Learning Hedonic Games

Jakub Sliwinski and Yair Zick

Abstract

Coalitional stability in hedonic games has usually been considered in the setting where agent preferences are fully known. We consider the setting where agent preferences are unknown; we lay the theoretical foundations for studying the interplay between coalitional stability and (PAC) learning in hedonic games. We introduce the notion of *PAC stability* — the equivalent of core stability under uncertainty — and examine the PAC stabilizability and learnability of several popular classes of hedonic games.

1 Introduction

Hedonic games [18] are often used to model agent interactions in collaborative environments. They describe a simple coalition formation process: agents express their preferences over coalitions; next, a central designer is interested in finding a *coalition structure* (i.e. a partition of the agents) that satisfies certain desiderata. Recent years have seen considerable efforts by the computational social choice community to find novel algorithms computing hedonic solution concepts — i.e. functions mapping hedonic games to (sets of) coalition structures guaranteed to satisfy certain properties. The core is a central solution concept in hedonic games. A coalition structure is *core stable* (or simply stable) if no subset of players can deviate, i.e. have all its members strictly better off should it form. Coalitional stability is a highly desirable property; there is a rich literature studying the core of hedonic games, establishing the existence (or lack thereof) of core outcomes for various classes of hedonic games. It is usually assumed that the agents' preferences are fully known in order to identify a stable coalition structure; this is a major obstacle in applying the research, as in many real-world scenarios full knowledge of the agent preferences cannot be assumed. There has been little work identifying hedonic solution concepts under *uncertainty.* This is where our work comes in; we study the following problem:

> *Can we find a coalition structure that is likely to be resistant to deviation, given limited knowledge about the underlying hedonic game?*

We lay the theoretical foundations for studying the interplay between two fundamental concepts: *coalitional stability*, and *PAC learning* in hedonic games. PAC learning is a canonical model that studies how good probabilistic function approximations can be derived from a (polynomial) number of samples (see Section 2.3 for details).

1.1 Our Contributions

We contrast between two objectives: learning the underlying hedonic game — i.e. provide an estimate for agent utilities over coalitions — and learning a stable outcome — i.e. provide a coalition structure that is likely to be resistant against coalitional deviations. We do this for a variety of classes of hedonic games (see Table 1). Our results show that hedonic games exhibit surprisingly diverse behavior: learnability of the underlying hedonic game does not necessarily guarantee one's ability to learn a stable outcome; conversely, there exist classes of hedonic games for which a PAC stable outcome can be efficiently computed, though the underlying hedonic game cannot be approximated in polynomial time. This result stands in stark contrast to recent results on coalitional games with transferable utility (TU cooperative

Hedonic Games	**Learnable**	**Stabilizable**
Additively Separable	✓	✗
Fractional	✓	✗
$\mathcal{W}$-games	✓	✗
$\mathcal{B}$-games	✓	✓
Top Responsive	✗	✓

Table 1: A summary of this paper's learnability/stabilizability results. For stabilizability, informativity is assumed, as explained in section 4.4.

games) [7]: in the TU domain, *any* coalitional game can be PAC stabilized, irrespective of the underlying function class.

1.2 Related Work

Hedonic games have been widely studied in the AI/MAS community; most works establish existential and computational properties of solution concepts for various classes of hedonic games [22, 3, 11, 17] (see [5] or [25] for an overview). Uncertainty in TU cooperative games has been studied extensively; some works take a Bayesian approach: assuming a known prior over agent abilities, and finding solution concepts under this regime (see [15, 16, 21]). However, the work that most closely mirrors our approach is by [7]; [7] use a PAC learning approach to learn both stable payoff divisions, and the underlying game. The PAC approach has been subsequently applied to other game-theoretic domains, such as learning combinatorial functions from pairwise comparisons [9], buyer valuations in combinatorial domains [6, 8], and adversary models in security games [24].

2 Preliminaries

Let $N = \{1, \ldots, n\}$ be a set of *agents* or *players*. A *coalition* is a nonempty subset of N; let $\mathcal{N}_i = \{S : S \subseteq N, i \in S\}$ be the set of all coalitions containing player i. A *coalition structure* is a partition π of N; we refer to the coalition containing i in π as $\pi(i)$. A *hedonic game* is a pair $\langle N, \succeq \rangle$, where $\succeq = (\succeq_1, \ldots, \succeq_n)$ is a preference list, reflecting each player i's preferences over the coalitions she would like to join. In other words, $\succeq_i$ is a reflexive, complete, and transitive binary relation over $\mathcal{N}_i$; given $S, T \in \mathcal{N}_i$, we say that $S \simeq_i T$ if $S \succeq_i T$ and $T \succeq_i S$.

2.1 The Core of a Hedonic Game

We say that a coalition S *core blocks* a coalition structure π, if every agent $i \in S$ strictly prefers S to $\pi(i)$, i.e. $S \succ_i \pi(i)$. A coalition structure π is said to be *core stable* if there is no coalition that core blocks π; that is, for every $S \subseteq N$, there exists at least one player $i \in S$ such that $\pi(i) \succeq_i S$. The set of core stable coalition structures is called the *core* of $\langle N, \succeq \rangle$, denoted $Core(N, \succeq)$.

2.2 Representing Hedonic Games

Hedonic games are represented by (complete) preference orders of players over subsets. In this work, we are making the explicit assumption that player utilities are given numerically, rather than by ranked preferences, thereby exploiting the full strength of the classic PAC learning model; hence we will use $v_i(S) \geq v_i(T)$ and $S \succeq_i T$ interchangeably.

A class of hedonic games can be represented by many possible utility functions; as we later show, the choice of representation functions plays an important role in the learnability of hedonic games. One can 'cheat' by encoding useful information in the representation function itself. For example, one can encode preferences by integer values, and other valuable information (e.g. a stable partition, the actual game) in the insignificant fractional terms. In what follows, we focus on functions that naturally encode agent preferences, without providing additional information about the underlying game.

2.3 PAC Learning

We briefly describe the PAC learning model; for a detailed exposition, see [20, 23]. The purpose of the Probably Approximately Correct (PAC) learning model is finding good function approximations. We are given some unknown function v belonging to some *hypothesis class* $\mathcal{H}$; $\mathcal{H}$ is a family of functions from subsets of players to $\mathbb{R}$. We let $\mathcal{H}_n$ be the restriction of $\mathcal{H}$ to functions over n players. We refer to the unknown function as the *target concept*, and the proposed approximation will be called a *hypothesis*.

An algorithm $\mathcal{A}$ that *PAC learns* $\mathcal{H}$ takes as input an error parameter $\varepsilon > 0$, a confidence parameter $\delta > 0$, and m samples $(\langle S_1, v(S_1)\rangle, ..., \langle S_m, v(S_m)\rangle)$; $S_1, \dots, S_m$ are assumed to be i.i.d. sampled from a distribution $\mathcal{D}$ over 2^N. We are guaranteed that with probability at least $1-\delta$, $\mathcal{A}$ outputs a hypothesis $v^* \in \mathcal{H}_n$ (here both the target concept and the hypothesis are assumed to be in $\mathcal{H}_n$) such that $\Pr_{S \sim \mathcal{D}}[v^*(S) \neq v(S)] < \varepsilon$. Intuitively, assuming that all observed samples were sampled i.i.d. from $\mathcal{D}$, it is very likely that the algorithm will output a hypothesis v^* accurately predicting the values of other sets sampled from $\mathcal{D}$. The class $\mathcal{H}$ is *efficiently PAC learnable* if there is some algorithm $\mathcal{A}$ that can PAC learn any $v \in \mathcal{H}$; furthermore, both the running time of $\mathcal{A}$ and m — the number of samples that it requires as input — are polynomial in n, $\frac{1}{\varepsilon}$ and $\log \frac{1}{\delta}$.

Key results in learning theory relate the inherent complexity of the hypothesis class to its PAC learnability. The VC dimension [20] is a canonical complexity measure for binary functions (i.e. whose outputs are in $\{0, 1\}$). For classes of real-valued functions, the analogous concept of *pseudo-dimension* [2, Chapter 11] is defined: we are given a list of sets $S_1, \dots, S_m \subseteq N$, and a set of real values $r_1, \dots, r_m \in \mathbb{R}$; a class of functions $\mathcal{H}$ can pseudo-shatter $(\langle S_1, r_1\rangle, \dots, \langle S_m, r_m\rangle)$ if for any possible labeling $\ell_1, \dots, \ell_m \in \{0, 1\}$ there exists some function $f \in \mathcal{H}$ such that $f(S_j) > r_j \iff \ell_j = 1$. The pseudo-dimension of $\mathcal{H}$, denoted $P_{dim}(\mathcal{H})$ is the maximal value m for which there exists a list of m set-value pairs that can be shattered by $\mathcal{H}$. We will make use of the following theorem:

Theorem 2.1 ([2])**.** *A hypothesis class $\mathcal{H}$ is (ε, δ) PAC learnable using m samples, where m is polynomial in $P_{dim}(\mathcal{H})$, $\frac{1}{\varepsilon}$ and $\log \frac{1}{\delta}$ by giving a hypothesis v^* consistent with the sample, i.e. $v^*(S_i) = v(S_i)$ for all i. Furthermore, if $P_{dim}(\mathcal{H})$ is superpolynomial in n, $\mathcal{H}$ is not PAC learnable.*

We are interested in finding a stable partition for a hedonic game $\mathcal{G} = \langle N, \succeq\rangle$; however, we do not know anything about players' preferences. Barring any other information, the task is virtually impossible. One natural compromise would be to offer a partition that is *likely* to be stable under some assumptions. More formally, we are given a set of m *samples*, i.e. m pairs $\langle S_j, \vec{v}(S_j)\rangle$; here, $S_j \subseteq N$ and $\vec{v}(S_j) = (v_i(S_j))_{i \in S_j}$ are the values assigned to S_j by the members of S_j. In what follows, we often refer to the hedonic game $\mathcal{G}$ as $\langle N, \vec{v}\rangle$, identifying agent preference orders with their utility function. Within this framework, we are interested in the following question: given a hedonic game $\mathcal{G} = \langle N, \vec{v}\rangle$ belonging to some class of hedonic games $\mathcal{H}$, can we find a partition that is *likely* to be stable, or output that no such partition exists?

3 PAC Stability in Hedonic Games

We now present a learning-theoretic notion of core stability, based on the PAC framework. We say that an algorithm $\mathcal{A}$ can *PAC stabilize* a class of hedonic games, if after seeing some examples, it is able to propose a partition that is unlikely to be core blocked by a coalition sampled from $\mathcal{D}$. More formally, given $\varepsilon > 0$, a partition π is *ε-PAC stable under $\mathcal{D}$* if

$$\Pr_{S \sim \mathcal{D}}[S \text{ core blocks } \pi] < \varepsilon.$$

Given an unknown hedonic game $\mathcal{G} = \langle N, \vec{v} \rangle$ belonging to some hypothesis class $\mathcal{H}$, a *PAC stabilizing algorithm* $\mathcal{A}$ takes as input m samples $(\langle S_1, \vec{v}(S_1) \rangle, \dots, \langle S_m, \vec{v}(S_m) \rangle)$, where $S_1, \dots, S_m \subseteq N$, and parameters $\varepsilon, \delta > 0$. The algorithm $\mathcal{A}$ *PAC stabilizes* $\mathcal{H}$, if for any hedonic game $\langle N, \vec{v} \rangle \in \mathcal{H}$, distribution $\mathcal{D}$ over 2^N, and parameters $\varepsilon, \delta > 0$, with probability at least $1 - \delta$, $\mathcal{A}$ outputs an ε-PAC stable coalition structure, or reports that the core is empty. We again require that the number of samples, m, is bounded by a polynomial in n, $\frac{1}{\varepsilon}$ and $\log \frac{1}{\delta}$. We similarly say that $\mathcal{H}$ is *PAC stabilizable* if there is some algorithm $\mathcal{A}$ that PAC stabilizes $\mathcal{H}$.

4 Learnability and Stabilizability of Hedonic Games

Learnability of a class of hedonic games alone does not imply stabilizability; one might think that the following procedure might work: PAC learn a hypothesis $\vec{v}^*$ for $\vec{v}$, compute a stable partition π^* for $\langle N, \vec{v}^* \rangle$, and hope that π^* PAC stabilizes $\langle N, \vec{v} \rangle$. Unfortunately, this will not necessarily work; intuitively, when learning a game, we only provide value guarantees with respect to coalitions sampled from $\mathcal{D}$. However, stable partitions for v^* can inevitably contain coalitions not sampled from $\mathcal{D}$, for which we have no information whatsoever. However, if one can ensure that the values of coalitions in the proposed partition are not overestimated, then core stability of that partition for the learned hypothesis will mean ε-PAC stability under $\mathcal{D}$ for the underlying game, as formalized in Proposition 4.1.

Proposition 4.1. *Given a hedonic game $\mathcal{G} = \langle N, v \rangle$ and v^* be a PAC approximation of v. If π is stable for $\langle N, v^* \rangle$ and $\forall i \in N$, $v_i^*(\pi(i)) \leq v_i(\pi(i))$ then π PAC stabilizes $\mathcal{G}$.*

Proof. We observe the following inequalities

$$\begin{aligned}
\Pr_{S \sim \mathcal{D}}[S \text{ blocks } \pi] &= \Pr_{S \sim \mathcal{D}}[S \text{ blocks } \pi \wedge v^*(S) = v(S)] + \Pr_{S \sim \mathcal{D}}[S \text{ blocks } \pi \wedge v^*(S) \neq v(S)] && (1)\\
&\leq \Pr_{S \sim \mathcal{D}}[S \text{ blocks } \pi \wedge v^*(S) = v(S)] + \epsilon && (2)\\
&= \Pr_{S \sim \mathcal{D}}[\forall i \in S : v_i^*(S) > v_i(\pi(i))] + \epsilon && (3)\\
&\leq \Pr_{S \sim \mathcal{D}}[\forall i \in S : v_i^*(S) > v_i^*(\pi(i))] + \epsilon && (4)\\
&= 0 + \epsilon = \epsilon && (5)
\end{aligned}$$

Line 2 holds since v^* is a PAC approximation of v, line 4 holds since $\forall i \in N$, $v_i^*(\pi(i)) \leq v_i(\pi(i))$ and line 5 holds since π is stable for $\langle N, v^* \rangle$. □

We now analyze specific classes of hedonic games for their PAC learnability and stabilizability.

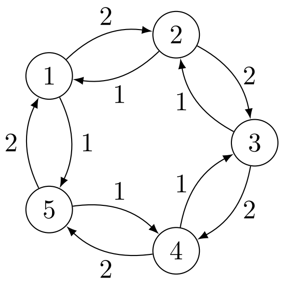


Figure 1: An ASHG. Unshown edges have unknown weights.

4.1 Additively Separable Hedonic Games

Additively separable hedonic games (ASHGs) are hedonic games where an agent's utility from a coalition is the sum of utilities she assigns to other members of that coalition. Formally, in an ASHG, each agent $i \in N$ assigns a value $v_i(j) \in \mathbb{R}$ to any other player j; we let $v_i(S) = \sum_{j \in S \setminus \{i\}} v_i(j)$. We make the natural assumption that ASHGs are represented by the utility functions used in their definition. We first observe that ASHGs are PAC learnable.

Proposition 4.2. *The class of additively separable hedonic games is PAC learnable.*

Proof. The representation functions of additively separable hedonic games are linear functions, well known to be PAC learnable. □

The existence of a stable coalition structure in an ASHG is not guaranteed [10], and as it turns out ASHGs are not PAC stabilizable; given some information about a game, it can extend to a game with non-empty core, or to a game with an empty core, such that some coalition from $\mathcal{D}$ will block any proposed partition.

Proposition 4.3. *The class of additively separable hedonic games is not PAC stabilizable.*

Proof. Consider an additively separable hedonic game with 5 players depicted on Figure 1 where the weight of the edge (i, j) is $v_i(j)$, and edges not shown have unknown weights. Consider a distribution $\mathcal{D}$ that assigns probability of 1/5 to each of the five sets $\{i, (i \bmod 5) + 1\}$ and 0 elsewhere. It is possible that the game has empty core, for example if all of the remaining values $v_i(j) = -4$ in which case for any partition π, there is a set in the support of $\mathcal{D}$ that blocks it, hence $\Pr_{S \sim \mathcal{D}}[S \text{ core blocks } \pi] \geq \frac{1}{5}$. However, if the unseen $v_i(j)$ are non-negative, the grand coalition is core stable, hence any algorithm potentially PAC stabilizing the game cannot output that the core is empty. Hence, the class is not PAC stabilizable. □

Fractional Hedonic Games [4] are different from ASHGs in that the coalitions' values are scaled by their size, i.e. $v_i(S) = \frac{\sum_{j \in S} v_i(j)}{|S|}$. An analysis similar to that of ASHGs shows that they share both their learnability and their non PAC stabilizability.

Proposition 4.4. *The class of Fractional Hedonic Games is PAC learnable, but not PAC stabilizable.*

4.2 $\mathcal{W}$-games

In hedonic games with $\mathcal{W}$-preferences ($\mathcal{W}$-hedonic games), each player i has a preference $\succeq_i'$ over other players and coalition's value is determined by the worst player in that coalition [13, 14]. Formally,

$$S \succeq_i T \iff \forall s \in S : \exists t \in T : s \succeq_i' t.$$

Proposition 4.5. *The class of all representation functions of $\mathcal{W}$-hedonic games is efficiently PAC learnable.*

Proof. Consider the learnability of v_i, the i-th representation function. If it is learnable, trivially all n of them are learnable. We will prove the pseudo-dimension of v_i is n, and hence that v_i is learnable. Consider any set of coalitions $\mathcal{S}$ of size $n + 1$. Suppose for contradiction that the hypothesis class can pseudo-shatter $\mathcal{S}$. Denote by $\mathcal{S}'$ the subset of $\mathcal{S}$ obtained by removing any sets that contain some player exclusively:

```
1: $\mathcal{S}' := \mathcal{S}$
2: while $\exists T \in \mathcal{S}' : T \setminus \bigcup_{U \in \mathcal{S}' \setminus \{T\}} U \neq \emptyset$ do
3:     $\mathcal{S}' = \mathcal{S}' \setminus T$
4: end while
5: return $\mathcal{S}'$
```

Line 3 results with the set $\bigcup_{U \in \mathcal{S}'} U$ shrinking by at least one player, hence is executed at most n times, therefore $\mathcal{S}'$ returned by the procedure is non-empty. Since $\mathcal{S}$ is pseudo-shattered, then its subset $\mathcal{S}'$ is pseudo-shattered. Relabel $\mathcal{S}' = \{S_1, ..., S_m\}$ s.t. the threshold t_1 associated with S_1 is the smallest. Hence, there exists a representation function v_i such that $v_i(S_1) < t_1$ and for $j = 2, 3, ..., m$, $v_i(S_j) > t_j$. Since v_i is a representation function for a $\mathcal{W}$-hedonic game and $v_i(S_1) < t_1$, there is a player $x \in S_1$, such that $\forall T, x \in T : v_i(T) < t_1$. Since the condition from line 2 of the procedure cannot hold for the returned $\mathcal{S}'$, $\exists S_j \in \mathcal{S}' : x \in S_j$, but $v_i(S_j) > t_j \geq t_1$, a contradiction.

Since all representation functions have pseudo-dimension bounded by n, they are learnable assuming we can compute a consistent hypothesis in polynomial time.

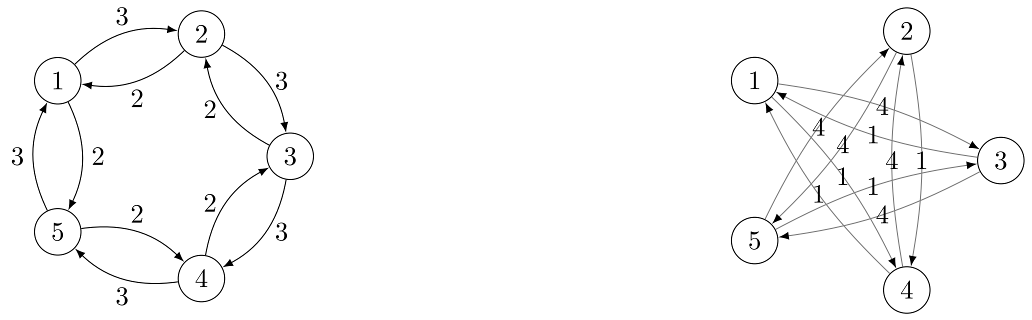
The following algorithm computes a hypothesis v_i^* consistent with a given sample $(\langle S_1, v(S_1)\rangle, ..., \langle S_m, v(S_m)\rangle)$. For every player $k \in N$, the score i assigns to k is $\max_{\{S_j : k \in S_j\}} v_i(S_j)$ or $-\infty$ if $\{S_j : k \in S_j\} = \emptyset$. The algorithm runs in time polynomial in n and m, hence the representation functions of $\mathcal{W}$-hedonic games are efficiently PAC learnable. □

Interestingly, the algorithm outputs an underestimate of v_i's, hence by Proposition 4.1, any partition in the core of the hypothesis is PAC stable. If we would only require some PAC hypothesis with an empty core in case we cannot give a PAC stable partition, this would be enough for stabilizability. However, we require that the core is empty *for certain.*

We prove $\mathcal{W}$-hedonic games are not PAC stabilizable, even for a particularly informative representation function: every player assigning any other player its Borda score (the number of players ranked lower), with no ties assumed. This representation function conveys a lot of information: $v_i(S)$ tells us the exact ranking of the worst agent in this coalition.

Proposition 4.6. *$\mathcal{W}$-hedonic games with the Borda score of the worst agent as the representation functions are not PAC stabilizable.*

Proof. Consider a $\mathcal{W}$-hedonic game with 5 players depicted in Figure 2a, where the weight of edge (i, j) is the Borda score for player j in i's preference. Consider a distribution $\mathcal{D}$ that returns one of the sets $\{i, (i \mod 5) + 1\}$ uniformly at random. Depending on the values of edges not shown, the game may extend to the game depicted on Figure 2b or



(a) A $\mathcal{W}$-game. Unshown edges have unknown weights.

(b) Possible extension of the game from Figure 2a.

Figure 2

a similar game that would be obtained if players were permuted according to $p(i) := (i \mod 5) + 1$. The respective sets of partitions resistant to coalitions supported by $\mathcal{D}$ are $A = \{\{\{1\}, \{2, 5\}, \{3, 4\}\}, \{\{1, 2, 5\}, \{3, 4\}\}\}$ and $B = \{\{\{2\}, \{3, 1\}, \{4, 5\}\}, \{\{2, 3, 1\}, \{4, 5\}\}\}$. Since $A \cap B = \emptyset$, even after observing all coalitions from $\mathcal{D}$, it is impossible to indicate a partition π for which $\Pr_{S \sim \mathcal{D}}[S \text{ core blocks } \pi] < \frac{1}{5}$ with non-zero confidence. Since in both cases the core of the game is not empty (respectively $\{\{1\}, \{2, 5\}, \{3, 4\}\}$ and $\{\{2\}, \{3, 1\}, \{4, 5\}\}$ are core stable), any algorithm potentially PAC stabilizing the class cannot output that the core is empty. Hence, the class is not PAC stabilizable. □

4.3 $\mathcal{B}$-games

$\mathcal{B}$-games [12, 14] are the counterpart of $\mathcal{W}$-games, where the value of a coalition depends on the favorite agent in that coalition. However, if two coalitions of different size share their most preferred agent, the smaller one is preferred. Formally, $S \succ_i T$ if and only if either: (a) for $s = \max_{\succ_i} S$ and $t = \max_{\succ_i} T$ we have $s \succ_i t$; or (b) for $s = \max_{\succ_i} S$ and $t = \max_{\succ_i} T$, $s = t$ and $|S| < |T|$. We assume strict agent preferences over individuals. Similarly to $\mathcal{W}$-games, $\mathcal{B}$-games are learnable.

Proposition 4.7. *$\mathcal{B}$-hedonic games with arbitrary representation functions are efficiently PAC learnable.*

Proof Sketch. $\mathcal{B}$-games differ from $\mathcal{W}$-games with respect to learning in that the values of coalitions with the same most preferred agent can differ if they are of a different size. However, if we consider only coalitions of the same size, then we can show that no set of size bigger than n can be pseudo-shattered (this is done in a manner similar to that of $\mathcal{W}$-games). Since n different coalition sizes are possible, the pseudo-dimension of $\mathcal{B}$-games is bounded by n^2.

To construct a hypothesis v_i^* consistent with $\mathcal{S} = \{S_1, \dots, S_m\}$, let us run the following procedure. Relabel S_k's s.t. $v_i(S_1) \geq \dots \geq v_i(S_m)$; given $j \in \{1, \dots, m\}$, we write $\mathcal{T}_j = \{S_1, \dots, S_j\}$. Set $\mathcal{T}$ to the highest indexed $\mathcal{T}_j$ s.t. (a) $\forall T_1, T_2 \in \mathcal{T} : v_i(T_1) > v_i(T_2) \iff |T_1| < |T_2|$; and (b) $X = \bigcap_{T \in \mathcal{T}} T \setminus \bigcup_{S \in \mathcal{S} \setminus \mathcal{T}} S \neq \emptyset$. We can now choose v_i^* for any (one) $x \in X$ in a way that ensures consistency with $v_i(T)$ for all $T \in \mathcal{T}$ (details omitted). The same procedure is then applied to $\mathcal{S} \setminus \mathcal{T}$, until a consistent labeling of the players is completed. Players that were not labeled by this procedure are given arbitrary small values. □

Representation functions for $\mathcal{B}$-games turn out to play an important role in their stabilizability; we begin by showing the following negative result.

Proposition 4.8. *The class of $\mathcal{B}$-games with arbitrary representation functions is not PAC stabilizable.*

Proof Sketch. Consider a $\mathcal{B}$-game with 7 players and a uniform distribution $\mathcal{D}$ of coalitions $S_1 = \{7, 1, 2\}, S_2 = \{2, 3\}, S_3 = \{3, 4\}, S_4 = \{4, 5\}, S_5 = \{5, 6, 7, 1\}$, where

$$\begin{array}{ccc} S_1 \succ_1 S_5, & S_1 \succ_7 S_5, & S_2 \succ_2 S_1, \\ S_3 \succ_3 S_2, & S_4 \succ_4 S_3, & S_5 \succ_5 S_4 \end{array}$$

Since $|S_5| > |S_1|$, it cannot be determined if $2 \succ_1 7$. The only coalition structure resistant to coalitions that $\mathcal{D}$ supports, that is common for all possible games if $2 \succ_1 7$ is assumed, is the grand coalition; however, if $7 \succ_1 2$, it is possible that S_5 blocks the grand coalition. The core is never empty for $\mathcal{B}$-games, hence the class cannot be PAC stabilized. □

As hinted in the proof sketch, the non-stabilizability stems from our inability to decide if two intersecting coalitions have the same most preferred agent. However, if we assume that the representation function allows us to decide whether a coalition S is preferred to T based on size or agent identity, then $\mathcal{B}$-games are PAC stabilizable; indeed, as we show in Section 4.4, we can PAC stabilize *Top Responsive games*, a superclass of $\mathcal{B}$-games, based on a similar assumption.

4.4 Top Responsive Games

In Top Responsive games [1], each agents appreciation of a coalition depends on the most preferred subset within the coalition. More formally, let the *choice sets* of agent i in coalition $S \in \mathcal{N}_i$ be defined by

$$Ch(i, S) := \{X \subseteq S : \forall Y \subseteq S, i \in Y : X \succeq_i Y\}.$$

If $|Ch(i, S)| = 1$, we denote the only element of $Ch(i, S)$ by $ch(i, S)$. A game satisfies *top-responsiveness* if:

1. For all $i \in N$ and $S \in \mathcal{N}_i$, $|Ch(i, S)| = 1$
2. For all $i \in N$ and $S, T \in \mathcal{N}_i$:
    (a) If $ch(i, S) \succ_i ch(i, T)$ then $S \succ_i T$
    (b) If $ch(i, S) = ch(i, T)$ and $S \subset T$, then $S \succ_i T$

Note that for $\mathcal{B}$-games, if $S \succ_i \{i\}$, then $|ch(i, S)| = 2$.

We assume that Top Responsive games are represented using an *informative* utility model; this means that for any $i \in N$ and $S, T \in \mathcal{N}_i$, it can be decided whether $ch(i, S) \succ_i ch(i, T)$, by observing the values $v_i(S)$ and $v_i(T)$.

This assumption is based on the intuition that coalitions are primarily judged by the worth of their choice set. If a representation function is informative and two coalitions have different choice sets, the function values will indicate the coalitions are in different 'buckets'. A simple example of an informative representation function, is one for which

$$\lfloor v_i(S) \rfloor = \lfloor v_i(T) \rfloor \iff ch(i, S) = ch(i, T).$$

Intuitively, the utility derived from the coalitions $ch(i, S)$ is given by integers, and any tiebreaking done by size constraints is done on lower order fractional terms.

We show that Top Responsive games are not PAC learnable, even if their representation functions are informative.

Proposition 4.9. *Top Responsive hedonic games with informative representation functions are not PAC learnable.*

Proof Sketch. Consider the class $\mathcal{R}$ of hedonic games where $\forall S, T \in \mathcal{N}_i : |S| > |T| \implies S \succ_i T$; $\mathcal{R}$ is a subclass of Top Responsive games, where the choice set of S is S itself for every S. The set $\mathcal{X} = \{S \in \mathcal{N}_i : |S| = \lceil \frac{n}{2} \rceil\}$ can be pseudo-shattered by $\mathcal{R}$. Since $|\mathcal{X}| > 2^{\frac{n-1}{2}}$, $P_{dim}(\mathcal{R}) > 2^{\frac{n-1}{2}}$; invoking Theorem 2.1 we conclude that $\mathcal{R}$ is not PAC learnable. □

Theorem 4.10. *Top Responsive hedonic games with informative representation are efficiently PAC stabilizable.*

Proof. We claim that Algorithm 1 PAC stabilizes Top Responsive games. Algorithm 1 is similar to the Top Covering Algorithm [1] used to find stable outcomes for Top Responsive hedonic games in the full-information setting. Intuitively, instead of using the exact choice sets, Algorithm 1 approximates them.

Algorithm 1 An algorithm finding a PAC stable outcome for Top Responsive games

```
Input: $\varepsilon$, $\delta$, set $\mathcal{S}$ of $m = (2n^4 + 2n^3)\lceil \frac{1}{\varepsilon} \log \frac{2n^3}{\delta} \rceil$ samples from $\mathcal{D}$
 1: $l \leftarrow 0$, $R_l \leftarrow N$, $\pi \leftarrow \emptyset$
 2: $\omega \leftarrow \lceil 2n^2 \frac{1}{\varepsilon} \log \frac{2n^3}{\delta} \rceil$
 3: while $R_l \neq \emptyset$ do
 4:     $\mathcal{S}' \leftarrow$ take and remove $\omega$ samples from $\mathcal{S}$
 5:     $\mathcal{S}' \leftarrow \{T : T \in \mathcal{S}', T \subseteq R_l\}$
 6:     for $i \in R_l$ do
 7:         if $i \notin \bigcup_{X \in \mathcal{S}'} X$ then
 8:             $B_{i,l} \leftarrow \{i\}$
 9:         else
10:             $B_{i,l} \in \arg\max_{T \in \mathcal{S}'} v_i(T)$
11:         end if
12:         $B_{i,l} \leftarrow \bigcap_{\{T \in \mathcal{S}' : ch(i,T) = ch(i,B_{i,l})\}} T.$
13:     end for
14:     for $j = 1, ..., |R_l|$ do
15:         $\mathcal{S}'' \leftarrow$ take and remove $\omega$ samples from $\mathcal{S}$
16:         for $i \in R_l$ do
17:             $B_{i,l} \leftarrow B_{i,l} \cap \bigcap_{T \in \mathcal{S}'' : ch(i,T) = ch(i,B_{i,l})} T.$
18:         end for
19:     end for
20:     $CC(i, R_l) \leftarrow \{i' \in R_l : \exists j_1, ..., j_k \in R_l : j_1 = i \wedge j_k = i' \wedge j_2 \in B_{j_1,l}, \ldots, j_k \in B_{j_{k-1},l}\}$
21:     $i^* \in \arg\min_{j \in R_l} |CC(j, R_l)|$
22:     $X_l \leftarrow CC(i^*, R_l)$
23:     $\pi \leftarrow \pi \cup X_l$
24:     $R_{l+1} \leftarrow R_l \setminus X_l$, $l \leftarrow l + 1$
25: end while
26: return $\pi$
```

Note that after step 19 the algorithm constructs a coalition of players based on (approximations of) choice sets, adds that coalition to π, removes it from the set R_l of players left to be considered, and if R_l is non-empty starts over, similarly to the Top Covering Algorithm. Note that R_l grows strictly smaller with every iteration, hence $l \leq n - 1$.

Define $\succeq_{i,R_l}$ as a modified preference of the player i, that equates any coalitions not entirely contained in R_l to the worst:

- If $T \subseteq R_l$ and $S \subseteq R_l$, $T \succeq_{i,R_l} S \iff T \succeq_i S$
- If $T \subseteq R_l$ but $S \nsubseteq R_l$, $T \succ_{i,R_l} S$
- If $T \nsubseteq R_l$ and $S \nsubseteq R_l$, $T \simeq_{i,R_l} S$

Given a distribution $\mathcal{D}$, we say that a coalition T is R_l-*top*-ε for player i, if

$$\Pr_{S\sim\mathcal{D}}[S \succ_{i,R_l} T] \leq \varepsilon.$$

Trivially, the probability of sampling an R_l-top-ε coalition for player i from $\mathcal{D}$ is at least ε. Note that if $\Pr_{S\sim\mathcal{D}}[S \subseteq R_l] \leq \varepsilon$, then any set is R_l-top-ε.

$B_{i,l}$ is the approximation of the choice set $ch(i, R_l)$. The algorithm's objective is to ensure that

$$\Pr_{S\sim\mathcal{D}}[v(S) \text{ is inconsistent with some } B_{i,l} = ch(i, R_l)] \leq \varepsilon.$$

Steps 4 to 12 approximate $ch(i, R_l)$ by $B_{i,l}$ by ensuring that sampling a coalition from $\mathcal{D}$ with a better choice set than $B_{i,l}$ is unlikely; this is done by examining enough coalitions so as to see some R_l-top-$\frac{\varepsilon}{2n^2}$ coalition for every player. Let us bound the probability that for a given i, none of the coalitions in $\mathcal{S}'$ are R_l-top-$\frac{\varepsilon}{2n^2}$:

$$\left(1 - \frac{\varepsilon}{2n^2}\right)^{\omega} = \left(1 - \frac{\varepsilon}{2n^2}\right)^{\lceil 2n^2 \frac{1}{\varepsilon} \log \frac{2n^3}{\delta} \rceil}$$

$$\leq \left(\left(1 - \frac{\varepsilon}{2n^2}\right)^{\frac{2n^2}{\varepsilon}}\right)^{\log \frac{2n^3}{\delta}} < \left(\frac{1}{e}\right)^{\log \frac{2n^3}{\delta}} < \frac{\delta}{2n^2}$$

Taking a union bound, the probability that there is some l and some player $i \in R_l$ s.t. there is no R_l-top-$\frac{\varepsilon}{2n^2}$ coalition for i in $\mathcal{S}'$ is at most $\frac{\delta}{2}$.

Note that seeing a coalition entirely contained in R_l may be very unlikely, and any coalitions not contained in R_l are removed from $\mathcal{S}'$ at step 5; hence, $\mathcal{S}'$ can end up not containing any coalition with some player i (condition in line 7). But then with high confidence, as above, every coalition is R_l-top-$\frac{\varepsilon}{2n^2}$, and the algorithm can pick for example $\{i\}$.

Note that steps 12 and 17 require an informative representation.

The intuition of steps 14 to 19 is to further approximate $ch(i, R_l)$ by $B_{i,l}$, ensuring that seeing a coalition from $\mathcal{D}$ with the same choice set as $B_{i,l}$, but not containing some player from $B_{i,l}$, is unlikely. Let $E(S, l, i)$ be the event $[S \subseteq R_l \wedge ch(i, B_{i,l}) = ch(i, S) \wedge B_{i,l} \setminus S \neq \emptyset]$. Suppose (*) $\Pr_{S\sim\mathcal{D}}[E(S, l, i)] \geq \frac{\varepsilon}{2n^2}$; then the probability that $\nexists S \in \mathcal{S}'' : E(S, l, i)$ is $\left(1 - \frac{\varepsilon}{2n^2}\right)^{\omega} < \frac{\delta}{2n^3}$. By the union bound, for every iteration of l and j, for all i s.t. (*) holds, $\exists S \in \mathcal{S}'' : E(S, l, i)$ with confidence $1 - \frac{\delta}{2}$.

Note that if $\exists S \in \mathcal{S}'' : E(S, l, i)$, then step 17 removes some agent from $B_{i,l}$, hence the loop at step 14 would be redundant with more iterations. Concluding, after the algorithm's execution, (*) does not hold for any l and i with confidence $1 - \frac{\delta}{2}$.

To examine if the algorithm achieves PAC stability, consider some coalition T that core blocks π. Let $k = \min\{l : \exists i \in T \cap X_l\}$ and $i \in T \cap X_k$. Since T core blocks π, $T \succ_i X_k$. By top-responsiveness, one of the following holds: (a) $ch(i, T) \succ_i ch(i, X_k)$ or (b) $X_k \setminus T \neq \emptyset$. Suppose that (a) holds; by construction of X_k, $ch(i, X_k) \succeq_i B_{i,k}$. By top-responsiveness,

$T \succ_i B_{i,k}$. Recall that with confidence $1 - \frac{\delta}{2}$ it can be assumed that $B_{i,k}$ is R_k-top-$\frac{\varepsilon}{2n^2}$. By the choice of k, $T \subseteq R_k$. By the union bound, $\Pr_{T \sim \mathcal{D}}[T \succ_i B_{i,k}] < \frac{\varepsilon}{2}$.

Next, suppose that (b) holds. By construction of X_k and $i \in T \cap X_k$, there is a player $j \in T \cap X_k$, s.t. $B_{j,k} \nsubseteq T$. Since T core blocks π, $T \succ_j X_k$. Then either $ch(j,T) \succ_j ch(j, B_{j,k})$ or $E(T,k,j)$ occurs. The former implies (a). Recall that with confidence $1 - \frac{\delta}{2}$, for all l and i, $\Pr_{T \sim \mathcal{D}}[E(T,l,i)] < \frac{\varepsilon}{2n^2}$. By union bound, the latter happens with probability smaller than $\frac{\varepsilon}{2}$. Taking a union bound, $\Pr_{T \sim \mathcal{D}}[T \text{ core blocks } \pi] < \varepsilon$ with confidence $1 - \delta$. □

5 Conclusions and Future Work

Our results seem to imply that classes of hedonic games with instances having an empty core are impossible to PAC stabilize. However, things are not as simple as that: some of our examples (Proposition 4.6 and 4.8) face an algorithm with a decision between two possible games with a non-empty core; however, partitions that stabilize one will destabilize the other.

One interesting direction for future work would be using *structural restrictions*: recent works study hedonic games where agent preferences follow a graph structure [19]. It would be useful to see whether certain graphical assumptions imply PAC stabilizability. Furthermore, while our work studies the core, one can focus on other hedonic solution concepts.

Beyond its theoretical interest, we believe that our approach sets a compelling research agenda. It is often necessary to assume uncertainty in real-world domains; indeed, by grounding stability on observed data rather than the entirety of agents' preferences, we offer a tailored domain-specific solution. Since our algorithms (and PAC learning in general) do not use any information regarding the distribution $\mathcal{D}$, they may output different results in different domain implementations. Moving forward in this direction, it is quite possible that making certain assumptions with respect to $\mathcal{D}$ will circumvent some of our negative stability results.

Our work formally relates observational data and stability in hedonic games; this relation opens the door to empirical evaluation of hedonic games, a further step towards their implementation in real-world systems.

Acknowledgements

The authors were supported by a Singapore MOE Grant no. R-252-000- 625-133, and by the Singapore NRF Fellowship no. R-252-000-750-733. The authors would like to thank the anonymous reviewers of IJCAI 2017 for their insightful comments.

References

[1] J. Alcalde and P. Revilla. Researching with whom? stability and manipulation. *Journal of Mathematical Economics*, 40(8):869–887, 2004.

[2] M. Anthony and P. Bartlett. *Neural Network Learning: Theoretical Foundations.* Cambridge University Press, 1999.

[3] H. Aziz and F. Brandl. Existence of stability in hedonic coalition formation games. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems*, AAMAS'12, pages 763–770, 2012.

[4] H. Aziz, F. Brandt, and P. Harrenstein. Fractional hedonic games. In *Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems*, AAMAS'14, pages 5–12, 2014.

[5] H. Aziz and R. Savani. Hedonic games. In F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A.D. Procaccia, editors, *Handbook of Computational Social Choice*, chapter 15. Cambridge University Press, 2016.

[6] M.F. Balcan, F. Constantin, S. Iwata, and L. Wang. Learning valuation functions. In *Proceedings of the 25th Annual Conference on Learning Theory*, COLT'12, pages 4.1–4.24, 2012.

[7] M.F. Balcan, A.D. Procaccia, and Y. Zick. Learning cooperative games. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, IJCAI'15, pages 475–481, 2015.

[8] M.F. Balcan, T. Sandholm, and E. Vitercik. Sample complexity of automated mechanism design. In *Advances In Neural Information Processing Systems*, NIPS'16, pages 2083–2091, 2016.

[9] M.F. Balcan, E. Vitercik, and C. White. Learning combinatorial functions from pairwise comparisons. In *Proceedings of the 29th Annual Conference on Learning Theory*, COLT'16, pages 1–35, 2016.

[10] S. Banerjee, H. Konishi, and T. Sönmez. Core in a simple coalition formation game. *Social Choice and Welfare*, 18(1):135–153, 2001.

[11] S. Brânzei and K. Larson. Social distance games. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, IJCAI'11, pages 91–96, 2011.

[12] K. Cechlárová and J. Hajduková. Computational complexity of stable partitions with B-preferences. *International Journal of Game Theory*, 31(3):353–364, 2003.

[13] K. Cechlárová and J. Hajduková. Stable partitions with W-preferences. *Discrete Applied Mathematics*, 138(3):333–347, 2004.

[14] K. Cechlárová and A. Romero-Medina. Stability in coalition formation games. *International Journal of Game Theory*, 29(4):487–494, 2001.

[15] G. Chalkiadakis and C. Boutilier. Bayesian reinforcement learning for coalition formation under uncertainty. In *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems*, AAMAS '04, pages 1090–1097, 2004.

[16] G. Chalkiadakis, E. Markakis, and C. Boutilier. Coalition formation under uncertainty: Bargaining equilibria and the bayesian core stability concept. In *Proceedings of the 6th International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '07, pages 412–419, 2007.

[17] V.G. Deineko and G.J. Woeginger. Two hardness results for core stability in hedonic coalition formation games. *Discrete Applied Mathematics*, 161(13):1837–1842, 2013.

[18] J. H. Drèze and J. Greenberg. Hedonic coalitions: Optimality and stability. *Econometrica*, 48(4):987–1003, May 1980.

[19] A. Igarashi and E. Elkind. Hedonic games with graph-restricted communication. In *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems*, AAMAS'16, pages 242–250, 2016.

[20] M.J. Kearns and U.V. Vazirani. *An introduction to computational learning theory.* MIT press, 1994.

[21] R.B. Myerson. Virtual utility and the core for games with incomplete information. *Journal of Economic Theory*, 136(1):260–285, 2007.

[22] D. Peters and E. Elkind. Simple causes of complexity in hedonic games. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, IJCAI'15, pages 617–623, 2015.

[23] A. Shashua. Introduction to machine learning: Class notes 67577. *arXiv preprint arXiv:0904.3664*, 2009.

[24] A. Sinha, D. Kar, and M. Tambe. Learning adversary behavior in security games: A PAC model perspective. In *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems*, AAMAS'16, pages 214–222, 2016.

[25] G.J. Woeginger. Core stability in hedonic coalition formation. In *39th International Conference on Current Trends in Theory and Practice of Computer Science*, SOFSEM'13, pages 33–50. Springer, 2013.

Jakub Sliwinski
ETH Zurich
Zurich, Switzerland
Email: `jakvbs@gmail.com`

Yair Zick
National University of Singapore
Singapore
Email: `dcsyaz@nus.edu.sg`