\in A_2(s)}\\
&\qquad\qquad Q(s,(a_1,a_2),\gamma\max[J_1^\star,V_1^\star])u_1(a_1)u_2(a_2).
\end{aligned}
$$

By combining (7.15) and (7.16), and combining (7.17) and (7.18), we have for each $(s,u_1)\in\Lambda_1$:

$$
\begin{aligned}
J_1^\star(s)&\le V_1^\star(s),\\
J_2^\star(s,u_1)&\ge V_2^\star(s,u_1)
\end{aligned}
$$

from which (7.16) and (7.18) can be simplified to:

$$
\begin{aligned}
V_1^\star(s)&=\max_{u_1\in\mathbb{P}(A_1(s))}\frac{1}{\gamma}V_2^\star(s,u_1)\\
V_2^\star(s,u_1)&=\min_{u_2\in\mathbb{P}(A_2(s))}\sum_{a_1\in A_1(s)}\sum_{a_2\in A_2(s)}Q(s,(a_1,a_2),\gamma V_1^\star)u_1(a_1)u_2(a_2)
\end{aligned}
$$

implying that:

$$
\begin{aligned}
\gamma V_1^\star(s)&=\max_{u_1\in\mathbb{P}(A_1(s))}\min_{u_2\in\mathbb{P}(A_2(s))}\sum_{a_1\in A_1(s)}\sum_{a_2\in A_2(s)}Q(s,(a_1,a_2),\gamma V_1^\star)u_1(a_1)u_2(a_2)\\
&=T(\gamma V_1^\star)\,.
\end{aligned}
$$

Thus, we have $\gamma V_1^\star=V^\star$, which completes the proof. □

Next we compute the strategies for the agents based on the function returned by the Minimax-action-free PI algorithm.

Definition 7.16 (CON-3 solution). *Let $f\in\mathbb{F}(\Lambda_{12})$. If there exists a BFCP Φ of S where, for each region $\phi\in\Phi$, $A(s)=A(s')$ for all $s,s'\in\phi$, and there exists a pair of probability measures $u_1^\phi\in\mathbb{P}(A_1(s))$ and $u_2^\phi\in\mathbb{P}(A_2(s))$ for all $s\in\phi$ such that $f(s,u_1^\phi,u_2^\phi)=\max_{u_1\in\mathbb{P}(A_1(s))}\min_{u_2\in\mathbb{P}(A_2(s))}f(s,u_1,u_2)$ for all $s\in\phi$, and if $\sigma_1\in\Sigma_1$, $\sigma_2\in\Sigma_2$ are such that $\sigma_1(s)=u_1^\phi$ and $\sigma_2(s)=u_2^\phi$ for all $s\in\phi$, then (σ_1,σ_2) is a constant-3 (CON-3) solution of f over ϕ.*

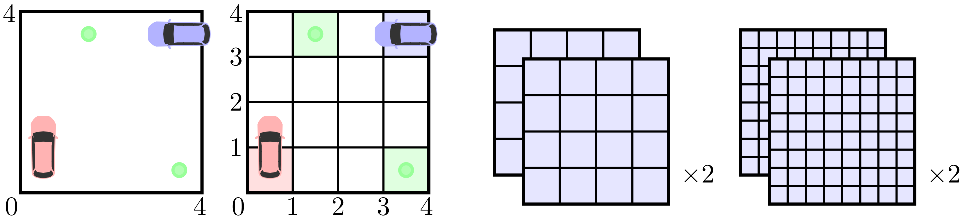

FIG. 3. *Two-vehicle dynamic parking. Left: continuous environment and 4×4 discrete percepts of the agents. Right: sum of perception and reward FCPs: $\Phi_P + \sum_{\alpha \in A} \Phi_R^\alpha$ (4-by-4 grid) and converged BFCP (8-by-8 grid).*

LEMMA 7.17 (PWC strategies). *Let V be a function returned by Minimax-action-free PI. If $(\sigma_1, \sigma_2) \in \Sigma$ achieves the maximum and the minimum in Definition 5.4 for V and all $s \in S$ via a CON-3 solution, then σ_1 and σ_2 are PWC stochastic kernels.*

Proof. By Theorems 7.11 and 7.15, V is bounded, PWC Borel measurable. Now for any $\alpha \in A$, $Q(\cdot, \alpha, V) : S \to \mathbb{R}$ is bounded, PWC Borel measurable by Lemma 6.1. Let Φ_Q be a BFCP of S such that $Q(\,\cdot\,, \alpha, V)$ is constant on each region in Φ_Q for all $\alpha \in A$, and Φ_A be a BFCP of S such that $A(s)$ is constant on each region in Φ_A. Then, given $u_1 \in \mathbb{P}(A_1(s))$ and $u_2 \in \mathbb{P}(A_2(s))$, the function $Q'(\,\cdot\,, u_1, u_2) : S \to \mathbb{R}$ where for any $s \in S$:

$$Q'(s, u_1, u_2) = \sum_{a_1 \in A_1(s)} \sum_{a_2 \in A_2(s)} Q(s, (a_1, a_2), V) u_1(a_1) u_2(a_2)$$

is constant in each region of $\Phi_Q + \Phi_A$. Therefore there exists a CON-3 solution (σ_1, σ_2) of $Q'(s, u_1, u_2)$ and since $\Phi_Q + \Phi_A$ is a BFCP, the result follows. □

8. Case study.

To illustrate our approach, we build a small NS-CSG model of a dynamic vehicle parking problem (a static version is considered in [3]). We then synthesise strategies for it using an implementation of the VI algorithm of Section 6.

8.1. NS-CSG modelling.

Figure 3 (left) shows two agents (vehicles) $N = \{1, 2\}$ and two parking spots $M = \{m_1, m_2\}$ in a (continuous) rectangular region $\mathcal{R} = \{(x, y) \in \mathbb{R}^2 \mid 0 \leq x, y \leq 4\}$ representing the states of the environment. The coordinates of these parking spots are $w_{m_1} = (1.5, 3.5)$ and $w_{m_2} = (3.5, 0.5)$, which are assumed to be known to agents. The zero-sum objective is for Ag_1 to park at a chosen parking spot without crashing into Ag_2, while Ag_2 seeks to prevent Ag_1 from parking. Suppose that the two agents can start from any positions in $\mathcal{R}$ and have the same speed. The agents' percepts are implemented via the same linear regression model for multi-class classification, see Figure 3 (left). The actions of the agents are to move either up, down, left or right, or park.

Formally, the agents and the environment are defined as follows.

- The local state spaces of the agents are given by $Loc_1 = \{w_{m_1}, w_{m_2}\}$ and $Loc_2 = \{loc_2\}$, where the local state of Ag_1 is the coordinate of the parking spot it aims to approach and the local state of Ag_2 is a dummy state. Here we work with integer percepts and the set of percepts of Ag_i for $1 \leq i \leq 2$ is given by $Per_i = \{1, 2, 3, 4\}^2 \times \{1, 2, 3, 4\}^2$ and the discrete coordinates the agent perceives are the positions of the two vehicles.
- The set of environment states is $S_E = \mathcal{R} \times \mathcal{R}$ and the environment is in state $s_E = (w_1, w_2)$ if $w_i \in \mathcal{R}$ is the continuous coordinate vector of Ag_i.

- The action set of Ag_i for $1 \leq i \leq 2$ is given by $A_i = \{u_i, d_i, l_i, r_i, p_i\}$ representing the four possible directions an agent can choose to move and park.
- The action availability function of Ag_i for $1 \leq i \leq 2$ is given by:

$$\Delta_i(loc_i, (\bar{w}_1, \bar{w}_2)) = \begin{cases} \{u_i, d_i, l_i, r_i, p_i\} & \text{if } \bar{w}_i \in \{\bar{w}_{m_1}, \bar{w}_{m_2}\} \\ \{u_i, d_i, l_i, r_i\} & \text{otherwise.} \end{cases}$$

- In line with the percepts, the observation functions of the agents are both implemented via the same linear regression model for multi-class classification with classifier boundaries given by

$$\textstyle\bigcup_{\ell \in \{1,2,3\}} \big(\{(x,y) \in \mathcal{R} \mid x = \ell\} \cup \{(x,y) \in \mathcal{R} \mid y = \ell\}\big) .$$

  The Borel measurable tie-breaking rule used here is assigning boundary points to the left and lower discrete coordinate, e.g., the class of environment state $(2, 3, 3.1, 1.7)$ is $(2, 3, 4, 2)$. We denote by $\bar{w}_i$ the coordinate of the class of w_i under obs_i.
- For any $s_1 = (z_1, (\bar{w}_1, \bar{w}_2)) \in S_1$ and $\alpha = (a_1, a_2)$ to define δ_1 we have the following two cases to consider:
  - if $\|\bar{z}_1 - \bar{w}_1\|_2 > \|\bar{z}_1 - \bar{w}_2\|_2$, i.e. Ag_2 is closer to the chosen parking spot of Ag_1, and the joint action (a_1, a_2) indicate both agents are trying to approach z_1, then $\delta_1(s_1, \alpha)(w_{m_j}) = 0.5$ for $1 \leq j \leq 2$, i.e., Ag_1 changes its chosen parking spot with probability 0.5;
  - otherwise $\delta_1(s_1, \alpha)(z_1) = 1$, i.e. Ag_1 sticks with its chosen parking spot.
- Since $Loc_2 = \{loc_2\}$, for any $s_2 = (loc_2, (\bar{w}_1, \bar{w}_2)) \in S_2$ and $\alpha = (a_1, a_2)$ we let $\delta_2(s_2, \alpha) = loc_2$.
- For any $(w_1, w_2) \in S_E$ and $(a_1, a_2) \in A$, we let $\delta_E((w_1, w_2), (a_1, a_2)) = (w_1', w_2')$ where for $1 \leq i \leq 2$

$$w_i' = \begin{cases} w_i + d_{a_i}\Delta t & \text{if } (w_i + d_{a_i}\Delta t) \in \mathcal{R} \\ w_i & \text{otherwise} \end{cases}$$

  d_{a_i} is the direction of movement of the action a_i, e.g., $d_{u_i} = (0, 1)$ and $d_{l_i} = (-1, 0)$ and $\Delta t = 0.5$ is the time step.

We consider the reward structure where all action rewards are 0, there is a negative reward in states where the Ag_1 has yet to reach its current parking spot and the agents crash (that is, $r_S(s) = -1000$ if $\bar{w}_1 = \bar{w}_2 \neq \bar{z}_1$), a positive reward in states where the Ag_1 has reached its parking spot which increases if the agents have not crashed (that is, $r_S(s) = 500$ if $\bar{w}_1 = \bar{w}_2 = \bar{z}_1$ and $r_S(s) = 1000$ if $\bar{w}_1 = \bar{z}_1$ and $\bar{w}_1 \neq \bar{w}_2$), and otherwise the state reward is 0. We fix the discount factor $\beta = 0.6$.

8.2. Strategy synthesis. We implement our VI algorithm (see Algorithm 6.1) using a polyhedra representation of regions. We have partitioned the state space of the game into two sets corresponding to the two possible local states of Ag_1.

The VI algorithm converges after 46 iterations when $\varepsilon = 10^{-6}$ and takes 3,825s to complete. For each set in the partition of the state space, the BFCP of this set converges to the product of two 8×8 grids, see Figure 3 (right). For the current chosen parking spot of Ag_1 (red square) and coordinate of Ag_2 (purple triangle), the value function with respect to the coordinate of Ag_1 is presented in Figure 4 (left) and shows that, the closer Ag_1 is to its chosen parking spot, the higher the optimal value. The lightest-colour class is caused by an immediate crash, and its position follows from the observation function.

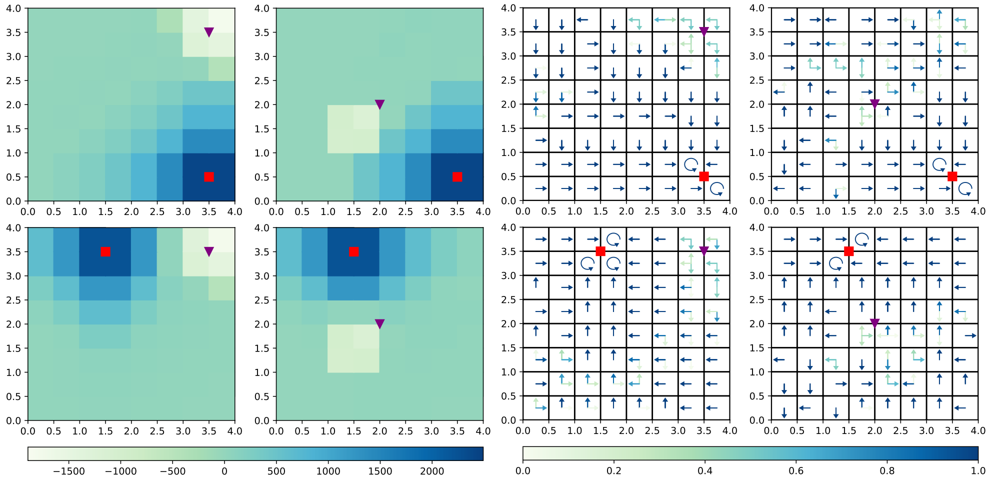

Fig. 4. *Value function (left) and optimal strategy (right) over different coordinates of* Ag_1 *for a fixed local state of* Ag_1 *(red square) and a fixed coordinate of* Ag_2 *(purple triangle).*

An (approximate) optimal strategy for Ag_1 is presented in Figure 4 (right), where the colour of an arrow is proportional to the probability of moving in that direction and the rotating arrow represents the parking action. There are several interesting choices which are not intuitive. For example, although a crash cannot be avoided before reaching its current parking spot, Ag_1 moves left when in $[1.0, 1.5] \times [3.5, 4.0]$ (top left) as it is better to crash later in a discounted setting and Ag_1 moves right when in $[1.5, 2.0] \times [0.5, 1.0]$ (down right) since by moving in this direction it will meet the conditions to (randomly) update its chosen parking spot required by Ag_1's local transition function.

9. Conclusions. We have proposed a novel modelling formalism called neuro-symbolic concurrent stochastic games (NS-CSGs) for representing probabilistic finite-state agents with neural network perception mechanisms interacting in a shared, continuous-state environment. We proved the existence of the value of NS-CSGs under Borel measurability and piecewise constant restrictions, and presented the first implementable policy iteration and value iteration algorithms for computing the values and strategies of NS-CSGs with respect to zero-sum discounted cumulative rewards, assuming a fully observable setting. We illustrated our approach by modelling a dynamic vehicle parking problem as an NS-CSG and synthesising optimal values and strategies using value iteration.

A number of improvements and challenges remain as future work. For example, in value iteration, the number of regions increases exponentially as the number of dimensions in the environment's state space increases, which directly influences computation costs in terms of both time and space. In addition, generalising to other observation functions, e.g., using Sigmoid NNs, will necessitate working with abstractions in order to represent regions with a more general shape. There is also work to be done in devising practical stopping criteria for iterative methods over BFCPs. Next, we plan to investigate extending the algorithms to allow for partial observability, as discussed in Section 3, and nonzero-sum objectives through equilibria-based properties [16]. For the latter, preliminary progress has been made in [35], building upon the NS-CSG model presented in this paper.

REFERENCES

[1] M. E. Akintunde, E. Botoeva, P. Kouvaros, and A. Lomuscio, *Verifying Strategic Abilities of Neural-symbolic Multi-agent Systems*, in Proc. 17th Int. Conf. Principles of Knowledge Representation and Reasoning (KR'20), IJCAI Organization, 9 2020, pp. 22–32.

[2] G. Anderson, A. Verma, I. Dillig, and S. Chaudhuri, *Neurosymbolic reinforcement learning with formally verified exploration*, in Proc. 33rd Int. Conf. Advances in Neural Information Processing Systems (NeurIPS'20), Curran Associates, Inc., 2020.

[3] D. Ayala, O. Wolfson, B. Xu, B. Dasgupta, and J. Lin, *Parking slot assignment games*, in Proc. 19th ACM SIGSPATIAL Int. Conf. Advances in Geographic Information Systems (GIS'11), ACM, 2011, p. 299–308.

[4] A. Basu and Ł. Stettner, *Zero-sum Markov games with impulse controls*, SIAM Journal on Control and Optimization, 58 (2020), pp. 580–604.

[5] D. P. Bertsekas, *Distributed asynchronous policy iteration for sequential zero-sum games and minimax control*, arXiv:2107.10406, (2021).

[6] D. P. Bertsekas and H. Yu, *Q-learning and enhanced policy iteration in discounted dynamic programming*, Mathematics of Operations Research, 37 (2012), pp. 66–94.

[7] N. Brown, A. Bakhtin, A. Lerer, and Q. Gong, *Combining deep reinforcement learning and search for imperfect-information games*, in Proc. 33rd Int. Conf. Advances in Neural Information Processing Systems (NeurIPS'20), Curran Associates, Inc., 2020.

[8] A. Cosso, *Stochastic differential games involving impulse controls and double-obstacle quasi-variational inequalities*, SIAM Journal on Control and Optimization, 51 (2013), pp. 2102–2131.

[9] J. Filar and K. Vrieze, *Competitive Markov decision processes*, Springer, 1997.

[10] J. Gupta, M. Egorov, and M. Kochenderfer, *Cooperative multi-agent control using deep reinforcement learning*, in Proc. 16th Int. Conf. Autonomous Agents and Multiagent Systems (AAMAS'17), Springer, 2017, pp. 66–83.

[11] O. Hernández-Lerma and J. Lasserre, *Zero-sum stochastic games in borel spaces: average payoff criteria*, SIAM Journal on Control and Optimization, 39 (2000), pp. 1520–1539.

[12] A. J. Hoffman and R. M. Karp, *On non-terminating stochastic games*, Management Science, 12 (1966), pp. 359–370.

[13] I. Hogeboom-Burr and S. Yuksel, *Comparison of information structures for zero-sum games and a partial converse to Blackwell ordering in standard borel spaces*, SIAM Journal on Control and Optimization, 59 (2021), pp. 1781–1803.

[14] V. Kovařík, M. Schmid, N. Burch, M. Bowling, and V. Lisý, *Rethinking formal models of partially observable multiagent decision making*, Artificial Intelligence, 303 (2022), p. 103645.

[15] P. R. Kumar and T.-H. Shiau, *Existence of value and randomized strategies in zero-sum discrete-time stochastic dynamic games*, SIAM Journal on Control and Optimization, 19 (1981), pp. 617–634.

[16] M. Kwiatkowska, G. Norman, D. Parker, and G. Santos, *Automatic verification of concurrent stochastic systems*, Formal Methods in System Design, (2021), pp. 1–63.

[17] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, *Multi-agent actor-critic for mixed cooperative-competitive environments*, in Proc. 31st Int. Conf. Neural Information Processing Systems (NIPS'17), Curran Associates Inc., 2017, p. 6382–6393.

[18] A. Maitra and T. Parthasarathy, *On stochastic games*, Journal of Optimization Theory and Applications, 5 (1970), pp. 289–300.

[19] A. S. Nowak, *Universally measurable strategies in zero-sum stochastic games*, The Annals of Probability, 13 (1985), pp. 269–287.

[20] A. S. Nowak, *Optimal strategies in a class of zero-sum ergodic stochastic games*, Mathematical methods of operations research, 50 (1999), pp. 399–419.

[21] K. R. Parthasarathy, *Probability measures on metric spaces*, American Mathematical Soc., 1967.

[22] J. Perolat, B. Scherrer, B. Piot, and O. Pietquin, *Approximate dynamic programming for two-player zero-sum Markov games*, in Proc. Int. Conf. Machine Learning, PMLR, 2015, pp. 1321–1329.

[23] M. A. Pollatschek and B. Avi-Itzhak, *Algorithms for stochastic games with geometrical interpretation*, Management Science, 15 (1969), pp. 399–415.

[24] L. D. Raedt, S. Dumancic, R. Manhaeve, and G. Marra, *From statistical relational to neural-symbolic artificial intelligence*, in Proc. 29th Int. Joint Conf. Artificial Intelligence (IJCAI'20), IJCAI Organization, 07 2020, pp. 4943–4950.

[25] J. H. Reif, *Universal games of incomplete information*, in Proc. ACM Symp. Theory of Com-

puting (STOC'79), ACM, 1979, pp. 288–308.

[26] J. H. Reif, *The complexity of two-player games of incomplete information*, Journal of Computer and System Sciences, 29 (1984), pp. 274–301.

[27] H. L. Royden and P. Fitzpatrick, *Real analysis (fourth edition)*, Macmillan New York, 2010.

[28] S. Shalev-Shwartz, S. Shammah, and A. Shashua, *Safe, multi-agent, reinforcement learning for autonomous driving*, arXiv:1610.03295, (2016).

[29] L. S. Shapley, *Stochastic games*, Proc. National Academy of Sciences, 39 (1953), pp. 1095–1100.

[30] D. Silver, A. Huang, C. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, et al., *Mastering the game of go with deep neural networks and tree search*, Nature, 529 (2016), pp. 484–489.

[31] B. Tolwinski, *Newton-type methods for stochastic games*, in Differential games and applications, Springer, 1989, pp. 128–144.

[32] J. Van Der Wal, *Discounted Markov games: Generalized policy iteration method*, Journal of Optimization Theory and Applications, 25 (1978), pp. 125–138.

[33] J. von Neumann, O. Morgenstern, H. Kuhn, and A. Rubinstein, *Theory of Games and Economic Behavior*, Princeton University Press, 1944.

[34] R. Yan, X. Duan, Z. Shi, Y. Zhong, J. Marden, and F. Bullo, *Policy evaluation and seeking for multi-agent reinforcement learning via best response*, IEEE Transactions on Automatic Control, 67 (2022), pp. 1898–1913.

[35] R. Yan, G. Santos, X. Duan, D. Parker, and M. Kwiatkowska, *Finite-horizon equilibria for neuro-symbolic concurrent stochastic games*, in Proc. 38th Conf. Uncertainty in Artificial Intelligence (UAI'22), AUAI Press, 2022.

[36] H. Yu, *On convergence of value iteration for a class of total cost Markov decision processes*, SIAM Journal on Control and Optimization, 53 (2015), pp. 1982–2016.

[37] H. Yu and D. P. Bertsekas, *A mixed value and policy iteration method for stochastic control with universally measurable policies*, Mathematics of Operations Research, 40 (2015), pp. 926–968.