Imperfect-Recall Abstractions with Bounds

Christian Kroer and Tuomas Sandholm

Computer Science Department
Carnegie Mellon University
ckroer@cs.cmu.edu, sandholm@cs.cmu.edu

Abstract. We develop the first general, algorithm-agnostic, solution quality guarantees for Nash equilibria and approximate self-trembling equilibria computed in imperfect-recall abstractions, when implemented in the original (perfect-recall) game. Our results are for a class of games that generalizes the only previously known class of imperfect-recall abstractions where any results had been obtained. Further, our analysis is tighter in two ways, each of which can lead to an exponential reduction in the solution quality error bound.
We then show that for extensive-form games that satisfy certain properties, the problem of computing a bound-minimizing abstraction for a single level of the game reduces to a clustering problem, where the increase in our bound is the distance function. This reduction leads to the first imperfect-recall abstraction algorithm with solution quality bounds. We proceed to show a divide in the class of abstraction problems. If payoffs are at the same scale at all information sets considered for abstraction, the input forms a metric space, and this immediately yields a 2-approximation algorithm for abstraction. Conversely, if this condition is not satisfied, we show that the input does not form a metric space. Finally, we provide computational experiments to evaluate the practical usefulness of the abstraction techniques. They show that running counterfactual regret minimization on such abstractions leads to good strategies in the original games.

1 Introduction

Game-theoretic equilibrium concepts provide a sound definition of how rational agents should act in multiagent settings. To operationalize them, they have to be accompanied by techniques to compute equilibria, an important topic that has received significant attention in the literature [23, 22, 10, 30, 16, 20].

Typically, equilibrium-finding algorithms do not scale to very large games. This holds even for two-player zero-sum games (that can be solved in polynomial time [18]) when the games get large. Therefore, the following has emerged as the leading framework for solving large extensive-form games [24]. First, the game is abstracted to generate a smaller game. Then the abstract game is solved for (near-)equilibrium. Then, the strategy from the abstract game is mapped back to the original game. Initially, game abstractions were created by hand, using domain dependent knowledge [26, 2]. More recently, automated abstraction has taken over [8, 10, 30]. This has typically been used for information abstraction, whereas action abstraction is still largely done by hand [12]. Recently, automated action abstraction approaches have also started to emerge [14, 15, 25, 3].

Ideally, abstraction would be performed in a lossless way, such that implementing an equilibrium from the abstract game results in an equilibrium in the full game. Lossless abstraction techniques were introduced by Gilpin and Sandholm [10] for a class of games called *game of ordered signals*. Unfortunately, lossless abstraction often leads to games that are still too large to solve. Thus, we must turn to lossy abstraction. However, significant abstraction *pathologies* (*nonmonotonicities*) have been shown in games that cannot exist in single-agent settings: if an abstraction is refined, the equilibrium strategy from that new abstraction can actually be worse in the original game than the equilibrium strategy from a coarser abstraction [28]! Until recently, all lossy abstraction techniques for general games of imperfect information were without any solution quality bounds. Basilico and Gatti [1] give bounds for the special game class called *patrolling security game*. Johanson et al. [17] provide computational methods for evaluating the quality of a given abstraction via computing a best response in the full game after the fact. Sandholm and Singh [25] provide lossy abstraction algorithms with bounds for stochastic games. Lanctot et al. [21] present regret bounds in a class of imperfect-recall abstractions for equilibria computed using the *counterfactual regret minimization algorithm (CFR)* [30], with their result also extending to perfect-recall abstractions. Finally, Kroer and Sandholm [19] show solution quality bounds for a broad class of perfect-recall abstractions. They leave as an open problem whether similar bounds can be achieved for imperfect-recall abstractions, which are the state of the art in practical poker solving [27, 17, 5]. In a somewhat related vein, Waugh et al. [29] introduce an approach that uses functional regret estimation within CFR; their technique converges to a Nash equilibrium if the regrets are realizable by the function approximator. In a somewhat different line of research, Brown and Sandholm [4] introduce a technique for simultaneously computing an equilibrium and refining the abstraction. This eventually converges, but does not give guarantees without after-the-fact best-response calculation.

Generalizing the notion of *skew well-formed games* introduced by Lanctot et al. [21], we adapt the techniques of Kroer and Sandholm [19] to give similar results for imperfect-recall abstractions. The solution quality bounds we derive are exponentially stronger than those of Lanctot et al. [21] which had a linear dependence on the number of information sets, and did not weight the leaf reward error by the probability of a given leaf being reached. The reward error term in our result has only a linear dependence on tree height (actually, just the number of information sets any single player can experience on a path of play). Our leaf reward error term is weighted by the probability of the leaf being reached. Each of these two reasons can lead to an exponentially tighter bound. Furthermore, our bounds are independent of the equilibrium computation method, while that prior work was only for CFR.

Our results actually extend to a new game class which we coin *chance-relaxed skew well-formed (CRSWF) games*, a relaxation of skew well-formed games. It allows for a richer set of abstractions where nodes can go in the same abstract information set even if the nature probabilities of reaching those nodes and going from those nodes are not the same. This enables dramatically smaller abstractions for such games.

For games where abstraction of a subset of information sets at a single level is guaranteed to result in a CRSWF game, we show an equivalence between the problem of computing an abstraction that minimizes our theoretical solution quality guarantees and a class of clustering problems. Using the decrease in solution quality bound from abstracting a pair of information sets as a distance function, we show that such abstraction problems form a metric space. This yields a 2-approximation algorithm for performing abstraction at a signle level in the game tree when information sets differ only by the actions taken by players. When information sets differ based on nature's actions, our equivalence yields a new clustering objective that has not, to our knowledge, been previously studied. Our clustering results yield the first abstraction algorithm for computing imperfect-recall abstractions with solution quality bounds. Finally, we use these results to conduct experiments on a simple die-based poker game that has been used as a benchmark for game abstraction in prior work. The experiments show that the CFR algorithm works well even on abstraction where different nature probabilities are abstracted together, and that the theoretical bound is within 0 to 2 orders of magnitude of the regrets at CFR convergence.

2 Extensive-form games

An extensive-form game $\mathit{\Gamma}$ is a tuple $\langle N, A, S, Z, \mathcal{H}, \sigma_0, u, \mathcal{I}\rangle$. N is the set of players in the game. A is the set of all actions in the game. S is a set of nodes corresponding to sequences of actions. They describe a tree with root node $r \in S$. At each node s, some Player i is active with actions A_s, and each branch at s denotes a different choice in A_s. The set of all nodes where Player i is active is called S_i. $Z \subset S$ is the set of leaf nodes, where $u_i(z)$ is the utility to Player i of node z. We assume, without loss of generality, that all utilities are non-negative. Z_s is the subset of leaf nodes reachable from a node s. $\mathcal{H}_i \subseteq \mathcal{H}$ is the set of heights in the game tree where Player i acts. $\mathcal{H}_0$ is the set of heights where nature acts. σ_0 specifies the probability distribution for nature, with $\sigma_0(s, a)$ denoting the probability of nature choosing outcome a at node s.

$\mathcal{I}_i \subseteq \mathcal{I}$ is the set of information sets where Player i acts. $\mathcal{I}_i$ partitions S_i. For any two nodes s_1, s_2 in the same information set I, Player i cannot distinguish among them, and $A_{s_1} = A_{s_2}$. We let $X(s)$ denote the set of information set and action pairs I, a in the sequence leading to a node s, including nature. We let $X_{-i}(s), X_i(s) \subseteq X(s)$ be the subset of this sequence such that actions by the subscripted player(s) are excluded or exclusively chosen. Let $X^b(s)$ be the set of possible sequences of actions players can take in the subtree at s, with $X^b_{-i}(s), X^b_i(s)$ being the set of future sequences excluding or limited to Player i, respectively. We denote elements in these sets as $\boldsymbol{a}$. $X^b(s, \boldsymbol{a}), X^b_{-i}(s, \boldsymbol{a}), X^b_i(s, \boldsymbol{a})$ are the analogous sets limited to sequences that are consistent with the sequence of actions $\boldsymbol{a}$. We let $s[I]$ denote the predecessor $\hat{s}$ of the node s such that $\hat{s} \in I$.

Perfect recall means that no player forgets anything that that player observed in the past. Formally, for every Player $i \in N$, information set $I \in \mathcal{I}_i$, and nodes $s_1, s_2 \in I : X_i(s_1) = X_i(s_2)$. Otherwise, the game has *imperfect recall*. For games $\mathit{\Gamma}' = \langle N, A, S, Z, \mathcal{H}, \sigma_0, u, \mathcal{I}'\rangle$ and $\mathit{\Gamma} = \langle N, A, S, Z, \mathcal{H}, \sigma_0, u, \mathcal{I}\rangle$, we say that $\mathit{\Gamma}$ is a perfect-recall refinement of $\mathit{\Gamma}'$ if $\mathit{\Gamma}$ has perfect-recall, and for any information set $I \in \mathcal{I} : \exists I' \in \mathcal{I}', I \subseteq I'$. That is, the game $\mathit{\Gamma}$ can be obtained by partitioning the nodes of each information set in $\mathcal{I}'$ appropriately. For any perfect-recall refinement $\mathit{\Gamma}$, we let $\mathcal{P}(I')$ denote the information sets $I \in \mathcal{I}$ such that $I \subseteq I'$ and $\bigcup_{I \in \mathcal{P}(I')} = I'$. For an information set

I in a perfect-recall refinement Γ of Γ', we let f_I denote the corresponding information set in Γ'.

We will focus on the setting where we start out with some perfect-recall game Γ, and wish to compute an imperfect-recall abstraction such that the original game is a perfect-recall refinement of the abstraction. Imperfect-recall abstractions will be denoted by $\Gamma' = \langle N, A, S, Z, \mathcal{H}, \sigma_0, u, \mathcal{I}' \rangle$. That is, they are the same game, except that some information sets have been merged.

We denote by σ_i a *behavioral strategy* for Player i. For each information set I where it is the player's turn to move, it assigns a probability distribution over A_I, the actions at the information set. $\sigma_i(I, a)$ is the probability of playing action a. A *strategy profile* $\sigma = (\sigma_0, \ldots, \sigma_n)$ consists of a behavioral strategy for each player. We will often use $\sigma(I, a)$ to mean $\sigma_i(I, a)$, since the information set uniquely specifies which Player i is active. As described above, randomness external to the players is captured by the nature outcomes σ_0. Using this notation allows us to treat nature as a player when convenient. We let $\sigma_{I \to a}$ denote the strategy profile obtained from σ by having Player i deviate to taking action a at $I \in \mathcal{I}_i$.

Let the probability of going from node s to node $\hat{s}$ under strategy profile σ be $\pi^\sigma(s, \hat{s}) = \Pi_{\langle \bar{s}, \bar{a} \rangle \in X_{s,\hat{s}}} \sigma(\bar{s}, \bar{a})$ where $X(s, \hat{s})$ is the set of pairs of nodes and actions on the path from s to $\hat{s}$. We let the probability of reaching node s be $\pi^\sigma(s) = \pi^\sigma(r, s)$, the probability of going from the root node r to s. Let $\pi^\sigma(I) = \sum_{s \in I} \pi^\sigma(s)$ be the probability of reaching any node in I. For probabilities over nature, $\pi_0^\sigma(s) = \pi_0^{\bar{\sigma}}(s)$ for all $\sigma, \bar{\sigma}, s \in S_0$, so we can ignore the superscript and write π_0. Finally, for all behavioral strategies, the subscript $-i$ refers to the same definition, but without including Player i.

For information set I and action $a \in A_I$ at level $k \in \mathcal{H}_i$, we let $\mathcal{D}_I^a$ be the set of information sets at the next level in $\mathcal{H}_i$ reachable from I when taking action a. Similarly, we let $\mathcal{D}_I^l$ be the set of descendant information sets at height $l \leq k$, where $\mathcal{D}_I^k = \{I\}$. Let t_a^s be the node transitioned to by performing action $a \in A_s$ at node s. Finally, we let $\mathcal{D}_s^{\boldsymbol{a},j}$ be the set of information sets reachable from node s when action-vector $\boldsymbol{s}$ is played with probability one.

2.1 Chance-relaxed skew well-formed (CRSWF) games

In this paper we will only be concerned with imperfect-recall abstractions where the original game is a perfect-recall refinement satisfying a certain set of properties. We call imperfect-recall games with such refinements *CRSWF games*. They are a relaxation of the class *skew well-formed games* introduced by [21].

Definition 1. *For an extensive-form game Γ', and a perfect-recall refinement Γ, we say that Γ' is an* CRSWF game with respect to Γ *if for all $i \in N, I' \in \mathcal{I}_i', I, \check{I} \in \mathcal{P}(I')$, there exists a bijection $\phi : Z_I \to Z_{\check{I}}$ such that for all $z \in Z_I$:*

1. *In Γ', $X_{-\{i,0\}}(z) = X_{-\{i,0\}}(\phi(z))$, that is, for two leaf nodes mapped to each other (for these two information sets in the original game), the action sequences of the other players on those two paths must be the same in the abstraction.*[1]

[1] It is possible to relax this notion slightly: if two actions of another player are not the same, as long as they are on the path (at the same level) to all nodes in their respective full-game

2. *In Γ', $X_i(z[I], z) = X_i(\phi(z)[\breve{I}], \phi(z))$, that is, for two leaf nodes mapped to each other (for information sets I and $\breve{I}$ in the original game), the action sequence of Player i from I to z and from $\breve{I}$ to $\phi(z)$ must be the same in the abstraction.*

This definition implicitly assumes that leaf nodes are all at the same level. This is without loss of generality, as any perfect-recall game can be extended to satisfy this.

With this definition, we can define the following error terms for a CRSWF refinement Γ of an imperfect-recall game Γ' for all $i \in N, I' \in \mathcal{I}_i', I, \breve{I} \in \mathcal{P}(I')$

- $\left|u_i(z) - \delta_{I,\breve{I}} u_i(\phi(z))\right| \leq \epsilon^R_{I,\breve{I}}(z)$, the reward error at z, after scaling by $\delta_{I,\breve{I}}$ at $\breve{I}$.
- $\left|\pi_0(z[I], z) - \pi_0(\phi(z)[\breve{I}], \phi(z))\right| \leq \epsilon^0_{I,\breve{I}}(z)$, the leaf probability error at z.
- $\left|\frac{\pi_0(z[I])}{\pi_0(I)} - \frac{\pi_0(\phi(z)[\breve{I}])}{\pi_0(\breve{I})}\right| \leq \epsilon^D_{I,\breve{I}}(z[I])$, the distribution error of $z[I]$.

Lanctot et al. [21] require $\pi_0(z) = l_{I,\breve{I}} \pi_0(\phi_{I,\breve{I}}(z))$, where $l_{I,\breve{I}}$ is a scalar defined on a per information set pair basis. We omit any such constraint, and instead introduce distribution error terms as above. Our definition allows for a richer class of abstractions. Consider some game where every nature probability in the game differs by a small amount. For such a game, no two information sets can be merged according to the skew well-formed games definition, whereas our definition allows such abstraction.

We define $\overline{u}_{I,\breve{I}}(s) = \max_{z \in Z_s} u_i(z) + \epsilon^R_{I,\breve{I}}(z)$, the maximum utility plus its scaled error achieved at any leaf node for Player i. This will simplify notation when we take the maximum over error terms related to probability transitions.

Conditions 1-3 above define approximation error terms. We now define additional aggregate approximation error terms. These will be useful when reasoning inductively about more than one height of the game at a time. We do not subscript by the player index i, since all analysis in the remainder of the paper is conducted on a per-player (assumed to be i) basis. We define the *reward approximation error* $\epsilon^R_{I,\breve{I}}(s)$ for information sets $I, \breve{I} \in \mathcal{P}(I')$ and any node s in $\breve{I}$ to be

$$\epsilon^R_{I,\breve{I}}(s) = \begin{cases} \epsilon^R_{I,\breve{I}}(z) & \text{if } \exists z \in Z : z = s \\ \sum_{a \in A_s} \sigma_0(s,a) \epsilon^R_{I,\breve{I}}(t^s_a) & \text{if } s \in S_0 \\ \max_{a \in A_s} \epsilon^R_{I,\breve{I}}(t^s_a) & \text{if } s \notin S_0 \wedge s \notin Z \end{cases},$$

We define the *transition approximation error* $\epsilon^0_{I,\breve{I}}(s)$ for information sets $I, \breve{I} \in \mathcal{P}(I')$ and any node s in $\breve{I}$ to be

$$\epsilon^0_{I,\breve{I}}(s) = \begin{cases} \epsilon^0_{I,\breve{I}}(z) \overline{u}_{I,\breve{I}}(z) & \text{if } \exists z \in Z : z = s \\ \sum_{a \in A_s} \epsilon^0_{I,\breve{I}}(t^s_a) & \text{if } s \in S_0 \\ \max_{a \in A_s} \epsilon^R_{I,\breve{I}}(t^s_a) & \text{if } s \notin S_0 \wedge s \notin Z \end{cases}$$

information sets (I and $\breve{I}$), they do not affect the distribution over nodes in the information sets, and are thus allowed to differ in the abstraction.

We define the *distribution approximation error* for an information set pair $I, \breve{I} \in \mathcal{P}(I')$ to be

$$\epsilon^D_{I,\breve{I}} = \sum_{s \in I} \epsilon^D_{I,\breve{I}}(s) \overline{u}_{I,\breve{I}}(s)$$

2.2 Value functions

We define value functions both for individual nodes and for information sets. The value for Player i of a given node s under strategy profile σ is $V_i^\sigma(s) = \sum_{z \in Z_s} \pi^\sigma(s,z) u_i(z)$. We use the definition of counterfactual value of an information set, introduced by Zinkevich et al. [30], to reason about the value of an information set under a given strategy profile. The counterfactual value of an information set I is the expected utility of the information set, assuming that all players follow strategy profile σ, except that Player i plays to reach I, normalized by the probability of reaching the information set. This latter normalization, introduced by Kroer and Sandholm [19], is not part of the original definition, but it is useful for inductively proving bounds over information sets at different heights of the game tree.

Definition 2. *For a perfect-recall game Γ, the counterfactual value for Player i of a given information set I under strategy profile σ is*

$$V_i^\sigma(I) = \begin{cases} \sum_{s \in I} \frac{\pi^\sigma_{-i}(s)}{\pi^\sigma_{-i}(I)} \sum_{z \in Z_s} \pi(s,z) u_i(z) & \textit{if } \pi^\sigma_{-i}(I) > 0 \\ 0 & \textit{if } \pi^\sigma_{-i}(I) = 0 \end{cases}$$

We sometimes write $V(I) = V_i^\sigma(I)$ when the strategy and player are both clear from context. For the information set I_r that contains just the root node r, we have that $V_i^\sigma(I_r) = V_i^\sigma(r)$, which is the value of playing the game with strategy profile σ. We assume that at the root node it is not nature's turn to move. This is without loss of generality since we can insert dummy player nodes above it. For imperfect-recall information sets, we let $W(I') = \sum_{s \in I'} \frac{\pi^\sigma(s)}{\pi^\sigma(I')} V(s)$ be the value of an information set.

In perfect-recall games, for information set I at height $k \in \mathcal{H}_i$, $V_i^\sigma(I)$ can be written as a sum over descendant information sets at height $\hat{k} \in \mathcal{H}_i$, where $\hat{k}$ is the next level below k that belongs to Player i (a proof is given by Kroer and Sandholm [19]):

$$V_i^\sigma(I) = \sum_{a \in A_I} \sigma(I,a) \sum_{\hat{I} \in \mathcal{D}_I^a} \frac{\pi^\sigma_{-i}(\hat{I})}{\pi^\sigma_{-i}(I)} V_i^\sigma(\hat{I}) \tag{1}$$

We will later be concerned with a notion of how much better a player i could have done at an information set: the regret for information set I and action a is $r(I,a) = V_i^{\sigma_{I \rightarrow a}}(I) - V_i^\sigma(I)$. That is, the increase in expected utility for Player i obtained by deviating to taking action a at I. The immediate regret at an information set I given a strategy profile σ is $r(I) = \max_{a \in A_I} r(I,a)$. Regret is define analogously for imperfect-recall games using $W(I)$.

2.3 Equilibrium concepts

In this section we define the equilibrium concepts we use. We start with two classic ones.

Definition 3 (ϵ-Nash and Nash equilibria). *An ϵ-Nash equilibriumis a strategy profile σ such that for all i, $\bar{\sigma}_i$: $V_i^{\sigma}(r) + \epsilon \geq V_i^{\sigma_{-i},\bar{\sigma}_i}(r)$. A Nash equilibrium is an ϵ-Nash equilibrium where $\epsilon = 0$.*

We will also use the concept of a *self-trembling equilibrium*, introduced by Kroer and Sandholm [19]. It is a Nash equilibrium where the player assumes that opponents make no mistakes, but she might herself make mistakes, and thus her strategy must be optimal for all information sets that she could mistakenly reach by her own fault.

Definition 4 (Self-trembling equilibrium). *For a game Γ, a strategy profile σ is a self-trembling equilibrium if it satisfies two conditions. First, it must be a Nash equilibrium. Second, for any information set $I \in \mathcal{I}_i$ such that $\pi_{-i}^{\sigma}(I) > 0$, and for all alternative strategies $\bar{\sigma}_i$, $V_i^{\sigma}(I) \geq V_i^{\sigma_{-i},\bar{\sigma}_i}(I)$. We call this second condition the* self-trembling *property.*

An ϵ-self-trembling equilibrium is defined analogously, for each information set $I \in \mathcal{I}_i$, we require $V_i^{\sigma}(I) \geq V_i^{\sigma_{-i},\bar{\sigma}_i}(I) - \epsilon$. For imperfect-recall games, the property $\pi_{-i}^{\sigma}(I') > 0$ does not give a probability distribution over the nodes in an information set I', since Player i can affect the distribution over the nodes. For such information sets, it will be sufficient for our purposes to assume that σ_i is (approximately) utility maximizing for some (arbitrary) distribution over $\mathcal{P}(I')$: our bounds are the same for any such distribution.

3 Strategies from abstract near-equilibria have bounded regret

To prove our main result, we first show that strategies with bounded regret at information sets in CRSWF games have bounded regret at their perfect-recall refinements (All proofs can be found in the appendix).

Proposition 1. *For any CRSWF game Γ', refinement Γ, strategy profile σ, and information set $I' \in \mathcal{I}'$ such that Player i has bounded regret $r(I', a)$ for all $a \in A_{I'}$, the regret $r(I, a^*)$ at any information set $I \in \mathcal{P}(I')$ and action $a^* \in A_I$ is bounded by*

$$r(I, a^*) \leq \max_{\check{I} \in \mathcal{P}(I')} \delta_{I,\check{I}} r(I', a^*) + 2 \sum_{s \in I} \frac{\pi^{\sigma}(s)}{\pi^{\sigma}(I)} \left(\epsilon_{I,\check{I}}^{0}(s) + \epsilon_{I,\check{I}}^{R}(s) \right) + \epsilon_{I,\check{I}}^{D}$$

Intuitively, the scaling variable $\delta_{I,\check{I}}$ ensures that if the regret at I' is largely based on some other information set, then the regret is scaled to fit with the payoffs at I.

With this result, we are ready to prove our main results. First, we show that strategies with bounded regret at each information set in CRSWF games are ϵ-self-trembling equilibria when implemented in any perfect-recall refinement.

Theorem 1. *For any CRSWF game Γ' and strategy σ with bounded immediate regret $r_{I'}$ at each information set $I' \in \Gamma'$ where $\sigma_{-i}(I') > 0$, σ is an ϵ-self-trembling equilibrium when implemented in any perfect-recall refinement Γ, where $\epsilon = \max_{i \in N} \epsilon_i$ and*

$$\epsilon_i = \max_{\boldsymbol{a} \in X_i^b(r)} \sum_{j \in \mathcal{H}_i, j \leq l} \sum_{I \in \mathcal{D}_r^{\boldsymbol{a},j}} \pi_{-i}^{\sigma}(I) \left(\max_{\breve{I} \in \mathcal{P}(f_I)} \delta_{I,\breve{I}} r(f_I) \right.$$
$$\left. +2 \sum_{s \in I} \frac{\pi^{\sigma}(s)}{\pi^{\sigma}(I)} \left(\epsilon_{I,\breve{I}}^{0}(s) + \epsilon_{I,\breve{I}}^{R}(s) \right) + \epsilon_{I,\breve{I}}^{D} \right)$$

This version of our bound weights the error at each information set by the probability of reaching the information set, and similarly, the error at each of the nodes in the information set is weighted by the probability of reaching it. This is important for CFR-style algorithms, where the regret at each information set I only goes to zero when weighted by $\pi_{-i}^{\sigma}(I)$, the probability of it being reached if Player i played to reach it. In some instances it might be desirable to work with a different version of our bound. If one wishes to compute an abstraction that minimizes the bound independently of a specific strategy profile, it is possible to take the maximum over all player actions. Importantly, this preserves the probability distribution over errors at nature nodes. In the previous CFR-specific results of Lanctot et al. [21], the reward error bound for each information set was the maximum reward error at any leaf node. Having the reward error be a weighted sum over the nature nodes and only maximized over player action sequences allows significantly finer-grained measurement of similarity between information sets. Consider any poker game where an information set represents the hand that the player holds, and three hands: a pair of aces I_A, pair of kings I_K, or pair of twos I_2. When the reward error is measured as the maximum over nodes in the information set, I_A and I_K are as dissimilar as I_A, I_2, since the winner changes for at least one hand held by the opponent for both information sets. In contrast to this, when reward errors are weighted by the probability of them being reached, we get that I_A, I_K are much more similar than I_A, I_2.

Our proof techniques have their root in those of Kroer and Sandholm [19]. We devise additional machinery, mainly Proposition 1 and the notion of CRSWF abstractions, to deal with imperfect recall. In doing so, our bounds get a linear dependence on height for the reward approximation error. The prior bounds [19] have no dependence on height for the reward approximation error, and are thus tighter for perfect-recall abstractions.

We now show a second version of our result, which concerns the mapping of Nash equilibria in CRSWF games to approximate Nash equilibria in perfect-recall refinements.

Theorem 2. *For any CRSWF game Γ' and Nash equilibrium σ, σ is an ϵ-Nash equilibrium when implemented in any perfect-recall refinement Γ, where $\epsilon = \max_{i \in N} \epsilon_i$ and*

$$\epsilon_i = \max_{\boldsymbol{a} \in X_i^b(r)} \sum_{j \in \mathcal{H}_i, j \leq l} \sum_{I \in \mathcal{D}_r^{\boldsymbol{a},j}} \pi_{-i}^{\sigma}(I) \left(\max_{\breve{I} \in \mathcal{P}(f_I)} 2 \sum_{s \in I} \frac{\pi^{\sigma}(s)}{\pi^{\sigma}(I)} \left(\epsilon_{I,\breve{I}}^{0}(s) + \epsilon_{I,\breve{I}}^{R}(s) \right) + \epsilon_{I,\breve{I}}^{D} \right)$$

For practical game solving, Theorem 1 has an advantage over Theorem 2: any algorithm that provides guarantees on immediate counterfactual regret in imperfect-recall games can be applied. For example, the CFR algorithm can be run on a CRSWF abstraction, and achieve the bound in Theorem 1, with the information set regrets $\pi^{\sigma}_{-i}(I)r(f_I)$ decreasing at a rate of $O(\sqrt{(T)})$. Conversely, no good algorithms are known for computing Nash equilibria in imperfect-recall games.

4 Complexity and algorithms

We now investigate the problem of computing CRSWF abstractions that minimize the error bounds in Theorem 2. First, we show that this is hard, even for games with a single player and a game tree of height two[2].

Theorem 3. *Given a perfect-recall game and a limit on the number of information sets, determining whether a CRSWF abstraction with a given bound as in Theorem 1 or 2 exists is NP-complete. This holds even if there is only a single player, and the game tree has height two.*

Performing abstraction at a single level of the game tree that minimizes our bound reduces to clustering if the information sets considered for clustering satisfy Conditions 1 and 2. The distance function for clustering depends on how the trees match on utility and nature error, and the objective function depends on the topology higher up the tree. In such a setting, an imperfect-recall abstraction with solution quality bounds can be computed by clustering valid information sets level-by-level in a bottom-up fashion. In general, a level-by-level approach has no optimality guarantees, as some games allow *no* abstraction unless coupled with other abstraction at different levels (a perfect-recall abstraction example of this is shown by Kroer and Sandholm [19]). However, considering all levels simultaneously is often impossible in practice. A medical example of a setting where such a level-by-level scheme could be applied is given by [6], where an opponent initially chooses a robustness measure, which impacts nature outcomes and utility, but not the topology of the different subtrees. Similarly, the *die-roll poker* game introduced by Lanctot et al. [21] as a game abstraction benchmark is amenable to this approach.

We now show that single-level abstraction problems where Conditions 1 and 2 of Definition 1 are satisfied for all merges form a metric space together with the distance function that measures the error bound for merging information set pairs. Clustering problems are often computationally easier when the input forms a metric space, yielding approximation algorithms with constant approximation factors [13, 7].

Definition 5. *A* metric space *is a set M and a distance function $d : M \times M \to \mathbb{R}$ such that the following holds for all $x, y, z \in M$:*

1. $d(x,y) \geq 0$
2. $d(x,y) = 0 \Leftrightarrow x = y$ *(identity of indiscernibles)*

[2] Sandholm and Singh [25] already showed hardness of computing an optimal abstraction in the sense of minimizing the actual loss of a unique equilibrium.

3. $d(x, y) = d(y, x)$ *(symmetry)*
4. $d(x, y) \leq d(x, z) + d(z, y)$ *(triangle inequality)*

Proposition 2. *For a set of information sets $\mathcal{I}^m$ such that any partitioning of $\mathcal{I}^m$ yields a CRSWF abstraction (without scaling variables), and a function $d : \mathcal{I}^m \times \mathcal{I}^m \to \mathbb{R}$ describing the loss incurred in the error bound when merging $I, \breve{I} \in \mathcal{I}^m$, the pair $(\mathcal{I}^m, d)$ forms a metric space.*

Conversely to our result above, if the scaling variables can take on any value, the triangle inequality does not hold, so $(\mathcal{I}^m, d)$ is not a metric space. Consider three information sets I_1, I_2, I_3, each with two nodes reached with probability 0.9 and 0.1, respectively. Let there be one action at each information set, leading directly to a leaf node in all cases. Let $I_1 = \{1, 2\}, I_2 = \{5, 11\}, I_3 = \{10, 23\}$, where the name of the node is also the payoff of Player 1 at the node's leaf. We have that I_1 and I_2 map onto each other with scaling variable $\delta_{I_1,I_2} = 5$ to get $\epsilon^R_{I_1,I_2} = 1$ and $I_{2,3}$ with $\delta_{I_2,I_3} = 2, \epsilon^R_{I_2,I_3} = 1$. However, I_1 and I_3 map onto each other with $\delta_{I_1,I_3} = 10$ to get $\epsilon^R_{I_1,I_3} = 3$ which is worse than the sum of the costs of the other two mappings, since all reward errors on the right branches are multiplied by the same probability 0.1, i.e.,

$$0.1 \cdot \epsilon^R_{I_1,I_2} + 0.1 \cdot \epsilon^R_{I_2,I_3} < 0.1 \cdot \epsilon^R_{I_1,I_3}$$

.

The objective function for our abstraction problem has two extreme versions. The first is when the information set that is reached depends entirely on players not including nature. In this case, the bound on error over the abstraction at the level is the maximum error of any single information set. This is equivalent to the minimum diameter clustering problem, where the goal is to minimize the maximum distance between any pair of nodes that share a cluster; Gonzalez [13] gave a 2-approximation algorithm when the distance function satisfies the triangle inequality. Above we gave conditions when the abstraction problem is a metric space (which implies that the triangle inequality is satisfied). This gives a 2-approximation algorithm for minimizing our bound over single-level abstraction problems.

The other extreme is when each of the information sets being reached differ only in nature's actions. In this setting, the error bound over the abstraction is a weighted sum of the error at each information set. This is equivalent to clustering where the objective function being minimized is the weighted sum over all elements, with the cost of each element being the maximum distance to any other element within its cluster. To our knowledge, clustering with this objective function has not been studied in the literature, even when the weights are uniform.

Generally, the objective function can be thought of as a tree, where a given leaf node represents some information set, and takes on a value equal to the maximum distance to any information set with which it is clustered. Each internal node either takes the maximum or weighted sum of its child-node errors. The goal is to minimize the error at the root node.

In practice, integer programs (IPs) have sometimes been applied to clustering information sets for extensive-form game abstraction [9, 11] (without bounds on solution quality, and just for perfect-recall abstractions), and are likely to perform well in our setting. An IP can easily be devised for any objective function in the above tree form. A real-valued variable is introduced for each internal node in the objective function tree described above. If the node takes the maximum, constraints are introduced forcing its variable to be larger than the value of each child node. If the node takes the weighted sum, a constraint is introduced forcing it to be larger than the weighted sum.

5 Experiments

We now investigate what the optimal single-level abstraction bounds look like for the *die roll poker (DRP)* game, a benchmark game for testing abstraction [21]. Die-roll poker is a simple two-player zero-sum poker game where dice, rather than cards, are used to determine winners. At the beginning of the game, each player antes one chip to the pot. Each player then rolls a private six-sided die. It is an incomplete-information game since the players don't observe each other's die rolls. After each player has rolled their first die, a betting round occurs. During betting rounds, a player may fold (causing the other player to win the game), call (match the current bet), or raise (increase the current bet by a fixed amount), with a maximum of two raises per round. In the first round, each raise is worth two chips. In the second round, each raise is worth four chips. The maximum that a player can bet is 13 chips, if each player uses all their raises. At the end of the second round, if neither player has folded, a showdown occurs. In the showdown, the player with the largest sum of the two dice wins all the chips in the pot. If the players are tied, the pot is split.

DRP has the nice property that abstractions computed at the bottom level of the tree satisfy Conditions 1 and 2 of Definition 1. At heights above that one we can similarly use our clustering approach, but where two information sets are eligible for merging only if there is a bijection between their future die rolls such that the information sets for the future rolls in the bijection have been merged. Thus, a clustering would be computed for each set in the partition that represents a group of information sets eligible for merging. In the experiments in this paper we will focus on abstraction at the bottom level of the tree. We use CPLEX to solve an IP encoding the single-level abstraction problem, which computes the optimal abstraction for the level, given a limit on the number of abstract information sets. The results are shown in Figure 1. For one or

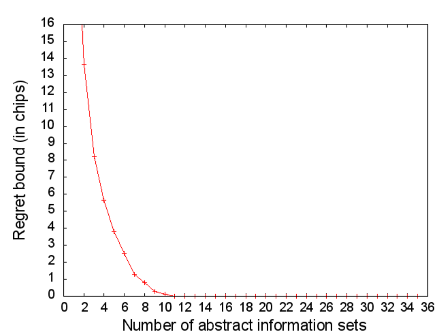

Fig. 1. Regret bounds for varying numbers of abstract information sets at the last level in DRP.

two clusters, the bound is bigger than the largest payoff in the game, but already at three clusters it is significantly lower. At eight clusters, the bound is smaller than that of always folding, and decreases steadily to zero at eleven clusters, where a lossless abstraction is found (the original game has 36 information sets). While these experiments show that our bound is relatively small for the DRP game, they are limited in that we only performed abstraction at a single level. If abstraction at multiple levels is performed, the bound is additive in the error over the levels. Nonetheless, our bounds are significantly tighter than the only previously known bounds, and in future work it would be interesting to see whether the height dependence can be removed.

Another important question is how well strategies computed in abstractions that are good—as measured by our bound—perform in practice. This has already been partially answered. Lanctot et al. [21] conducted experiments to investigate the performance of CFR strategies computed in imperfect-recall abstractions of several games: DRP, Phantom tic-tac-toe (where moves are observed a turn later than in regular tic-tac-toe), and Bluff (also known as Liar's Dice, Dudo, and Perudo). They found that CFR computes strong strategies in imperfect-recall abstractions of all these games, even when the abstraction did not necessarily fall under their framework. Their experiments validate a subset of the class of CRSWF abstractions: ones where there is no chance (nature) error. Our framework provides exponentially stronger bounds than those of Lanctot et al. [21] for these settings.

Due to the existing experimental work of Lanctot et al. [21], we focus our experiments on problems where abstraction does introduce nature error. One broad class of problems where such error can naturally occur are settings where players observe imperfect signals of some phenomenon. For such settings, one would expect that there is correlation between the observations made by the players. Concrete examples include negotiation, sequential auctions, and strategic acquisition.

DRP can be thought of as a game where the die rolls are the signals. Regular DRP has a uniform distribution over the signals. We now consider a generalization of DRP where die rolls are correlated: *correlated die-roll poker* (CDRP). There are many variations on how one could make the rolls correlated; we use the following. We have a single correlation parameter c, and the probability of any pair of values (v_1, v_2), for Player 1 and 2 respectively, is $\frac{1}{\#sides^2} - c\,|v_1 - v_2|$. The probabilities for the second round of rolls is independent of the first round. As an example, the probability of Player 1 rolling a 3 and Player2 rolling a 5 with a regular 6-sided die in either round would be $\frac{1}{36} - 2c$. We generate DRP games with a 4-sided die and $c \in \{0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07\}$.

For each value of c, we compute the optimal bound-minimizing abstraction for the second round of rolls, with a static mapping between information sets such that for any sequence of opponent rolls, the nodes representing that sequence in either information set are mapped to each other. The bound cost of the mappings is precomputed, and the optimal abstraction is found with a standard MIP formulation of clustering. This scheme ensures that we compute an imperfect-recall abstraction that falls under the CRSWF game class. After computing the optimal abstraction for a given game, we run CFR on the abstraction, and measure the regret for either player in terms of their regret in the full game. Figure 2 shows the results of these experiments. On the x-axis is the number of CFR iterations. On the y-axis is $r_1 + r_2$, where r_i is the regret for Player i for the strategy at a given iteration. Furthermore, the horizontal lines denote the regret bound of Theorem 2 for an exact Nash equilibrium. On the left in Figure 2 is shown

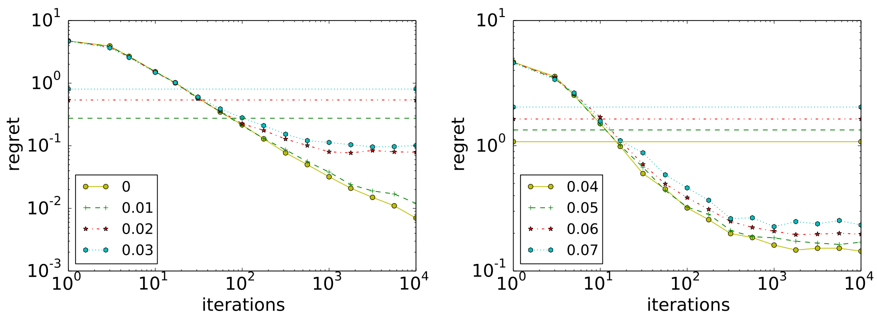

Fig. 2. Sum of the two players' regrets as a function of CFR iterations on the bound-minimizing abstraction of CDRP. The legends give the amount of correlation in the die rolls of the different CDRP games on which we ran experiments. The horizontal lines show the respective ex-ante regret bound of Theorem 2 for each of the CDRP games. (In the first game on the left where the correlation is zero, the abstraction is lossless, so the horizontal line (not shown) would be at zero.)

the results for the four smallest values of c, on the right the four largest values. As can be seen, CFR performs well on the abstractions, even for large values of c: when $c = 0.7$, a very aggressive abstraction, the sum of regrets still goes down to ~ 0.25 (for reference, always folding has a regret of 1). We also see that for $c \geq 0.2$, the regret stops decreasing after around 1000 iterations. This is likely where CFR converges in the abstraction, with the remaining regret representing the information lost through the abstraction. We also see that our theoretical bound is at the same order of magnitude as the actual bound even when CFR converges.

6 Conclusions and future research

This paper presented the first general solution quality guarantees for strategies computed in imperfect-recall abstractions of games. We defined CRSWF games, extending skew well-formed games to incorporate nature error, which was left as an open problem by Lanctot et al. [21]. We proved exponentially stronger solution quality guarantees than prior related bounds, achieving the first solution quality bounds that take nature probabilities into account when measuring reward error. The exponential improvement in bound quality was achieved in two senses: first, the prior result has a linear dependence on information sets in the tree while ours only has a linear dependence on the number of information sets that a given player can experience during one path of play. Second, our bound weights the error by nature probabilities, which can decrease the bound exponentially in the number of nature branches on any given path from the root to a leaf node.

We then investigated computing CRSWF abstractions and showed that this is NP-complete. For single-level abstraction problems, we showed an equivalence between the problem of computing a bound-minimizing abstraction within our framework, and that of computing a distance-minimizing clustering under a class of objective functions; this lead to showing that clustering information sets with our error bound as the distance function yields a metric space when the rewards at the information sets are at the same scale. We also gave a counterexample, showing that clustering of information sets with different reward scaling does not satisfy the triangle inequality, and thus does not form a metric space with our distance function. Our metric space result immediately yields a 2-approximation algorithm for the single-level abstraction problem in games where information sets are distinguished by players' choice (not nature's). We also introduced a new class of objective functions for clustering. One extreme of this class is the maximum-diameter clustering problem. The other is a new natural objective function that minimizes the sum over all elements of maximum intra-cluster distances. Finally, our experiments showed that single-level abstraction problems can be solved and yield bounds that are at the same order of magnitude as the regrets after CFR convergence. We leave open the question of whether abstraction at multiple levels using our theory yields bounds of practical importance, and whether the bounds can be tightened such that the error has no dependence on tree height.

The perfect-recall results of Kroer and Sandholm [19] allow abstraction not only by merging information sets as we do but also by removing branches from the tree. The following approach can be adopted for bounded imperfect-recall abstraction with branch removal. First, a valid perfect-recall abstraction is computed, where the desired branches are removed. The results by Kroer and Sandholm [19] give bounds on the solution quality of equilibria computed in this abstraction. An imperfect-recall abstraction can then be computed from this perfect-recall abstraction, with our results providing bounds on solution quality for this step. Solution quality bounds can then be achieved for the final abstraction by taking the sum of the bounds for the two steps. It is likely that tighter bounds can be achieved by analyzing the distance between the original game and the final abstraction directly.

References

[1] Nicola Basilico and Nicola Gatti. "Automated Abstractions for Patrolling Security Games". In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. 2011.

[2] Darse Billings, Neil Burch, Aaron Davidson, Robert Holte, Jonathan Schaeffer, Terence Schauenberg, and Duane Szafron. "Approximating Game-Theoretic Optimal Strategies for Full-scale Poker". In: *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI)*. 2003.

[3] Noam Brown and Tuomas Sandholm. "Regret Transfer and Parameter Optimization". In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. 2014.

[4] Noam Brown and Tuomas Sandholm. "Simultaneous Abstraction and Equilibrium Finding in Games". In: *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. 2015.

[5] Noam Brown, Sam Ganzfried, and Tuomas Sandholm. "Hierarchical Abstraction, Distributed Equilibrium Computation, and Post-Processing, with Application to a Champion No-Limit Texas Hold'em Agent". In: Mimeo. 2014.

[6] Katherine Chen and Michael Bowling. "Tractable Objectives for Robust Policy Optimization". In: *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*. 2012.

[7] Tomás Feder and Daniel Greene. "Optimal algorithms for approximate clustering". In: *Proceedings of the Annual Symposium on Theory of Computing (STOC)*. 1988.

[8] Andrew Gilpin and Tuomas Sandholm. "A Competitive Texas Hold'em Poker Player via Automated Abstraction and Real-Time Equilibrium Computation". In: *Proceedings of the National Conference on Artificial Intelligence (AAAI)*. 2006, pp. 1007–1013.

[9] Andrew Gilpin and Tuomas Sandholm. "Better Automated Abstraction Techniques for Imperfect Information Games, with Application to Texas Hold'em Poker". In: *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*. 2007, pp. 1168–1175.

[10] Andrew Gilpin and Tuomas Sandholm. "Lossless Abstraction of Imperfect Information Games". In: *Journal of the ACM* 54.5 (2007). Early version in EC-06.

[11] Andrew Gilpin, Tuomas Sandholm, and Troels Bjerre Sørensen. "Potential-Aware Automated Abstraction of Sequential Games, and Holistic Equilibrium Analysis of Texas Hold'em Poker". In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. 2007.

[12] Andrew Gilpin, Tuomas Sandholm, and Troels Bjerre Sørensen. "A Heads-Up No-Limit Texas Hold'em Poker Player: Discretized Betting Models and Automatically Generated Equilibrium-Finding Programs". In: *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*. 2008.

[13] Teofilo F Gonzalez. "Clustering to minimize the maximum intercluster distance". In: *Theoretical Computer Science* 38 (1985).

[14] John Hawkin, Robert Holte, and Duane Szafron. "Automated action abstraction of imperfect information extensive-form games". In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. 2011.

[15] John Hawkin, Robert Holte, and Duane Szafron. "Using sliding windows to generate action abstractions in extensive-form games". In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. 2012.

[16] Albert Jiang and Kevin Leyton-Brown. "Polynomial-time computation of exact correlated equilibrium in compact games". In: *Proceedings of the ACM Conference on Electronic Commerce (EC)*. 2011.

[17] Michael Johanson, Neil Burch, Richard Valenzano, and Michael Bowling. "Evaluating State-Space Abstractions in Extensive-Form Games". In: *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*. 2013.

[18] Daphne Koller, Nimrod Megiddo, and Bernhard von Stengel. "Efficient Computation of Equilibria for Extensive Two-Person Games". In: *Games and Economic Behavior* (1996).

[19] Christian Kroer and Tuomas Sandholm. "Extensive-Form Game Abstraction With Bounds". In: *Proceedings of the ACM Conference on Economics and Computation (EC)*. 2014.

[20] Christian Kroer, Kevin Waugh, Fatma Kılınç-Karzan, and Tuomas Sandholm. "Faster First-Order Methods for Extensive-Form Game Solving". In: *Proceedings of the ACM Conference on Economics and Computation (EC)*. 2015.

[21] Marc Lanctot, Richard Gibson, Neil Burch, Martin Zinkevich, and Michael Bowling. "No-Regret Learning in Extensive-Form Games with Imperfect Recall". In: *International Conference on Machine Learning (ICML)*. 2012.

[22] Richard Lipton, Evangelos Markakis, and Aranyak Mehta. "Playing Large Games Using Simple Strategies". In: *Proceedings of the ACM Conference on Electronic Commerce (ACM-EC)*. 2003.

[23] Michael Littman and Peter Stone. "A polynomial-time Nash equilibrium algorithm for repeated games". In: *Proceedings of the ACM Conference on Electronic Commerce (ACM-EC)*. 2003.

[24] Tuomas Sandholm. "The State of Solving Large Incomplete-Information Games, and Application to Poker". In: *AI Magazine* (2010). Special issue on Algorithmic Game Theory, pp. 13–32.

[25] Tuomas Sandholm and Satinder Singh. "Lossy stochastic game abstraction with bounds". In: *Proceedings of the ACM Conference on Electronic Commerce (EC)*. 2012.

[26] Jiefu Shi and Michael Littman. "Abstraction Methods for Game Theoretic Poker". In: *CG '00: Revised Papers from the Second International Conference on Computers and Games*. 2002.

[27] Kevin Waugh, Martin Zinkevich, Michael Johanson, Morgan Kan, David Schnizlein, and Michael Bowling. "A Practical Use of Imperfect Recall". In: *Symposium on Abstraction, Reformulation and Approximation (SARA)*. 2009.

[28] Kevin Waugh, David Schnizlein, Michael Bowling, and Duane Szafron. "Abstraction Pathologies in Extensive Games". In: *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*. 2009.

[29] Kevin Waugh, Dustin Morrill, J. Andrew Bagnell, and Michael Bowling. "Solving Games with Functional Regret Estimation". In: *AAAI Conference on Artificial Intelligence*. 2015.

[30] Martin Zinkevich, Michael Bowling, Michael Johanson, and Carmelo Piccione. "Regret Minimization in Games with Incomplete Information". In: *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*. 2007.

A Proof of Proposition 1

Proof. Given some I' such that $\pi^\sigma_{-i}(I') > 0$, we assume that $\pi^\sigma_i(I') > 0$. For information sets where this is not the case, we assume any distribution over the choices of Player i leading to I'. Note that other players cannot affect the distribution over $\mathcal{P}(I')$ due to Condition 1 of Definition 1. By the definition of regret of an action, we have:

$$r(I', a^*) = W^{\sigma_{I\to a^*}}(I') - W^\sigma(I')$$

$$= \sum_{s'\in I'} \frac{\pi^\sigma(s')}{\pi^\sigma(I')} \sum_{z'\in Z_{t^{s'}_{a^*}}} \pi^\sigma(t^{s'}_{a^*} z') u_i(z') - \sum_{s'\in I'} \frac{\pi^\sigma(s')}{\pi^\sigma(I')} \sum_{a\in A_I} \pi^\sigma(I', a) \sum_{z'\in Z_{t^{s'}_a}} \pi^\sigma(t^{s'}_a, z') u_i(z')$$

$$= \sum_{\breve{I}\in\mathcal{P}(I')} \sum_{s'\in\breve{I}} \frac{\pi^\sigma(s')}{\pi^\sigma(I')} \left(\sum_{z'\in Z_{t^{s'}_{a^*}}} \pi^\sigma(t^{s'}_{a^*}, z') u_i(z') - \sum_{a\in A_I} \pi^\sigma(I', a) \sum_{z'\in Z_{t^{s'}_a}} \pi^\sigma(t^{s'}_a, z') u_i(z') \right)$$

Note that $\sum_{\breve{I}\in\mathcal{P}(I')} \sum_{s'\in\breve{I}} \frac{\pi^\sigma(s')}{\pi^\sigma(I')} = \sum_{\breve{I}\in\mathcal{P}(I')} \frac{\pi^\sigma(\breve{I})}{\pi^\sigma(I')}$ sums over a probability distribution on $\mathcal{P}(I')$. We take the minimum over this distribution:

$$\geq \min_{\breve{I}\in\mathcal{P}(I')} \sum_{s'\in\breve{I}} \frac{\pi^\sigma(s')}{\pi^\sigma(\breve{I})} \left(\sum_{z'\in Z_{t^{s'}_{a^*}}} \pi^\sigma(t^{s'}_{a^*}, z') u_i(z') - \sum_{a\in A_I} \pi^\sigma(I', a) \sum_{z'\in Z_{t^{s'}_a}} \pi^\sigma(t^{s'}_a, z') u_i(z') \right)$$

Let $I_m = \arg\min_{\breve{I}\in\mathcal{P}(I')}$ be the minimizer. Now we can bound the value using the reward approximation error term:

$$= \sum_{s'\in I_m} \frac{\pi^\sigma(s')}{\pi^\sigma(I_m)} \left(\sum_{z'\in Z_{t^{s'}_{a^*}}} \pi^\sigma(t^{s'}_{a^*}, z') u_i(z') - \sum_{a\in A_I} \pi^\sigma(I', a) \sum_{z'\in Z_{t^{s'}_a}} \pi^\sigma(t^{s'}_a, z') u_i(z') \right)$$

$$\geq \frac{1}{\delta_{I,I_m}} \left(\sum_{s'\in I_m} \frac{\pi^\sigma(s')}{\pi^\sigma(I_m)} \left(\sum_{z'\in Z_{t^{s'}_{a^*}}} \pi^\sigma(t^{s'}_{a^*}, z') u_i(\phi_{I_m,I}(z')) \right.\right.$$
$$\left.\left. - \sum_{a\in A_I} \pi^\sigma(I', a) \sum_{z'\in Z_{t^{s'}_a}} \pi^\sigma(t^{s'}_a, z') u_i(\phi_{I_m,I}(z')) \right) - \frac{1}{\delta_{I,I_m}} 2\epsilon^R_{I,I_m}(s') \right)$$

Multiplying both sides by δ_{I,I_m} gives

$$\delta_{I,I_m} r(I', a^*) \geq \sum_{s'\in I_m} \frac{\pi^\sigma(s')}{\pi^\sigma(I_m)} \left(\sum_{z'\in Z_{t^{s'}_{a^*}}} \pi^\sigma(t^{s'}_{a^*}, z') u_i(\phi_{I_m,I}(z')) \right.$$
$$\left. - \sum_{a\in A_I} \pi^\sigma(I', a) \sum_{z'\in Z_{t^{s'}_a}} \pi^\sigma(t^{s'}_a, z') u_i(\phi_{I_m,I}(z')) - 2\epsilon^R_{I,I_m}(s') \right)$$

From now on, let $s' = z'[I_m]$, $s = \phi_{I_m,I}(z')[I]$, and similarly $z = \phi_{I_m,I}(z')$. Now we can apply the distribution approximation error:

$$\geq \sum_{s' \in I_m} \left(\frac{\pi^\sigma(s)}{\pi^\sigma(I)} - \epsilon^D_{I,I_m}(s) \right) \left(\sum_{z' \in Z_{t^{s'}_{a^*}}} \pi^\sigma(t^{s'}_{a^*}, z') u_i(\phi_{I_m,I}(z')) \right.$$
$$\left. - \sum_{a \in A_I} \pi^\sigma(I', a) \sum_{z' \in Z_{t^{s'}_a}} \pi^\sigma(t^{s'}_a, z') u_i(\phi_{I_m,I}(z')) - 2\epsilon^R_{I,I_m}(s') \right)$$

For all $a \in A_I$, $\sum_{s' \in \breve{I}} \sum_{z' \in Z_{t^{s'}_a}}$ can be rewritten as the sum $\sum_{s \in I} \sum_{z \in Z_{t^s_a}}$ as Condition 2 of Definition 1 ensures that if $(\breve{I}, a)$ is on the path to z, then (I, a) is on the path to $\phi_{I_m,I}(z)$. This gives us

$$= \sum_{s \in I} \left(\frac{\pi^\sigma(s)}{\pi^\sigma(I)} - \epsilon^D_{I,I_m}(s) \right) \left(\sum_{z \in Z_{t^s_{a^*}}} \pi^\sigma_0(t^{s'}_{a^*}, z') u_i(z) \right.$$
$$\left. - \sum_{a \in A_I} \pi^\sigma(I', a) \sum_{z \in Z_{t^s_a}} \pi^\sigma_0(t^{s'}_a, z') u_i(z) - 2\epsilon^R_{I,I_m}(s) \right)$$
$$\geq \sum_{s \in I} \frac{\pi^\sigma(s)}{\pi^\sigma(I)} \left(\sum_{z \in Z_{t^s_{a^*}}} \pi^\sigma_0(t^{s'}_{a^*}, z') u_i(z) \right.$$
$$\left. - \sum_{a \in A_I} \pi^\sigma(I', a) \sum_{z \in Z_{t^s_a}} \pi^\sigma_0(t^{s'}_a, z') u_i(z) - 2\epsilon^R_{I,I_m}(s) \right) - \epsilon^D_{I,I_m}$$

We rewrite the summation over $Z_{t^s_a}$ so that we first sum over the possible sequences of actions $X^b_{-0}(t^s_a) = X^b_{-0}(t^{s'}_a)$ players excluding nature can take. We then sum over the possible sequences of actions $X_0(t^{s'}_{\boldsymbol{a}})$ nature can take for the chosen sequence $\boldsymbol{a}$. Since this uniquely specifies leaf nodes, we can treat elements of this summation as such. Call this set $Z^{\boldsymbol{a}}_s$. For any such node s and leaf node z, $\pi^\sigma(s, z) = \pi^\sigma_{-0}(\boldsymbol{a})\pi_0(s, z)$. We use this observation along with the transition approximation error to get

$$\geq \sum_{s \in I} \frac{\pi^\sigma(s)}{\pi^\sigma(I)} \left(\sum_{\boldsymbol{a}^* \in X^b_{-0}(t^s_{a^*})} \pi^\sigma(\boldsymbol{a}^*) \sum_{z \in Z^{\boldsymbol{a}^*}_s} \pi^\sigma_0(t^s_{a^*}, z) u_i(z) \right.$$
$$\left. - \sum_{\boldsymbol{a} \in X^b_{-0}(s)} \pi^\sigma(\boldsymbol{a}) \sum_{z \in Z^{\boldsymbol{a}}_s} \pi^\sigma_0(t^s_a, z) u_i(z) - 2\epsilon^0_{I,I_m}(s) - 2\epsilon^R_{I,I_m}(s) \right) - \epsilon^D_{I,I_m}$$

Rearranging terms gives us that this is exactly equal to

$$
\begin{aligned}
&= V^{\sigma_{I\to a^*}}(I) - V^{\sigma}(I) - 2\sum_{s\in I}\frac{\pi^{\sigma}(s)}{\pi^{\sigma}(I)}\left(\epsilon^{0}_{I,I_m}(s) + \epsilon^{R}_{I,I_m}(s)\right) - \epsilon^{D}_{I,I_m} \\
&= r(I,a^*) - 2\sum_{s\in I}\frac{\pi^{\sigma}(s)}{\pi^{\sigma}(I)}\left(\epsilon^{0}_{I,I_m}(s) + \epsilon^{R}_{I,I_m}(s)\right) - \epsilon^{D}_{I,I_m}
\end{aligned}
$$

Summarizing, this gives us

$$
\begin{aligned}
\delta_{I,I_m} r(I',a^*) &\geq r(I,a^*) - 2\sum_{s\in I}\frac{\pi^{\sigma}(s)}{\pi^{\sigma}(I)}\left(\epsilon^{0}_{I,I_m}(s) + \epsilon^{R}_{I,I_m}(s)\right) - \epsilon^{D}_{I,I_m} \\
\Leftrightarrow r(I,a^*) &\leq \delta_{I,I_m} r(I',a^*) + 2\sum_{s\in I}\frac{\pi^{\sigma}(s)}{\pi^{\sigma}(I)}\left(\epsilon^{0}_{I,I_m}(s) + \epsilon^{R}_{I,I_m}(s)\right) + \epsilon^{D}_{I,I_m} \\
&\leq \max_{\breve{I}\in\mathcal{P}(I')} \delta_{I,\breve{I}} r(I',a^*) + 2\sum_{s\in I}\frac{\pi^{\sigma}(s)}{\pi^{\sigma}(I)}\left(\epsilon^{0}_{I,\breve{I}}(s) + \epsilon^{R}_{I,\breve{I}}(s)\right) + \epsilon^{D}_{I,\breve{I}}
\end{aligned}
$$

which completes the proof.

B Proof of Theorem 1

Proof. Consider some alternative strategy σ^* where Player i deviates to a best response and $\sigma_{-i} = \sigma^*_{-i}$. We prove the bound by induction over the levels $\mathcal{H}_i$ belonging to Player i. For the base case, consider any abstract information set $I' \in \mathcal{I}'_i$ and any $I \in \mathcal{P}(I')$ at the lowest level l in $\mathcal{H}_i$. We know that no mixed strategy is better than the single best action when the strategies of the other players are held constant. This fact and Proposition 1 gives us that:

$$
\begin{aligned}
V^{\sigma^*}(I) &\leq \max_{a\in A_I} V^{\sigma_{I\to a}}(I) \\
&\leq V^{\sigma}(I) + \max_{\breve{I}\in\mathcal{P}(I')} \delta_{I,\breve{I}} r(I') + 2\sum_{s\in I}\frac{\pi^{\sigma}(s)}{\pi^{\sigma}(I)}\left(\epsilon^{0}_{I,\breve{I}}(s) + \epsilon^{R}_{I,\breve{I}}(s)\right) + \epsilon^{D}_{I,\breve{I}}
\end{aligned}
$$

For the inductive step, we assume the following holds for all information sets I at heights $l < k \in \mathcal{H}_i$:

$$
V^{\sigma^*}(I) \leq V^{\sigma}(I) + \max_{\boldsymbol{a}\in X^b_i(I)} \sum_{j\in\mathcal{H}_i, j\leq l} \sum_{\hat{I}\in\mathcal{D}^{\boldsymbol{a},j}_I} \frac{\pi^{\sigma}_{-i}(\hat{I})}{\pi^{\sigma}_{-i}(I)}\psi(\hat{I})
$$

$$
\psi(\hat{I}) = \left(\max_{\breve{I}\in\mathcal{P}(f_{\hat{I}})} \delta_{\hat{I},\breve{I}} r(f_{\hat{I}}) + 2\sum_{s\in\hat{I}}\frac{\pi^{\sigma}(s)}{\pi^{\sigma}(\hat{I})}\left(\epsilon^{0}_{\hat{I},\breve{I}}(s) + \epsilon^{R}_{\hat{I},\breve{I}}(s)\right) + \epsilon^{D}_{\hat{I},\breve{I}}\right) \tag{2}
$$

Now consider some information set I at height k. We use Equation 1 to write the value of an information set, and apply the inductive assumption:

$$
\begin{aligned}
V^{\sigma^*}(I) &= \sum_{a\in A_I} \sigma^*(I,a) \sum_{\hat{I}\in\mathcal{D}_I^a} \frac{\pi_{-i}^{\sigma}(\hat{I})}{\pi_{-i}^{\sigma}(I)} V^{\sigma^*}(\hat{I}) \\
&\leq \sum_{a\in A_I} \sigma^*(I,a) \sum_{\hat{I}\in\mathcal{D}_I^a} \frac{\pi_{-i}^{\sigma}(\hat{I})}{\pi_{-i}^{\sigma}(I)} \left(V^{\sigma}(\hat{I}) + \max_{\boldsymbol{a}\in X_i^b(\hat{I})} \sum_{j\in\mathcal{H}_i, j<k} \sum_{\grave{I}\in\mathcal{D}_{\hat{I}}^{\boldsymbol{a},j}} \frac{\pi_{-i}^{\sigma}(\grave{I})}{\pi_{-i}^{\sigma}(\hat{I})} \psi(\grave{I}) \right) \\
&\leq \max_{a\in A_I} \sum_{\hat{I}\in\mathcal{D}_I^a} \frac{\pi_{-i}^{\sigma}(\hat{I})}{\pi_{-i}^{\sigma}(I)} V^{\sigma}(\hat{I}) + \max_{\boldsymbol{a}\in X_i^b(I)} \sum_{j\in\mathcal{H}_i, j<k} \sum_{\grave{I}\in\mathcal{D}_I^{\boldsymbol{a},j}} \frac{\pi_{-i}^{\sigma}(\grave{I})}{\pi_{-i}^{\sigma}(I)} \psi(\grave{I})
\end{aligned}
$$

The last inequality is obtained by taking the maximum over A_I, splitting the terms, and multiplying in $\frac{\pi_{-i}^{\sigma}(\hat{I})}{\pi_{-i}^{\sigma}(I)}$. Now we can apply Proposition 1 to bound the immediate regret:

$$
\begin{aligned}
&\leq V^{\sigma}(I) + \max_{\breve{I}\in\mathcal{P}(f_I)} \delta_{I,\breve{I}} r(I', a^*) + 2\sum_{s\in I} \frac{\pi^{\sigma}(s)}{\pi^{\sigma}(I)} \left(\epsilon_{I,\breve{I}}^{0}(s) + \epsilon_{I,\breve{I}}^{R}(s) \right) + \epsilon_{I,\breve{I}}^{D} \\
&\quad + \max_{\boldsymbol{a}\in X_i^b(I)} \sum_{j\in\mathcal{H}_i, j<k} \sum_{\grave{I}\in\mathcal{D}_I^{\boldsymbol{a},j}} \frac{\pi_{-i}^{\sigma}(\grave{I})}{\pi_{-i}^{\sigma}(I)} \psi(\grave{I}) \\
&= V^{\sigma}(I) + \psi(I) + \max_{\boldsymbol{a}\in X_i^b(I)} \sum_{j\in\mathcal{H}_i, j<k} \sum_{\grave{I}\in\mathcal{D}_I^{\boldsymbol{a},j}} \frac{\pi_{-i}^{\sigma}(\grave{I})}{\pi_{-i}^{\sigma}(I)} \psi(\grave{I}) \\
&= V^{\sigma}(I) + \max_{\boldsymbol{a}\in X_i^b(I)} \sum_{j\in\mathcal{H}_i, j\leq k} \sum_{\grave{I}\in\mathcal{D}_I^{\boldsymbol{a},j}} \frac{\pi_{-i}^{\sigma}(\grave{I})}{\pi_{-i}^{\sigma}(I)} \psi(\grave{I})
\end{aligned}
$$

This gives a bound on the regret at any information set I. Taking the regret at the root node gives the desired result.

C Proof of Theorem 2

Proof. Assume that we are given a strategy σ that is a Nash equilibrium in Γ', and a strategy $\sigma^* = (\sigma_{-i}, \sigma_i^*)$ where Player i best responds in Γ. For information sets I' where $\sigma_{-i}(I') > 0, \sigma(I') = 0$, a Nash equilibrium does not put any constraints on behavior. However, we know that Player i could have played a strategy satisfying the self-trembling property. Assume any such strategy σ^{ST}, where it is equal to σ everywhere except at such information sets, where a utility-maximizing strategy is played for some arbitrary, fixed distribution over $\mathcal{P}(I')$. We can then apply Theorem 1 to get the following (where $\psi(I)$ is defined as in Equation 2):

$$V_i^{\sigma^*}(r) \leq V_i^{\sigma^{ST}}(r) + \max_{\boldsymbol{a} \in X_i^b(r)} \sum_{j \in \mathcal{H}_i} \sum_{\grave{I} \in \mathcal{D}_r^{\boldsymbol{a},j}} \frac{\pi_{-i}^{\sigma}(\grave{I})}{\pi_{-i}^{\sigma}(I)} \psi(\grave{I})$$

Where all regrets $r(I', a^*) = 0$ since σ^{ST} is a Nash equilibrium. Now, we observe that the utility is the same for σ and any σ^{ST} at the root node, $V_i^{\sigma^{ST}}(r) = V_i^{\sigma}(r)$:

$$V_i^{\sigma^*}(r) \leq V_i^{\sigma}(r) + \max_{\boldsymbol{a} \in X_i^b(r)} \sum_{j \in \mathcal{H}_i} \sum_{\grave{I} \in \mathcal{D}_r^{\boldsymbol{a},j}} \frac{\pi_{-i}^{\sigma}(\grave{I})}{\pi_{-i}^{\sigma}(I)} \psi(\grave{I})$$

which is the result we wanted.

D Proof of Theorem 3

Proof. Consider the *two-dimensional k-center clustering decision problem* with the L_q distance metric. It is defined as follows: given a set $P = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ of n points in the plane, and an integer k, does there exist a partition of P into k clusters $\mathcal{C} = \{c_1, \ldots, c_k\}$ such that the maximum distance $\|p - p'\|_q \leq c$ between any pair of points p, p' in the same cluster is minimized. This problem is NP-hard to approximate within a factor of 2 for $q = \infty$, amongst others. [7].

Given such a problem, we construct a perfect-recall game as follows. For each point $p \in P$, we construct an information set I_p. We insert two nodes s_p^x, s_p^y in each information set I_p, representing the dimensions x, y respectively. All these nodes descend directly from the root node r, where Player 1 acts. At each information set we have two actions, a_c, a_v. For any point p, we add leaf nodes at the branch a_c with payoff $M, 2M$ at the nodes s_p^x, s_p^y respectively. If we pick a sufficiently large M, this ensures that for any two points p, p', their nodes $s_p^x, s_{p'}^x$ will map to each other, and similarly for y. This also ensures that the scaling variable has to be set to 1 for all information set mappings. For the branches a_v, we add leaf nodes with utility equal to the x, y coordinate of p at the s_p^x, s_p^y nodes respectively.

There is a one-to-one mapping between clusterings of the points P and partitions of the information sets $\{I_p : p \in P\}$. The quality of a clustering is

$$\max_{z \in \{x,y\}} \max_{j=1,\ldots,k} \max_{p,p' \in c_j} |p(z) - p'(z)|$$

. Since Player 1 acts at r, the abstraction quality bound is equal to the maximum difference over any two leaf nodes mapped to each other, as $\epsilon^0 = \epsilon^D = 0$. This is the same as the quality measure of the clustering. Thus, an optimal k size clustering is equivalent to an optimal k information set abstraction.

Given some CRSWF abstraction, verifying the solution is easy to do: in one top-down traversal of the game tree, compute the node distributions at each information set. For each full-game information set, this gives the distribution-approximation error. For each information set pair mapped to each other, the transition- and reward-approximation error can now be computed by a single traversal of the two. Thus the problem is in NP.

E Proof of Proposition 2

Proof. The first condition follows from the other three. Condition 2, identity of indiscernibles, does not hold for information sets. However, any pair of information sets with distance zero can be merged losslessly in preprocessing, thus rendering the condition true (having distance zero is transitive, so the minimal preprocessing solution is unique). Condition 3, symmetry, holds by definition, since our distance metric is defined as the error incurred from merging two information sets, which considers the error from both directions of the mapping.

Finally, we show that Condition 4, the triangle inequality holds. Consider any three information sets $I_1, I_2, I_3 \in \mathcal{I}^m$. We need to show that $d(I_1, I_3) \leq d(I_1, I_2) + d(I_2, I_3)$. Let $\phi_{I_1,I_2}, \phi_{I_2,I_3}$ be the mappings for I_1, I_2 and I_2, I_3 respectively. We construct a mapping $\phi_{I_1,I_3} = \phi_{I_2,I_3} \circ \phi_{I_1,I_2}$ and show that it satisfies the triangle inequality. For the leaf payoff error, since $\delta_{I_1,I_2} = \delta_{I_2,I_3} = 1$, at any leaf $z \in Z_{I_1}$ we get:

$$u_i(z) \leq u_i(\phi_{I_1,I_2}(z)) + \epsilon_{I_1,I_2}(z) \leq u_i(\phi_{I_2,I_3}(\phi_{I_1,I_2}(z))) + \epsilon_{I_2,I_3}(\phi_{I_1,I_2}(z)) + \epsilon_{I_1,I_2}(z)$$

For the nature leaf probability error we can apply the same reasoning:

$$\begin{aligned}
&\pi_0^\sigma(z[I_1], z) \\
\leq\ &\pi_0^\sigma(\phi_{I_1,I_2}(z[I_1]), \phi_{I_1,I_2}(z)) + \epsilon^0_{I_1,I_2} \\
\leq\ &\pi_0^\sigma(\phi_{I_2,I_3}(\phi_{I_1,I_2}(z[I_1])), \phi_{I_2,I_3}(\phi_{I_1,I_2}(z))) + \epsilon^0_{I_2,I_3}(\phi_{I_1,I_2}(z)) + \epsilon^0_{I_1,I_2}(z)
\end{aligned}$$

Again, we derive the distribution error using a similar approach:

$$\begin{aligned}
&\frac{\pi_0(z[I_1])}{\pi_0(I_1)} \\
\leq\ &\frac{\pi_0(\phi_{I_1,I_2}(z[I_1]))}{\pi_0(I_2)} + \epsilon^D_{I_1,I_2}(z[I_1]) \\
\leq\ &\frac{\pi_0(\phi_{I_2,I_3}(\phi_{I_1,I_2}(z[I_1])))}{\pi_0(I_3)} + \epsilon^D_{I_2,I_3}(\phi_{I_2,I_3}(z[I_1])) + \epsilon^D_{I_1,I_2}(z[I_1])
\end{aligned}$$

This completes the proof.