Technical University of Crete
School of Electrical and Computer Engineering

Hedonic Games in the Real World: Machine Learning and Theoretical Extensions

Ηδονικά Παίγνια στον Πραγματικό Κόσμο: Μηχανική Μάθηση και Θεωρητικές Επεκτάσεις

Athina Georgara

Chania, 2019

Hedonic Games in the Real World: Machine Learning and Theoretical Extensions

Ηδονικά Παίγνια στον Πραγματικό Κόσμο: Μηχανική Μάθηση και Θεωρητικές Επεκτάσεις

A thesis submitted in fulfilment of the requirements for the degree of Master of Science in

Electronic and Computer Engineering

awarded by the

School of Electrical and Computer Engineering

at **Technical University of Crete** *Chania, 2 August 2019*

Athina Georgara

COMMITTEE:

Supervisor : Georgios Chalkiadakis, Associate Professor

Committee Member : Michail Lagoudakis, Associate Professor

Committee Member : Euripides Petrakis, Professor

Technical University of Crete

School of Electrical and Computer Engineering

Abstract

Successful as it may be in mathematically modelling many real-life problems, *cooperative game theory* regularly adopts assumptions that cannot possibly stand in most specific everyday scenarios. Given this, in our work in this thesis we focus on *Hedonic Games*—a class of cooperative games in which players form coalitions based on their individual preferences regarding their potential partners— and tackle certain such assumptions in order to provide a model that fits better the real world. To this end, we make several contributions to Hedonic Games, approaching them from both a theoretical and practical point of view and extending them in various ways.

To begin, a prevalent assumption in the literature is that agents are interested solely on the composition of their own coalition. In our work we lift this assumption by allowing agents to develop preferences not only on coalitions, but also on coalition structures—i.e., partitions of the agents' space. Specifically, we motivate and put forward the formal definition of *hedonic games in partition function form (PFF-HGs)*, and extend well-studied hedonic games' classes to this setting.

Another usual assumption in the hedonic games literature is that of complete information. However, in the real world this is almost never the case. To tackle this, we examine the problem of uncertainty regarding *hidden preference relations* in hedonic games. Specifically, we assume that agents interact within an unknown hedonic game setting, observe a small number of game instances, and attempt to learn the hidden aspects of the game. Then, we employ several learning models, both supervised and unsupervised, to approximately extract the latent preference relations and detect desirable collaboration patterns. In particular, we provide a thorough evaluation of the use of *linear regression*, *regression with basis functions*, *feed forward neural networks*, and the online *Latent Dirichlet Allocation (LDA) algorithm* for approximately learning the unknown preferences across several classes of hedonic games.

Last but not least, we initiate the study of a novel class of cooperative games, the *Hedonic Utility Games (HUGs)*, that takes into consideration both hedonic and utility-related preferences. We formally define HUGs, and show how to extend and apply existing stability solution concepts to them. Then, we put forward a novel solution concept, *Individually Rational - Individually Stable (IRIS)*, which characterizes the stability of coalition structures in HUGs and was developed specifically for such settings. In addition, we propose a natural, "trichotomous" hedonic preferences model; study certain HUGs prop-

erties in that model; and exploit it to characterize the feasibility of HUGs coalitions, and to obtain a probability bound for pruning the coalitional space. Pruning can thus be exploited to reduce the computational load of deriving globally acceptable ("kernel-stable") payoff configurations for IRIS partitions.

Περίληψη

Παρά την ικανότητά της να μοντελοποιεί μαθηματικά σε ένα αφαιρετικό επίπεδο πραγματικά προβλήματα, η συνεργατική θεωρία παιγνίων συχνά υιοθετεί υποθέσεις που τελικά δεν ευσταθούν σε πραγματικά περιβάλλοντα. Με βάση τα παραπάνω, σε αυτήν την μεταπτυχιακή εργασία εστιάζουμε στα *Ηδονικά Παίγνια*–μια κλάση συνεργατικών παιγνίων στην οποία οι παίκτες σχηματίζουν συνασπισμούς με βάση προσωπικές προτιμήσεις που σχετίζονται με τους πιθανούς συμπαίκτες τους– και αναιρούμε κάποιες συνήθεις υποθέσεις, με στόχο να παρέχουμε ένα μοντέλο που περιγράφει ακριβέστερα τον πραγματικό κόσμο. Η εργασία μας προσέγγισε και επέκτεινε τα Ηδονικά Πάιγνια τόσο από θεωρητική όσο και από πρακτική σκοπιά, και κατέληξε σε ποικίλλες επιστημονικές συνεισφορές.

Κατ' αρχάς, μια επικρατούσα υπόθεση στην βιβλιογραφία είναι ότι οι πράκτορες ενδιαφέρονται αποκλειστικά και μόνο για την σύνθεση του δικού τους συνασπισμού. Στην παρούσα εργασία, αφαιρούμε αυτόν την υπόθεση με το να επιτρέπουμε στους πράκτορες να αναπτύσσουν προτιμήσεις όχι μόνον σχετικά με συνασπισμούς, αλλά και σχετικά με δομές συνασπισμών– δηλαδή, διαμερίσεις του χώρου των πρακτόρων. Συγκεκριμένα, εισάγουμε έναν τυπικό ορισμό για τα *ηδονικά παίγνια σε μορφή συνάρτησης διαμέρισης (PFF-HGs)*, και επεκτείνουμε γνωστές κλάσεις ηδονικών παιγνίων σε αυτή την μορφή.

Μια άλλη υπόθεση που γίνεται συνήθως στα ηδονικά παίγνια, είναι αυτή της πλήρους πληροφόρησης. Ωστόσο, στον πραγματικό κόσμο αυτό δεν ισχύει σχεδόν ποτέ. Για να αντιμετωπίσουμε το θέμα, εξετάζουμε το πρόβλημα της *αβεβαιότητας* όσον αφορά *κρυμμένες σχέσεις προτιμήσεων* σε ηδονικά παίγνια. Συγκεκριμένα, υποθέτουμε ότι οι πράκτορες αλληλεπιδρούν σε ένα άγνωστο περιβάλλον ηδονικού παιγνίου, παρατηρούν ένα μικρό αριθμό από στιγμιότυπα, και προσπαθούν να μάθουν τις μή εμφανείς πτυχές του παιχνιδιού. Για το σκοπό αυτό, εφαρμόζουμε διάφορα μοντέλα επιβλεπόμενης και μή (μηχανικής) μάθησης, για να εξάγουμε προσεγγιστικά τις κρυφές σχέσεις προτιμήσεων και να ανιχνεύσουμε επιθυμητά μοτίβα συνεργασιών. Πιο συγκεκριμένα, παρέχουμε μια ενδελεχή αξιολόγηση με την χρήση *γραμμικής παλινδρόμησης, παλινδρόμησης με συναρτήσεις βάσης, προωθητικών νευρωνικών δικτύων* αλλά και του αλγορίθμου πιθανοτικής θεματικής μοντελοποίησης*Latent Dirichlet Allocation (LDA)*, για την προσεγγιστική μάθηση των αγνώστων προτιμήσεων σε διάφορες κλάσεις ηδονικών παιγνίων.

Τέλος, προτείνουμε και εισάγουμε την μελέτη μιας νέας κλάσης συνεργατικών παιγνίων, των *Ηδονικών Παιγνίων Χρησιμότητας (HUGs)*, τα οποία λαμβάνουν υπ' όψιν τόσο "ηδονικές' προτιμήσεις (σχετικές με τη σύνθεση της ομάδας), όσο και προτιμήσεις σχετιζόμενες με χρησι-

μότητα. Δίνουμε τον επίσημο ορισμό των HUGs, και επεκτείνουμε υπάρχουσες λύσεις ευστάθειας σε αυτά τα παίγνια. Εν συνεχεία, εισάγουμε μια καινούρια λύση ευστάθειας, την οποία καλούμε *Μεμονωμένα Ορθολογικό - Μεμονωμένα Ευσταθές (IRIS)* σημείο ισορροπίας, που σχεδιάστηκε ειδικά για τα παίγνια HUGs, και η οποία χαρακτηρίζει ευσταθείς δομές συνασπισμών σε αυτά τα παίγνια. Επιπροσθέτως, προτείνουμε ένα φυσικό μοντέλο "τριχοτόμησης" του χώρου των ηδονικών προτιμήσεων, μελετάμε συγκεκριμένες ιδιότητες των HUGs σε αυτό το μοντέλο, και το εκμεταλλευόμαστε ώστε να χαρακτηρίσουμε την εφικτότητα σχηματισμού συνασπισμών στα HUGs, και να υπολογίσουμε ένα πιθανοτικό άνω όριο για το "κλάδεμα" του χώρου συνασπισμών. Ως εκ τούτου, το κλάδεμα μπορεί να χρησιμοποιηθεί για να μειώσουμε το υπολογιστικό φορτίο για τον υπολογισμό κοινώς αποδεκτών από τους παίκτες (τεχνικά, "ευσταθών στον πυρήνα kernel") πληρωμών σε IRIS δομές συνασπισμών.

Acknowledgements

First I would like to thank my supervisor, Professor Georgios Chalkiadakis, for all his support and inspiration throughout my time in Technical University of Crete. By encouraging and entrusting me to work with multifaceted topics in multi-agent systems, machine learning, and game theory, Prof. Chalkiadakis helped me to open up new horizons regarding my scientific interests. Moreover, I would like to thank him for encouraging me to pursue further studies at the Artificial Intelligence Research Institute (IIIA) in Barcelona.

Secondly, I would like to express my gratitude to Professor Michail Lagoudakis and Professor Euripidis Petrakis, who as members of my examination committee helped me to complete my studies. Moreover, I would like to thank my friends, Marina, Eleftheria C., and Eleftheria G., and my colleagues, Thalia, Dimitris, Dia and Antonis, for all their help and support.

Last but not least, I would like to thank my parents, Christos and Lina, and my sister, Victoria, without whose support and encouragement I wouldn't have managed to fulfil my studies.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

A key notion that governs modern societies is that of collaboration and how individuals team up in order to achieve common goals. People need to form teams, agree upon actions and decision, and ultimately execute the decided action plan in order to accomplish a specific task. The field of Game theory (GT) [Osborne and Rubinstein, 1994], and in particular cooperative game theory [Chalkiadakis et al., 2011], along with multi-agent systems (MAS) [Wooldridge, 2009] study settings where there is such need of collaboration; and therefore intend to formally describe mathematical models that can appropriately depict such situations. Over the past decades the fields of game theory and multi-agent systems have drawn attention to a plethora of scientific research areas and primarily in *computer sciences* and *economics*.

A cooperative game, is a well-defined, specific framework, within which there is a set of participating agents, a set of actions, and a set of rules. The goal of cooperative game theory is to study, analyse and at some point be able to predict the behaviour of agents; i.e., which agents will work together, and what action will take under the set of rules defined by the game. In GT and MAS an 'agent' is an individual entity that acts and make decisions autonomously. As such, an agent can be an individual person, a robot, a software program, an investor, a player etc. regarding the real-life problem each cooperative game intends to model.

Moreover, one of the main points of interest in game theory focuses on the stability matter, i.e., explore the conditions that establish a situation where no one is willing to diverge from. Specifically, a state of a model is described by the teams, also refer to as coalitions, which are formed, along with some reward earned, referring either to the whole team or to each individual. A *stable* state is one where no participant has incentive to leave a team, join another team, or claim better reward. With this in mind, researchers examine the conditions that lead to such a stable state, from both a theoretcial and a computational perspective. In game theory, a set of such conditions influence the strategies followed by the agents, as such they are called *solution concepts*. One step further, the natural question rises: "Apart from finding sets of necessary and desired conditions, how can a set of individuals converge to a stable state?"; therefore research turns to centralized mechanisms [Oh et al., 2015], decentralized and self-organized formatting protocols [Chalkiadakis and Boutilier, 2004, Mamakos and Chalkiadakis, 2017, Taywade et al., 2019], which lead to stable settings.

Another crucial matter that has also drawn much attention is that of uncertainty [Chalkiadakis and Boutilier, 2004],[Sliwinski and Zick, 2017], [Mamakos and Chalkiadakis, 2018]. Questions such as "What kind of information is revealed to the coalition formation mechanism?","How can this information be exploited?", and "How can we extract new and valuable information?", naturally arise when we are dealing with the 'unknown'. As a natural consequence a new path of research opens in which we attempt to extract hidden information about the participants, their competences, and the interactions with one another. Thus the coupling of machine learning (ML) [Bishop, 2006] with game theory and multi-agent systems was inevitable. Including uncertainty as a factor into the game theoretic models is an attempt of making these models even more realistic, and transform them into ones that can fit the real world even better.

1.1 Motivation

In this thesis, we turn our attention to the class of *Hedonic Games* [Aziz et al., 2016b, Aziz et al., 2016a, Chalkiadakis et al., 2011]. Hedonic games constitute a class of cooperative games that intuitively attempts to capture the interpersonal relations amongst the players and the social bonds of the formed coalitions. This class of games can model a plethora of real-life scenarios in which each individual highly cares and takes into consideration the identity of the rest of the participants.

The majority of hedonic games literature, so far, considers complete information over the game. However, in a more realistic framework that would not be a plausible assumption. In real-life settings that can be modelled as hedonic games, we face the problem of *uncertainty*, i.e. the agents have little or no information about the overall game. That is, we cannot expect each agent to know the exact preference relation over a vast space of coalitions, especially as the number of agents increases. As such, it is essential to investigate hedonic games under uncertainty, explore the several learning techniques and approach the problem of learning preferences under the framework of hedonic games.

Cooperative games in *partition function form* [Michalak et al., 2008, Michalak et al., 2009, Michalak et al., 2010, Skibski et al., 2015] try to capture the 'bigger picture' of a game. That is, the agents instead of focusing in their 'microcosm', i.e., their own coalition and set of actions, takes into consideration all the coalitions; and they adapt their decision making mechanisms to also consider such externalities. Therefore, it is natural to approach hedonic games under the point of view of partition function form.

Usually, during the process of coalition formation in cooperative games, we distinguish the following two primary motives: *(a)* earn as much as possible in terms of payoff, and *(b)* be as much satisfied as possible within a coalition. The latter, clearly can be modelled by hedonic games; while the former (and most common motivative aspect) can be modelled by Transferable Utility Games, etc. [Chalkiadakis et al., 2011]. However, in many real-world applications, such an absolute demarcation among motives does not exist. On the contrary, people value (maybe in different proportions each) *both* hedonic preferences and payoff shares, when they are to collaborate with others in order to carry out a task. Thus, in the general case, when people are to form coalitions, they take into consideration all motivating aspects.

1.2 Contributions

This thesis, motivated by the previously mentioned aspects, approaches hedonic games from both a theoretical and a practical point of view. In fact, we distinguish this work in two main parts:

- the theoretical one: providing theoretical extensions on the model of classic hedonic games that can fit the real world more accurately; and

- the practical one: exploring several learning techniques in order to ultimately discover preference relations.

In the theoretical part of the thesis, we have introduced several extensions of the classic hedonic games. Particularly, we put forward the formal definition of *Hedonic Games in Partition Function Form*, which can be seen as a generalization of hedonic games; and provided several natural settings which this extension can model. We described the extension of several well-known classes of hedonic games to their partition function form; and within this scope, we proposed as well a propositional language to represent Boolean Hedonic Games in Partition Function Form. Moreover, we introduce a novel hybrid class of cooperative games, the *Hedonic Utility Games (HUGs)*. This class constitutes the vehicle for combining the usual utility-related preferences studied in TU games, with private, hedonic ones. We provide an initial study of their computational aspects, focusing on stability-related considerations. We have also devised a novel theoretical solution concept, namely *Individually Rational - Individually Stable (IRIS)*, which is specific to HUGs. We studied its existence properties, and provided a randomized transitions scheme to reach potential IRIS outcomes. Last but not least, we propose a instantiation of HUGs model that allow us to characterize feasible coalitions, obtain a probability bound on feasible coalition, and therefore reduce the computational load of finding *kernel-stable* payoffs, which is a game theoretic stability-related solution concept.

In the practical part of the thesis, we have thoroughly studied several machine learning models, both supervised and unsupervised, on the problem of extracting hedonic preference relations. We conducted a systematic evaluation on each learning model, that confirms the effectiveness of our work. Specifically, we exploited *Linear Regression*, *Regression with Basis Functions*, *Feed Forward Neural Networks* [Bishop, 2006] and *Probabilistic Topic Modeling* [Blei, 2012] in order to extract valuable information regarding hedonic preferences in several and, essentially different, classes of hedonic games. Moreover, within the scope of the learning process, we developed and proposed two evaluating metrics that capture the qualitative proximity of the learnt preference relation compared with the actual preference relation. Furthermore, we proposed an interpretation method in order to convert a coalition-sample into a document-sample that can be eventually used in PTMs.

The work presented in this thesis gave rise to two publications: "Learning hedonci games via probabilistic topic modeling" [Georgara et al., 2019a], and "Extracting hidden preferences over partitions in hedonic cooperative games" [Georgara et al., 2019b], while a third one "Hedonic

Utility Games" has been submitted for publication to an algorithmic game theory conference, and is currently under review.

1.3 Thesis Structure

In Chapter 2 we provide the necessary background for this thesis. We present the required theoretical notions on cooperative games, hedonic games, game theoretic solution concepts, and machine learning focusing on the techniques used here. In Chapter 3 we put forward the theoretical extension of *Hedonic Games in Partition Function Form*. Chapter 4 focuses on learning hedonic preferences in cooperative games. There we detail how we exploited several machine learning methods to learn preferences in this setting, and present a systematic experimental evaluation of the various methods. In Chapter 5 we introduce the novel class of cooperative games *Hedonic Utility Games*, study the application of existing stability concepts into the such settings, and put forward a novel solution concept especially devised for Hedonic Utility Games.

Chapter 2

Theoretical Background and Related Work

In this section we will present the necessary theoretical background, and we will walk through some baseline notions for game theory and machine learning. Moreover we will present and discuss prior works relative to ours. In what follows, we firstly discuss two wide classes of game theory, and then we focus on the specific class of *Hedonic Games*. Next, we move on to the notion of *Solution Concept* by presented the most prevalent concepts. Last but not least we go through machine learning, and we discuss both the *supervised* and *unsupervised* learning theory and methods.

2.1 Cooperative Games

In *cooperative games* agents work together in order to achieve a common goal. The need for collaboration mainly derives from at least one of the following scenarios: *(a)* the desired goal expresses the wider public good of the community, *(b)* the individuals cannot accomplish the task on their own (due to lack of resources or competences), *(c)* it is more beneficial to each individual to cooperate with others. A cooperative game is defined as:

Definition 1. [Wooldridge, 2009] Cooperative Game *A cooperative or coalitional game is a pair $G = \langle N; v \rangle$, where $N = \{1, \cdots, n\}$ is a set of agents, and $v : 2^N \rightarrow \mathbb{R}$ is called the characteristic function of the game.*

The agents need to form *coalitions* that have to execute a decided action plan as a whole in order to reach the desired outcome. To be more specific, a coalition is a group of agents that work with each other, join their resources and competences, and tackle together some task. Formally, given a finite, non-empty set of agents $N = \{1, \cdots, n\}$, a *coalition* C is a subset of the original set of agents, i.e., $C \subseteq N$. A coalition consisting of only one agent, $C = \{i\}$, is called *singleton*; and the coalition consisting of all agents, $C = N$, is called *grand coalition*. A collection of coalitions that constitute a partition of N is called *coalition structure*, that is, if $CS = \{C_1, \cdots, C_k\}$ is a coalition structure then any two coalition in CS are disjoint $C_i \cap C_j = \emptyset \; \forall \; i, j$, and the union of all coalitions in CS compose the original set of agents: $\bigcup_{C_i \in CS} C_i \equiv N$. The collection of all coalitions that contain a particular agent $i \in N$ is denoted as $N_i = \{C \subseteq N | i \in C\}$. Given a specific coalition structure CS, we denote with $CS(i)$ the unique coalition in CS that contains agent i, i.e. $CS(i) \equiv C \in CS$ s.t. $i \in C$.

In the majority of cooperative games the 'worth' of the desired goal is expressed through *utility*. That is, the fulfilment of a task, and the extend of the success, is described by a numerical value. In literature cooperative games are divided into two main classes: the transferable utility (TU) games and the non-transferable utility (NTU) games [Chalkiadakis et al., 2011].

2.1.1 Transferable Utility Games

In transferable utility games we find the notion of 'transfer'. A coalition by performing a course of actions can reach an outcome, and this outcome results a utility to the whole coalition. This utility can therefore be distributed to the members of the coalition in the form of a payoff. Thus, in a TU game there can be a 'flow' (transfer) of utility from one agent to another, since the model itself does not define how the earned utility will be distributed at the end.

The utility achieve by each coalition is determined by a function v, usually referred to as *characteristic function* or *utility function*. The utility function relates each possible coalition C to a numeric value, and intuitively reflects the potential or the performance of the coalition on accomplishing a specific task.

Definition 2. [Chalkiadakis et al., 2011] TU-GAME *A Transferable Utility (TU) Game G is given by a pair $G = \langle N; v \rangle$, where $N = \{1, \cdots, n\}$ is a finite, non-empty set of agents, and $v : 2^N \to \mathbb{R}$ is a characteristic function, which maps each coalition $C \subseteq N$ to real number $v(C)$.*

The outcome of a TU game G is given by a pair $\langle CS; \text{x} \rangle$, where CS is a coalition structure, and $\text{x} \in \mathbb{R}^n$ is a payoff vector. The payoff vector contains the payoffs assigned to each agent, i.e. the portion of the coalitional utilities assigned to each coalitional member. Each coalitional utility is distributed to its members, i.e., for each $C \in CS$ we have that $\sum_{i \in C} x_i \leq v(C)$; this is a *feasibility* requirement of the payoff vector, each coalition can reward its members with part of the obtained utility. Each agent receives a non-negative payoff, i.e., $x_i \geq 0$ for any agent $i \in N$. If agent i's payoff x_i is greater or equal to the coalitional value of the singleton $\{i\}$, then x_i is a *individually rational* payoff. If all payoffs x_i are individually rational, then the payoff vector is an *imputation*.

2.1.2 Non-Transferable Utility Games

On the other hand, non-transferable utility games do not comprize any transfer of utility from one agent to another. In the settings modelled by NTU games, the utility obtained is agent-related; that is, when a task is performed by a coalition C, there is not a single dividable utility achieved by the team, but there is a utility achieved by each member of the coalition.

Definition 3. [Shoham and Leyton-Brown, 2008] NTU-Game *A Non-Transferable Utility (NTU) Game G is given by a pair* $G = \langle N; v \rangle$*, where* $N = \{1, \cdots, n\}$ *is a finite, non-empty set of agents, and* $v : 2^N \rightarrow 2^{\mathbb{R}^{|C|}}$ *is a function that associates each coalition* $C \subseteq N$ *a set of value vectors,* $v(C) \subseteq \mathbb{R}^{|C|}$*, which can be interpreted as the different sets of payoffs that C is able to achieve for each of its members.*

To clear any vagueness about the settings that can be modelled by NTU games, let us use the following example (from [Chalkiadakis et al., 2011]):

Example 1. *Consider a senior tenured professor A at university X who cooperates with a junior, non-tenured assistant professor B at university Y. Both will obtain some benefit from writing the paper, but the benefit that B obtains may well be much greater than the benefit that A obtains, simply because the value added to B's career is greater than the value added to A's, and the benefits that B obtains (enhanced reputation, scientific credibility, standing in the field) cannot easily be transferred from B to A.*

Looking at the example we can easily infer that the scientific credibility B receives results from both the paper and the collaboration with professor A, while A's scientific credibility derives solely from the paper; apparently, junior professor B cannot transfer part of his/her received credibility to A.

In [Chalkiadakis et al., 2011] the authors provide a slightly different definition for NTU games, pointing out that the class of NTU games can be thought of as a generalization of TU games. According to this definition in an NTU game each coalition has a set of choices describe by a function $v : 2^N \rightarrow 2^\Lambda$, and each agent express a complete, reflexive, and transitive preference relation, $\succsim$, over these choices. That is, let C be a coalition, then $v(C) = \{\lambda_1^C, \lambda_2^C, \cdots\}$ are the choices coalition C has; for each $i \in C$ the preference relation $\succsim_i$ defines a preference ordering on the choices $\{\lambda_1^C, \lambda_2^C, \cdots\}$. The outcome of an NTU game is a pair $\langle CS; \mathrm{c}\rangle$, where $CS = \{C_1, \cdots, C_k\}$ is a coalition structure, and $\mathrm{c} = \{\lambda_1, \cdots, \lambda_k\}$ a choice vector with $\lambda_1 \in v(C_1), \lambda_2 \in v(C_2), \cdots, \lambda_k \in v(C_k)$. Thus, with this approach it is not hard to see that if we map the set of choices of C described by $v(C)$ to the (infinite) set of all feasible payoff vectors, we can describe a TU game as its NTU counterpart. Therefore, NTU can be seen as a generalization of TU games.

2.1.3 Hedonic Games

A special class of cooperative games, and the central point of interest in this thesis, is that of *Hedonic Games* (HGs). The main difference among hedonic games and the other classes of cooperative games is that here there is no notion of utility at all. In contrary, hedonic games intend to capture the essence of personal satisfaction via collaboration; that is, hedonic games model settings where we focus on "is individual i enjoying the company of his/her partners in coalition C".

Definition 4. [Aziz et al., 2016b] Hedonic Game *A Hedonic Game G is given by a pair* $\langle N, \succsim\rangle$, *where* $N = \{1, \cdots, n\}$ *is a finite, non-empty set of agents, and* $\succsim = \{\succsim_1, \cdots, \succsim_n\}$ *is a* preference profile *that specifies for each agent* $i \in N$ *a complete, reflexive, and transitive* preference relation $\succsim_i$ *on* N_i. *The outcome of G is a coalition structure CS.*

The main component of a hedonic game is the *preference profile* of the agents within the game. That is, given a set of agents, a preference profile accumulates the personal preferences of all agents. Each agent expresses her preferences over all the possible coalitions she can be member of. An agent i's preferences are encoded with a complete, reflexive, and transitive preference relation $\succsim_i \subseteq N_i \times N_i$ as follows: for any two coalitions $S, T \in N_i = \{C \subseteq N | i \in C\}$ there is a relation of the form $S \succsim_i T$ or $T \succsim_i S$, where $S \succsim_i T$ is interpreted as "i prefers coalition S at least as much as coalition T". We say that i *strictly* prefers coalition S over T if and only if it holds $S \succsim_i T$ but not $T \succsim_i S$, and we denote this relation as $S \succ_i T$. Agent i is *indifferent* between coalitions S and T if and only if hold both $S \succsim_i T$ and $T \succsim_i S$, and we denote this relation as $S \sim_i T$.

Usually hedonic games are considered as a subclass of NTU games with the following compromises: the set of choices Λ contains all the possible coalitions of N, i.e. $\Lambda = \{C | C \subseteq N \setminus \emptyset\}$; the choices available of a coalition C consist of a single choice: the coalition itself, i.e., $v(C) = \{C\}$ and $|v(C)| = 1$. Thus, the outcome of a hedonic game, described as an NTU game, is a coalition structure CS and the unique choice vector available to this particular CS, which is identical to the coalition structure and therefore redundant.

Next we present some well-known classes of hedonic games and their properties. These classes of hedonic games will be subject to our study within the scope of this thesis.

Additively Separable Hedonic Games

A hedonic game is said to have additively separable preferences or be an *Additively Separable Hedonic Game* (ASHG) [Chalkiadakis et al., 2011] if each agent i's preference relation is formed according to some function v_i of the form $v_i(C) = \sum_{j \in C} M_{i,j}$. Specifically, there exists an $|N| \times |N|$ real value matrix M ($M \in \mathbb{R}^{|N| \times |N|}$) such that each agent i's preferences obey the following rule: $S \succsim_i T$ if and only if $v_i(S) \geq v_i(T) \Leftrightarrow \sum_{j \in S} M_{i,j} \geq \sum_{k \in T} M_{i,k}$, where $M_{i,j}$ is the real value agent i assigns to agent j. Intuitively, each agent i shows her preferences over the other agents through the values $M_{i,j}$ $\forall j \in N$, and these preferences over agents are then 'lifted' to preferences over coalitions through accumulative function v_i which sums up all the preferences over individuals participating in each coalition.

$\mathcal{B}/\mathcal{W}-$ and Fractional Hedonic Games

Similarly to the class of ASHGs there are other classes that use preferences over agents which are later lifted to preferences over coalitions via some function v_i.

The classes of $\mathcal{B}/\mathcal{W}-$Hedonic Games ($\mathcal{B}/\mathcal{W}-$HGs) [Chalkiadakis et al., 2011] form an agent's preferences based on the *best* ($\mathcal{B}$) and the *worst* ($\mathcal{W}$), respectively, agent within each coalition. In particular, in $\mathcal{B}-$HGs agent i prefers coalition $S \in N_i$ over coalition $T \in N_i$ if and only if i prefers the most preferred agent $s \in S$ over the most preferred agent $t \in T$, i.e. $S \succsim_i T$ if and only if $s \succsim_i t$. Respectively, in $\mathcal{W}-$HGs agent i prefers coalition $S \in N_i$ over coalition $T \in N_i$ if and only if i prefers the least preferred agent $s \in S$ over the least preferred agent $t \in T$, i.e. $S \succsim_i T$ if and only if $s \succsim_i t$. It is not hard to see, that if we have the real value matrix $\mathcal{M}$, the function v_i in the case of the $\mathcal{B}-$HGs is of the form $v_i(C) = \max_{j \in C}\{\mathcal{M}_{i,j}\}$ and the preferences over coalitions follows the rule: $S \succsim_i T$ if and only if $v_i(S) \geq v_i(T) \Leftrightarrow \max_{j \in S}\{\mathcal{M}_{i,j}\} \geq \max_{k \in T}\{\mathcal{M}_{i,k}\}$; while in the case $\mathcal{W} - HGs$ the function v_i is of the form $v_i(C) = \min_{j \in C}\{\mathcal{M}_{i,j}\}$ and the preferences over coalitions follows the rule: $S \succsim_i T$ if and only if $v_i(S) \geq v_i(T) \Leftrightarrow \min_{j \in S}\{\mathcal{M}_{i,j}\} \geq \min_{k \in T}\{\mathcal{M}_{i,k}\}$.

The class of *Fractional Hedonic Games* (FHGs) was introduced in [Aziz et al., 2014] as a natural consequence of ASHGs and $\mathcal{B}/\mathcal{W}-$HGs. In FHGs the defining function v_i that produces the preference relation over coalitions for agent i is the average operator. That is, here we have that $v_i(C) = \text{avg}_{j \in C}\{\mathcal{M}_{i,j}\} = \frac{\sum_{j \in C} \mathcal{M}_{i,j}}{|C|}$, where $\mathcal{M}_{i,j}$ is the real value i assigns to j, and $|C|$ denotes to the size of the coalition. Therefore, agent i prefers coalition $S \in N_i$ over $T \in N_i$ if and only if $v_i(S) \geq v_i(T) \Leftrightarrow \text{avg}_{j \in S}\{\mathcal{M}_{i,j}\} \geq \text{avg}_{k \in T}\{\mathcal{M}_{i,k}\}$.

Boolean Hedonic Games

Boolean Hedonic Games (BHGs) focus on hedonic games with *dichotomous* preferences. [Aziz et al., 2016a] initiated the study of such games and provided a succinct representation for BHGs using propositional formulas. In hedonic games with dichotomous preferences each agent i classifies all the coalitions she is a member of into two disjoint sets. Similar frameworks to this have been studied in economics and social sciences regarding individuals distinguishing available options or outcomes into good ones and bad ones (for example in [Bogomolnaia et al., 2005]). Formally, each agent i defines two disjoint sets, the first set contains the coalitions that i prefers and is denoted as N_i^+, while the second one contains the coalitions that i does not prefer and is denoted as N_i^-. Each coalition $C \in N_i$ must exist to either N_i^+ or N_i^-, and it must exist to exactly one of these sets, i.e., $N_i^+ \cup N_i^- = N_i$ and $N_i^+ \cap N_i^- = \emptyset$. Having defined N_i^+ and N_i^-, the preference relation is formed as follows: agent i strictly prefers all coalitions in N_i^+ over any coalition in N_i^-, while i is indifferent about coalitions existing in the same set. That is, for two coalitions $S, T \in N_i$ it holds that:

- $S \succ_i T$ if and only if $S \in N_i^+$ and $T \in N_i^-$;
- $S \sim_i T$ if and only if $S, T \in N_i^+$ or $S, T \in N_i^-$.

The propositional formula ϕ_i proposed in [Aziz et al., 2016a] consists a descriptive declaration of the set N_i^+. That is, formula ϕ_i encodes the collaboration patterns that agent i desires: the formula takes the form $\phi_i = \phi_{i,1} \vee \phi_{i,2} \vee \cdots \vee \phi_{i,k}$, where $\phi_{i,j}$ describes a desired collaboration pattern. In each $\phi_{i,j}$ we find the agents i would like to be in a coalition, and the agents she wouldn't. In its generality the formula $\phi_{i,j}$ can take the form:

$$\phi_i = (p_{1,i} \vee \neg p_{1,i}) \wedge (p_{2,i} \vee \neg p_{2,i}) \wedge \cdots \wedge (p_{i-1,i} \vee \neg p_{i-1,i}) \wedge (p_{i,i+1} \vee \neg p_{i,i+1}) \wedge \cdots \wedge (p_{i,n} \vee \neg p_{i,n})$$

where $p_{i,j}$ is a propositional variable that indicates if agents i and j are in the same coalition. For notational simplicity in [Aziz et al., 2016a] the authors use the indicative literal ij instead of the variable $p_{i,j}$, while in [Peters, 2016] the author eliminates the indicator i from ij as it is implied that $\phi_{i,j}$ refers to agent i. Therefore, the formula $\phi_{i,j}$ can now be written as: [1]

$$\phi_{i,j} = \bigwedge_{k \in \text{Appealing Partners}_j} k \bigwedge_{l \in \text{Repellent Partners}_j} \bar{l}$$

Any coalition $C \in N_i$ that includes all agents in the "Appealing Partners$_j$" set and does not include any agent in the "Repellent Partners$_j$" set satisfies the collaboration pattern $\phi_{i,j}$, satisfies the formula ϕ_i, and therefore coalition C is positioned into N_i^+ set. Respectively, any coalition that cannot satisfy any collaboration pattern, does not satisfy formula ϕ_i, and is therefore position into N_i^- set.

[1] The symbols $\bar{l}$ and $\neg l$ are equivalent; here we use the notation $\bar{l}$, since it was the one used in the original paper.

2.2 Solution Concepts

In game theory a solution concept is a formal rule for predicting how a game will eventually be played. These predictions, i.e., the "solutions" of the game, describe which strategies would (or should) be adopted by *rational* players and, therefore, define the result, i.e., the final outcome, of the game. "A rational agent is one that acts so as to achieve the best outcome or, when there is uncertainty, the best expected outcome" [Russell and Norvig, 2009]. Depending on the characteristics and the scalability (number of agents) of a game, the number of possible outcomes may become extremely large–and in the case of TU games where there is a dividable coalitional utility to be shared amongst the coalitional members, the number of possible outcomes can be uncountable. It is not hard to see that not all outcomes are equally desired by all agents, nor all outcome are equally to occur [Chalkiadakis et al., 2011]. For this reason, the outcomes are evaluated according to the following two properties: *(i) fairness*, and *(ii) stability*. In this work, we focus on the matter of stability, and in following subsections we discuss some well-established and well-studied classes of solution concepts regarding stability.

2.2.1 Individual Rationality

As previously mentioned, in game theory and artificial intelligence we intend to predict how rational agents will behave during a game. The simplest notion of rationality guarantees that no agent will be harmed. To clarify this, *individual rationality* ensures that the outcome of the game will be for all agents at least as good as what they could achieve on their own. In other words, if an outcome is individually rational then for each agent this outcome is at least as desirable as an outcome where the agent would be singleton. Formally we say that:

Definition 5. [Chalkiadakis et al., 2011] NTU/TU-IR *Give a NTU/TU game* $G = \langle N; v \rangle$, *an outcome* $\langle CS; \mathrm{x} \rangle$ *is called individually rational (IR) if for every agent* $i \in N$ *it holds that* $x_i \geq v(\{i\})$.

Definition 6. [Aziz and Brandl, 2012] HG-IR *Given a hedonic game* $G = \langle N, \succsim \rangle$, *an outcome* CS *is called individually rational (IR) if no agent has an incentive to become alone, i.e., for all* $i \in N$ *it holds that* $CS(i) \succsim_i \{i\}$.

Individual rationality is a solution concept that can be applied in all classes of cooperative games (TU, NTU, and hedonic games), and it is actually considered to be a minimum required property in terms of stability [Aziz et al., 2016b].

2.2.2 Nash Stability

Nash equilibrium is a foundational solution concept mostly used in non-cooperative settings. In non-cooperative game theory a Nash equilibrium is described by an action profile a^* which satisfies the following property: each player i cannot benefit by altering his/her action a_i^* to another a_i', while all other player $j \neq i$ stick to their action a_j^* [Osborne, 2004]. That is for each agent i the action a_i^* is the best response to action profile $a^* = \{a_1^*, \cdots, a_{i-1}^*, a_{i+1}^*, \cdots, a_n^*\}$ [Shoham and Leyton-Brown, 2008].

To best to our knowledge, there is no formal definition regarding Nash Equilibrium regarding TU games, or NTU games in general. In hedonic games' literature however, we find the concept of *Nash Stability*, where an outcome is Nash stable if no agent can benefit, i.e. be in a more desirable coalition than its current one, by unilaterally deviate into another existing coalition with the understanding that all other agents $j \neq i$ remain to their current coalitions [Dréze and Greenberg, 1980a, Bogomolnaia and Jackson, 2002]. Formally:

Definition 7. [Bogomolnaia and Jackson, 2002] Nash Stability *Given a hedonic game* $G = \langle N, \succsim \rangle$*, an outcome CS is called Nash stable (NS) if for all agents* $i \in N$ *and all coalitions* $S_k \in CS \cup \{\emptyset\}$ *it holds that* $CS(i) \succsim_i S_k \cup \{i\}$.

2.2.3 Individual Stability

A very attractive solution concept that we find in the hedonic games' literature is that of *individual stability*. According to this stability concept, the agents are allowed to perform unilateral deviation to existing coalitions or an empty one. In a individually stable coalition structure, no agent prefers to unilaterally deviate into a new coalition and, at the same time, is welcomed by this new coalition. Formally:

Definition 8. IS-deviate *In a hedonic game* $G = \langle N, \succsim \rangle$*, given a coalition structure CS, an agent* $i \in N$ *can IS−deviate into coalition* $S \in CS \cup \{\emptyset\}$ *if for agent i it holds that* $S \cup \{i\} \succ_i CS(i)$*, and for each agent* $j \in S$ *it holds that* $S \cup \{i\} \succsim_j S$.

Definition 9. Individual Stability *Given a hedonic game $G = \langle N, \succsim \rangle$, an outcome CS is called individually stable (IS) if there is no agent that can IS-deviate into any coalition in CS, i.e., $\nexists\, i \in N, S \in CS$ such that i can IS-deviate into S.*

"The individual stability can be seen as a myopic property of Nash Stability which does not consider simultaneous moves by several individuals" [Dréze and Greenberg, 1980a].

2.2.4 Core Stability

The most powerful solution concept in cooperative game theory is that of the *Core*. Researcher working in the field of any class of cooperative games have been studying the core solution concept, its properties, and how a core-stable outcome can be reached. In TU games, the core consists of all the pairs $\langle CS; \mathrm{x} \rangle$ where the payoff vector x is such that for any possible coalition $C \subseteq N$ the summation of the agents' payoffs are at least as good as the utility of the coalition. In words, if an outcome is core-stable, it ensures that there exists no coalition that can provide strictly better payoff to some player. Formally:

Definition 10. [Chalkiadakis et al., 2011] TU-Core *The Core $C(G)$ of a TU game $G = G = \langle N; v \rangle$ is the set of all outcomes $\langle CS; \mathrm{x} \rangle$ such that $x(C) = \sum_{i \in C} x_i \geq v(C)$ for every $C \subseteq N$.*

In the NTU settings we come upon the notion of *objection*. An objection to some outcome expresses that for some coalition $C \subseteq N$ there is a choice that is strictly better (more preferable) for all members in C than the current choice in the outcome. Formally:

Definition 11. [Chalkiadakis et al., 2011] NTU-Core *Let $G = G = \langle N; v \rangle$ be an NTU game, and let $\langle CS; \mathrm{c} \rangle$ be an outcome of G. Then we say there is an objection to* c *if exists a choice λ and a coalition $C \subseteq N$ such that $\lambda \in v(C)$ and $\lambda \succ_i c_i$ for all $i \in C$. The Core $C(G)$ of G is the set of all outcomes that omits is no objection.*

Similarly to the NTU games, in hedonic games we consider *blocking* instead of objections. That is, given an HG we say that a group of agents *blocks* a coalition structure if all agents in the group *strictly* prefer this group to their current coalition in the coalition structure. Therefore, a coalition structure is core-stable if it cannot be blocked by any group of agents. Formally:

Definition 12. [Aziz and Brandl, 2012] HG-Core *Given a hedonic game* $G = \langle N, \succsim \rangle$*, a coalition* $S \subseteq N$ *blocks a coalition structure CS, if each player* $i \in S$ *strictly prefers S to her current coalition* $CS(i)$ *in coalition structure CS. A coalition structure which admits not blocking coalition is said to be in the core* $C(G)$*.*

2.2.5 Kernel Stability

The *kernel* solution concept was introduced by [Davis and Maschler, 1965], and it refers solely to TU settings, as it is highly related to the essence of transfer. In particular, in a kernel-stable outcome no player can claim any portion of payoff of another player. That is, given an outcome $\langle CS; \mathrm{x} \rangle$ if each pair of agents within the same coalition is in *bilateral equilibrium*, then the outcome is kernel stable.

A key notion in the kernel solution concept is that of *excess*; given a game $G = \langle N; v \rangle$, the excess of a coalition C with respect to a payoff vector x is given by $e(C) \triangleq v(C) - \sum_{i \in C} x_i$. We define the *maximum surplus* of agent i over agent j *wrt.* an outcome $\langle CS; \mathrm{x} \rangle$ as $s_{i,j} \triangleq \max_{C \in I_{i,j}} \{e(C)\}$, where $I_{i,j} \triangleq \{C \subset N | i \in C \wedge j \notin C\} \equiv N_i \setminus N_j$ is the collection of all coalitions that contains agent i but not agent j. Agent i *outweighs* agent j *wrt.* $\langle CS; \mathrm{x} \rangle$ if $s_{i,j} > s_{j,i}$ and $x_j > v(\{j\})$. Thus, given a game $G = \langle N; v \rangle$ and an outcome $\langle CS; \mathrm{x} \rangle$, two agents $i, j \in N$ are in bilateral equilibrium if-f neither agent i outweighs agent j, nor agent j outweighs agent i. A coalition C is said to be *balanced wrt.* $\langle CS; \mathrm{x} \rangle$ if-f every pair of agents $i, j \in C$ is in bilateral equilibrium.

Definition 13. [Davis and Maschler, 1965] Kernel-Stable *The* kernel K *of a TU game* $G = \langle N; v \rangle$ *is the set of all the outcomes having balanced coalitions with individual rational payoff vector. Or equivalently,* $\langle CS; \mathrm{x} \rangle \in K$ *if and only if*

- *for each player* $i \in N$ $x_i \geq v(\{i\})$ *; and*
- *each two players are in bilateral equilibrium* wrt. $\langle CS; \mathrm{x} \rangle$.

An important property of the kernel solution concept is that is always *non-empty*, i.e., for any arbitrary game there is always an outcome $\langle CS; \mathrm{x} \rangle$ that is kernel-stable [Schmeidler, 1969].

2.3 Machine Learning

One of the primary goals of artificial intelligences and multi-agent systems is to build anthropomorphic agents (software, robots, machines, etc.) which can act and take decisions autonomously. People, however, besides a "pre-installed", inherent capacity for rational thinking and logical deduction, they also possess the ability to learn. Therefore, in order to give agents the ability to learn, a multitude of techniques have been developed that ultimately do exactly that: they allow agents to acquire knowledge in a way similar to human beings.

In the attempt of gaining this valuable skill of learning, researchers have developed concrete mathematical models that allow agents to systematically absorb and process the observed information. The machine learning process, can be classified into two main classes of models: the *supervised* models, and the *unsupervised* ones. These two classes are distinguished in the fact that a supervised model needs to "observe" a number of samples along with their corresponding actual outcome before the agent will be able to correctly predict the outcome of a previously unobserved input. In the contrary, a unsupervised model processes the observations without having any knowledge of the corresponding outcome, and discover links and interdependencies among the observations in its own. Intuitively, one could say that in supervise learning we know what we are looking for, and we try to find the mathematical pattern, i.e., a mathematically formal model, that encodes the information within each observation; while, on the other hand, in the unsupervised learning we "blindly" investigate observations and ultimately reach some conclusions for which we may have no a priori knowledge of their existence.

In what follows in this section we will present the base-line background for the machine learning techniques we work with in the current thesis.

2.3.1 Supervised Learning

In supervised learning [Bishop, 2006] the model is trained, and tested, with a number of sample observations along with the corresponding target value. An observation contains all the stimuli the model has access to, while the target value represents a desirable feature that we consider as output. Formally:

Definition 14. *Let an* observation *be a vector* $\mathrm{x}_k \in \mathbb{R}^n$*, where each dimension* $x_{k,i}$ *represents the*

i^{th} *stimuli. The corresponding* target value $t \in \mathbb{R}$ *represents the single-dimension outcome of the particular observation* x_k.

The input data are pairs of ⟨observation, target value⟩ ($\langle \mathrm{x}_k, t_k \rangle$). The output of a supervised learning model (SLM) is a function of the form $y = \mathrm{SLM}(\mathrm{x_k})$, where $\mathrm{x}_k \in \mathbb{R}^n$ is an observation, and $y \in \mathbb{R}$ is a prediction of the target value that corresponds to observation x_k.

Intuitively, supervised learning "targets" a specific feature, which is determined via the target values provided, and attempts to find a predictor which given an observation, i.e., a set of stimuli, will come with an legitimate outcome, i.e., an accurate value of the desirable feature.

Linear Regression

The *Linear Regression Model (LRM)* [Bishop, 2006] is a simple but yet powerful data analysis tool. As indicated by its name, an LRM provides us with a line that can describe the input data. In fact, an LRM attempts to find the line that fits the input data most accurately. The linear regression considers the input data to be linear, and treat them as such; that is, the output function is of the form:

$$y(\mathrm{x}_k) = w_0 + \sum_{i=1}^{n} w_i \cdot x_{k,i}$$

where $\mathrm{x} = \begin{bmatrix} x_1, x_2, \cdots, x_n \end{bmatrix}$ is an observation, the coefficients $\mathrm{w} = \begin{bmatrix} w_1, w_2, \cdots, w_n \end{bmatrix} \in \mathbb{R}^n$ are the *weights*, and $w_0 \in \mathbb{R}$ is the *bias factor*. The weight vector $\mathrm{w} \in \mathbb{R}^n$ assigns to each dimension of the observation vector (to each stimuli) a weight that shows the contribution of this dimension to the determination of the desired feature; while the bias factor (or bias parameter) allows for any fixed offset in the input data.

During the training phase of the learning procedure, an LRM uses K pairs of $\langle \mathrm{x}_k, t_k \rangle$. Thus, we have a vector that contains all the observations $\mathrm{x} = \begin{bmatrix} \mathrm{x}_1 \\ \mathrm{x}_2 \\ \vdots \\ \mathrm{x}_K \end{bmatrix} = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{K,1} & x_{K,2} & \cdots & x_{K,n} \end{bmatrix} \in \mathbb{R}^{K \times n}$; where each row corresponds to one of the K observations. Similarly, we also have a vector that contains all the target values $t = \begin{bmatrix} t_1 \\ t_2 \\ \vdots \\ t_K \end{bmatrix} \in \mathbb{R}^K$, where the k^{th} element is the target value of the k^{th} observation. According to these sample pairs, the LRM builds the weight vector and specifies the bias factor. Specifically, the weights are computed through the following relation: $w = (x^T \cdot x)^{-1} \cdot x^T \cdot t \in \mathbb{R}^{1 \times n+1}$. The bias factor can easily be included in the above computation by adding a feature in each observation; that is, if $x_k = \begin{bmatrix} x_{k,1}, & x_{k,2}, & \cdots, & x_{k,n} \end{bmatrix}$ is the k^{th} observation we add one more dimension with unitary value: $x_k = \begin{bmatrix} x_{k,1}, & x_{k,2}, & \cdots, & x_{k,n} \end{bmatrix} \in \mathbb{R}^{n+1}$. Therefore, the weight vector $w^T = \begin{bmatrix} w_0, & w_1, & w_2, & \cdots, & w_n \end{bmatrix} \in \mathbb{R}^{n+1}$ includes the bias factor w_0 and all weights w_k.

Regression with Basis Functions

The simple mathematical model of linear regression, even though it consists a powerful tool, in the same time it is quite limited due to the fact that it considers the input data to be linear, and builds a linear function of the form $y_k = w_0 + \sum_{i=1}^{n} w_i x_{k,i}$. However, in most cases the data are not linear, and therefore, use a line as a predictor is not the most appropriate solution. For this reason, instead of the linear regression model, the *regression with basis functions (RMBF)* is preferred. The RMBFs, with the use of basis functions $\phi : \mathbb{R}^n \rightarrow \mathbb{R}$ can provide us a much more accurate predictor.

Let $\phi_1, \phi_2, \cdots, \phi_M$ be M such basis functions, then each observation x_k is related with M real values $\phi_1(x_k), \phi_2(x_k), \cdots, \phi_M(x_k)$. Now, the predictor takes the form $y_k(x_k) = w_0 + \sum_{i=1}^{M} w_i \cdot \phi_i(x_k)$. The number M of the basis functions and the nature of the basis functions ϕ_i is highly related to the problem at hand, i.e., depends on the nature of the input data and the desirable feature we want to formulate at the time. In order to compute the weight vector in an RMBF, let $\Phi = \begin{bmatrix} \phi_1(x_1), & \phi_2(x_1), & \cdots, & \phi_M(x_1) \\ \phi_1(x_2), & \phi_2(x_2), & \cdots, & \phi_M(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_1(x_K), & \phi_2(x_K), & \cdots, & \phi_M(x_K) \end{bmatrix} \in \mathbb{R}^{K \times M}$ be a matrix that contains all values from each basis function ϕ_i for each observation x_k. Then the weights are computed via the equation $w = (\Phi^T \cdot \Phi)^{-1} \cdot \Phi^T \cdot t$. As in the LRM, in the RMBF by adding a 'dummy' basis function $\phi_0(\mathrm{x}_k) = 1$ for every k observation, we can also compute the bias factor w_0: $\Phi = \begin{bmatrix} 1, & \phi_1(\mathrm{x}_1), & \phi_2(\mathrm{x}_1), & \cdots, & \phi_M(\mathrm{x}_1) \\ 1, & \phi_1(\mathrm{x}_2), & \phi_2(\mathrm{x}_2), & \cdots, & \phi_M(\mathrm{x}_2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1, & \phi_1(\mathrm{x}_K), & \phi_2(\mathrm{x}_K), & \cdots, & \phi_M(\mathrm{x}_K) \end{bmatrix} \in \mathbb{R}^{K \times M+1}$ and the weight vector is $\mathrm{w}^T = \begin{bmatrix} w_0, & w_1, & \cdots, & w_M \end{bmatrix} \in \mathbb{R}^{M+1}$.

The basis functions that should be used in a RMBF is highly dependent on the problem that is under examination at the time; there are various functions that can be used, and their parameters needs tuning based on the input data. However, a very common type of functions that are used in many cases is that of the Gaussian Basis Functions. The Gaussian basis function is an exponential of the form

$$\phi_i(\mathrm{x}_k) = e^{\left\{-\frac{||\mathrm{x}_k - \mu_i||}{2\sigma_i^2}\right\}}$$

where the parameter μ_i reflects the location of the input data for the function ϕ_i, and the parameter σ_i their spatial scale. The parameters μ_i and σ_i needs to be tuned based on the input data of the problem at hand.

Feed Forward Neural Networks

The neural networks (NN) consist a really powerful, somewhat complex, machine learning tool, that over the past decade have gain great attention. The *Feed Forward Neural Networks (FFNN)* is a quite simple type of NNs that allows the information to flow one-way: from the input layer to the output layer. To be more specific, in a FFNN we find the following structures: an *input layer*, a number of intermediate *hidden layers*, the *activation function*, and the *output layer*; as they are depicted in the Figure 2.1. Each node is thought of as a 'neuron' while each arc between the neurons represents a 'synapse', with correspondence to a biological neural network. Each neuron (node) produces as an outgoing message an linear combination of the incoming message which then goes through an activation function f; while each synapse scales with a weight coefficient the message that flows in it. That is, the outgoing message that produces the neuron r in the first layer is of the form: $m_r^{(l_1)} = f(\sum_{i \in \text{connecting } r \text{ with input layer} l_{\text{in}}} w_i^{(l_{\text{in}} \rightarrow l_1)} \cdot m_i^{(l_{\text{in}})})$. A message that flows from the neuron i in the input layer to the first hidden layer though synapse j coincides with the value of the i^{th} dimension of observation x_k. i.e., $m_j^{(l_{\text{in}})} \equiv x_{k,i}$; which within neuron r in the first hidden layer will be multiplied by the synapse j's weight, $w_j^{(l_{\text{in}} \to l_1)} \cdot m_j^{(l_{\text{in}})} \equiv w_j^{(l_{\text{in}} \to l_1)} \cdot x_{k,i}$. Assume that we have in total m hidden layers $(l_1, l_2, \cdots, l_m)$ and the layer l_m contains q neurons (we denote their outgoing messages with $m_1^{(l_m)}, m_2^{(l_m)}, \cdots, m_q^{(l_m)}$); then the output of the FFNN model for observation x_k is given by:

$$y_k = f_{\text{out}}\Big(\sum_{i=1}^{q} w_i^{l_m \to l_{\text{out}}} \cdot m_i^{(l_m)} \Big)$$

The number of hidden layers, the number of neurons in each layer, the weight coefficient in each synapse, and the nature of the activation functions vary and are highly dependent on the problem at hand. The ultimate tuning of the above *hyperparameters* affects the performance of the FFNN, while the selection of these hyperparameters is related to the complexity of the pattern to be learnt. In general, each layer may have different number of neurons, each neuron may be related with different activation function, and each neuron may be connected with different number of synapses (for example in Figure 2.1 each neuron in one layer is connected via synapses with all neurons in the next layer). We can maximize the performance of our FFNN (or NN in general) by optimizing the chosen set of the aforementioned hyperparameters, by using an optimization algorithm.

2.3.2 Probabilistic Topic Modelling

Probabilistic topic models (PTMs) [Blei, 2012] belong to the family of unsupervised machine learning. In fact, probabilistic topic modelling is a statistical approach used in analyzing words of documents that was originally used in data mining to discover a distribution over topics related to a given text document. PTMs were introduced within the linguistic scenario of uncovering underlying (latent) topics in a collection of documents. Topic modeling algorithms have been also adapted to other scenarios as well, for example in genetic data, images and social networks. In this thesis, instead, we are inspired by recent work of [Mamakos and Chalkiadakis, 2018], and employ a widely used PTM algorithm, *online Latent Dirichlet Allocation (LDA)* [Blei et al., 2003], to operate on instances of formed coalitions, in order to discover the ordinal preferences relations of each agent.

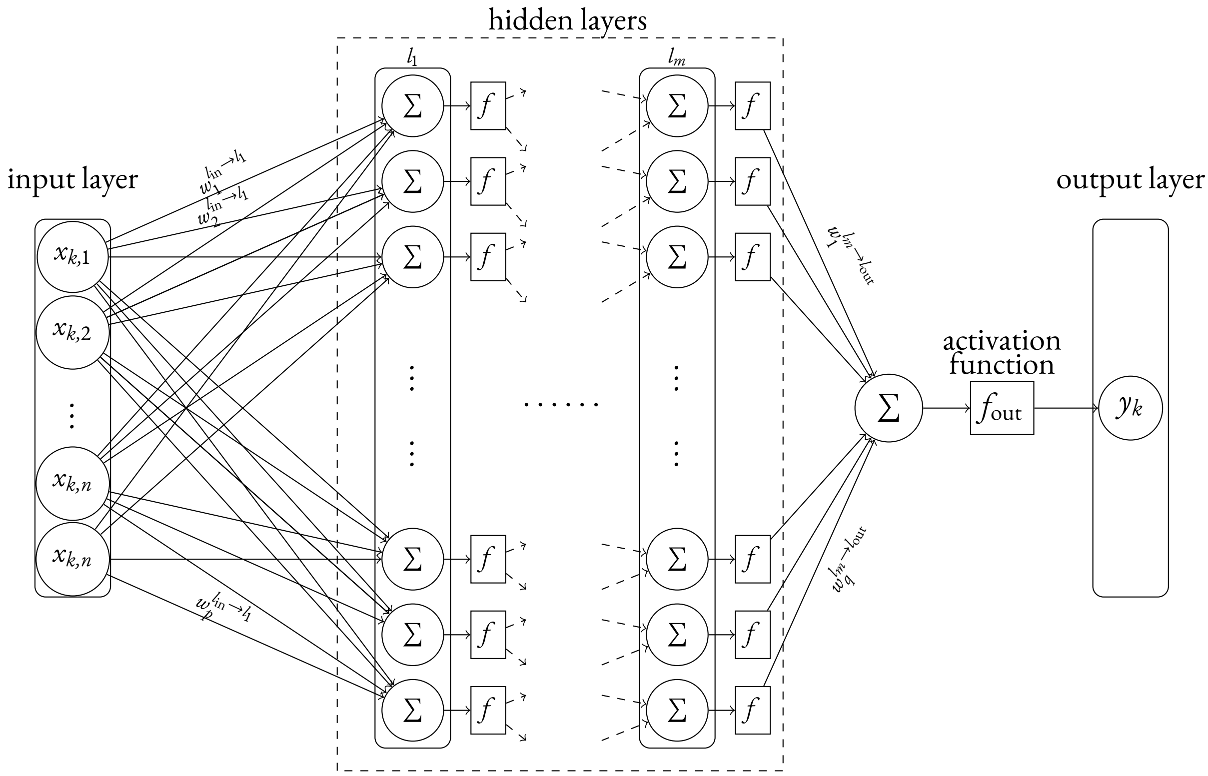


Figure 2.1: Feed Forward Neural Network

Latent Dirichlet Allocation

We first describe the basic terms behind latent Dirichlet allocation (LDA) following [Blei et al., 2003], [Mamakos and Chalkiadakis, 2018], and [Georgara et al., 2019a]:

- A *word* is the basic unit of discrete data. A vocabulary consists of words and is indexed by $\{1,2,...,V\}$. The vocabulary is fixed and is fed as input to the LDA model.
- A *document* is a series of L words, $(w_1, w_2, \dots, w_L)$.
- A *corpus* is a collection of D documents.
- A *topic* is a distribution over a vocabulary.

LDA is a Bayesian probabilistic topic model, in which each document can be described by a mixture of topics. A generative process for each document in the collection is assumed by LDA, where a random distribution over topics is chosen, and for each word in a document a topic is chosen from the distribution; finally, a word is chosen from the chosen topic. The same set of topics is shared by the documents of the corpus, but each document exhibits topics in different proportions.

Latent variables, describing the hidden structure LDA indents to uncover, are assumed to be included to the generative process. The topics are $\beta_{1:K}$, where K is the dimensionality of the topic variable, which is known and fixed. Each topic β_k, is a distribution over the vocabulary of the corpus, where $k \in \{1, \ldots, K\}$; and β_{kw} is the probability of word w in topic k. For the d^{th} document, ϑ_d is the distribution over topics; and ϑ_{dk} is the topic proportion of topic k in d. The topic assignments for the d^{th} document are indicated by z_d, and the topic assignment for the l^{th} word of the d^{th} document is denoted by z_{dl}. Consequently, w is the only observed variable of the model and w_{dl} represents the l^{th} word seen in the d^{th} document, while β, ϑ and z are latent variables. The posterior of the topic structure given the documents is:

$$p(\beta_{1:K}, \vartheta_{1:D}, z_{1:D} \mid w_{1:D}) = \frac{\beta_{1:K}\vartheta_{1:D}z_{1:D}w_{1:D}}{p(w_{1:D})}$$

where D is the number of documents, and the computation of the denominator, i.e. the probability of seeing the given document under any topic structure, is intractable [Blei et al., 2003]. Moreover, LDA includes priors, so that β_k is drawn from a Dirichlet distribution with parameter η and ϑ_d is drawn respectively from a Dirchlet also, with parameter α.

As mentioned, the posterior cannot be computed. To approximate it, the two most prominent approaches are (a) variational inference introduced in [Jordan et al., 1999] and (b) Markov Chain Monte Carlo sampling methods proposed in [Jordan, 1999]. In variational inference, the true posterior is approximated by a simpler distribution q, which depends on parameters $\phi_{1:D}$,$\gamma_{1:D}$ and $\lambda_{1:K}$ defined as:

$$\phi_{dwk} \propto \exp\{E_q[\log \vartheta_{dk}] + E[\log \beta_{kw}]\},$$

$$\gamma_{dk} = \alpha + \sum_{w} n_{dw}\phi_{dwk}\lambda_{kw} = \eta + \sum_{d} n_{dw}\phi_{dwk}$$

The probability that topic assignment of word w in d is k, under distribution q, is denoted by ϕ_{dwk}. Variable n_{dw} represents how many times the word w has been seen in document d. The variational parameters $\gamma_{1:D}$ and $\lambda_{1:K}$ are associated with variable n_{dw}. The variational inference algorithm's intuition is to minimize the *Kullback-Leibler(KL) divergence* between the variation distribution and the true (intractable) posterior. This is accomplished by iterating between assigning values to document-level variables, and updating topic-level variables (see Algorithm 1).

Algorithm 1: Variational Inference for LDA [Blei et al., 2003]

Randomly initialize λ;
repeat
    Expectation step:
    for ($d = 1 \rightarrow D$):
        $\gamma_{dk} = 1$;
        repeat
            Set $\phi_{dwk} \propto \exp\{E_q[\log \vartheta_{dk}] + E_q[\log \beta_{kw}]\}$;
            Set $\gamma_{dk} = \alpha + \sum_w n_{dw} \cdot \phi_{dwk}$;
        until ($\frac{1}{K} \cdot \sum_{k=1}^{K}$ *change in* $\gamma_{dk}|| < \varepsilon$);
    Maximization step:
    Set $\lambda_{kw} = \eta + \sum_{d=1}^{D} n_{dw} \cdot \phi_{dwk}$;
until *(relative KL divergence has not significantly decreased)*;

Online Latent Dirichlet Allocation In the online version of LDA topic model [Hoffman et al., 2010], the documents are received on streams rather than a single batch of the original LDA algorithm. In this approach, the exact number of documents is not required to be known, though an estimation is at least required. As a result, online LDA can adapt to very large corpora. The value of the variational parameter $\lambda_{1:K}$ is updated every time a new batch arrives, while the rate at which the documents of batch t actually affects the value of $\lambda_{1:k}$ is controlled by $\varrho_t = (\tau_0 + t)^{-k}$. The variational inference for the online version of LDA is shown in Algorithm 3, where it eventuates that α and η are assigned to a value once, and remain fixed. Ultimately, the estimated probability of the term w in topic k is $\beta_{kw} = \frac{\lambda_{kw}}{\sum \lambda_k}$.

Algorithm 3: Online Variational Inference for LDA [Hoffman et al., 2010]

```
Randomly initialize λ;
for ( t = 1 → ∞ ):
    ϱ_t = (τ_0 + t)^{-k};
    Expectation step:
    Randomly initialize γ_tk;
    repeat
        Set φ_twk ∝ exp{E_q[log ϑ_tk] + E_q[log β_kw]};
        Set γ_tk = α + Σ_w n_tw φ_twk;
    until ((1/k) · Σ |change in γ_tk| ≤ ε);
    Maximization step:
    Compute λ̃_kw = η + D n_nt φ_twk;
    Set λ = (1 − ϱ_t)λ + ϱ_t λ̃;
```

Chapter 3

Hedonic Games in Partition Function Form

In this thesis we study hedonic games, and we formally introduce some theoretical extensions that allow such cooperative settings to model real-life application more accurately. In this section we discuss the natural extension of classic hedonic games that consider coalition-wide preference relations into hedonic games in partition function form (PFF-HGs) that consider partition-wide preference relations. To best to our knowledge, all works regarding hedonic games so far express preferences over coalition; and [Aziz et al., 2016b] specifically notes that in hedonic games each agent is interested only in the members of her coalitions, and does not care about the composition of other coalitions. In real-world though, the composition of other coalitions can, and in many cases does, affect the preferences of individuals. Thus, here we put forward the formal definition of *Hedonic Games in Partition Function Form*, and formally express classes of classic hedonic games to their partition function form extension [Georgara et al., 2019b].

As such, in this chapter we contribute in the hedonic games as follows:

- we provide the formal definition of *Hedonic Games in Partition Function Form*;
- we extend well-known classes of hedonic games to their partition function form; and
- we propose a propositional language to represent Boolean Hedonic Games in Partition Function Form.

The work presented in this chapter is included in [Georgara et al., 2019b].

3.1 What is the *Partition Function Form* of a game?

In cooperative games, one may come across the term *partition function game*, or *game with externalities*, or *game in partition function form* (see, eg., [Chalkiadakis et al., 2011, Michalak et al., 2008, Michalak et al., 2009]). All these terms refer to and are used to describe the same notion: they capture situations where achieved by the agents utility (either within a transferable or non-transferable setting) does not depend solely on a single coalition's composition, but actually on the formation of all groupings. For instance, in [Michalak et al., 2008] the authors examined the behaviour of four different types of Partition Function Games (super-additive with positive/negative externalities, and sub-additive with positive/negative externalities), and proved that in such settings they can bound the coalitional values in order to use the state-of-the-art *coalition structure generation* algorithm to obtain optimal coalition structures. One step further, in [Michalak et al., 2009] the authors proposed new representations for coalitional games with externalities, and study various notions of externalities; while in [Michalak et al., 2010] the authors work with computing extensions of the Shapley value to coalitional games with externalities. Nonetheless, all works cited above consider Transferable Utility Games, and not Hedonic Games, as we do in this work.

In order to conceive the kind of settings that captures a game expressed in partition function form, consider the following:

Example 2. Participation in AI coding competition *Consider n programmers that take part in an AI coding competition. In this competition the programmers form groups to work together in order to build intelligent software that plays a board game (e.g. Go), that will eventually compete each other in a tournament. Let $f(S)$ be a function that expresses the power of the intelligent software coalition S builds. The final utility of a coalition S though depends also on the power of their opponents, i.e.* $v(S) = \sum_{T \in CS \setminus S} \left(f(S) - f(T)\right)$.

In the above example, we see that there are cases in cooperative games, where it is very natural someone's reward to be affected by all coalitions formed. Another example described in [Chalkiadakis et al., 2011] refers to an economic setting where the composition of all coalitions in the formed coalition structure regulates the price of the produced good, and therefore the utility achieved by each coalition. Clearly, the Example 2, which is stated as a TU game, can be rephrased into an NTU game by indicating that each programmer receives different level of professional education[1] (that cannot be transferred). Thus the personal utility function of each agent $i \in S$ will be described as $v_i(S) = \sum_{T \in CS \setminus S} \big(f_i(S) - f_i(T)\big)$, where $f_i(S)$ expresses how agent i perceives the capabilities of a team.

Within a hedonic game framework, consider the following example.

Example 3. Synergies for Government Contracts *Think of a number of companies that establish synergies in order to acquire government contracts. Each company aims to establish the best for it synergy, but the satisfaction reached by the collaboration is certainly affected by the synergies of the rest of the companies, as well. In particular, consider four companies, i.e.* $N = \{a, b, c, d\}$*; then company a wants to team up with company b as working together is highly satisfactory. In the same time, however, a prefers companies c and d to also establish a synergy, since due to some former rivalry such a synergy would repulse the contract auctioneer; that is,* $CS = \big\{\{a, b\}, \{c, d\}\big\} \succ_a CS' = \big\{\{a, b\}, \{c\}, \{d\}\big\}$.

3.2 Formal Definition

In order to formally define a game in partition function form, we first need to define some auxiliary entities. Let $N = \{1, \cdots, n\}$ be a finite, non-empty set of players of size $|N| = n$. The set $N_i = \{S \subseteq N : i \in S\}$ is the collection of all coalitions that contain agent i, and $CS(i) = S \in CS : i \in S$ denotes the single coalition S within coalition structure CS that contains agent i. We denote the set of all coalition structures that can result from the set of players N as $\mathscr{CS}_N$, i.e., $\mathscr{CS}_N$ contains all the possible partitions of set N. We relate a specific coalition to a specific coalition structure referring to this relation as *embedded coalition*.

Definition 15. Embedded Coalition *An* embedded coalition *is given by a pair* $\langle S, CS \rangle$*, where* S *is a coalition,* $CS \in \mathscr{CS}_N$ *is a possible coalition structure over* N*, and* $S \in CS$.

The set of all embedded coalition over N is denoted as $\mathscr{E}_N = \{\langle S, CS\rangle\ \forall S \in CS, \forall CS \in \mathscr{CS}_N\}$; while the set of all embedded coalitions over N that contains a specific agent i is denoted as $\mathscr{E}_N(i) = \{\langle CS(i), CS\rangle\ \forall CS \in \mathscr{CS}_N\}$. In words, an embedded coalition corresponds

[1] Participating in such a competition, apart from any actual cash-prize for the winning team, all participants gain knowledge and experiences, train their skills, and get stimulated by others' ideas. In Chapter 5 we will see this example again, under a new hybrid model of games.

to a coalition in a fixed coalition structure.

Generally, a game in partition function form is described as follows ([Chalkiadakis et al., 2011, Saad et al., 2012]):

Definition 16. NTU/TU-PFF *A game G in PFF is given by a pair* $\langle N, v \rangle$*, where N is a set of agents, and*

- *if G is TU game,* $v : \mathscr{E}_N \rightarrow \mathbb{R}$ *is a utility function that maps each embedded coalition* $\langle S, CS \rangle \in \mathscr{E}_N$ *to a unique real value*
- *if G is NTU game,* $v : \mathscr{E}_N \rightarrow \mathbb{R}^n$ *is a mapping that associates each embedded coalition* $\langle S, CS \rangle \in \mathscr{E}_N$ *to a real n-vector (with the understanding that all agents not in the embedded coalition,* $\forall\ i \notin \langle S, CS \rangle$*, receive zero utility).*

Now, consider the classic hedonic games, where each agent expresses hedonic preferences over coalitions. By extending hedonic games into partition function form, we require each agent to define a preference relation over *embedded coalitions*, i.e., each agent expresses a partition-wide preference relation instead of a coalition-wide one. Therefore, we define a hedonic game in partition function form as follows:

Definition 17. PFF-HG *A hedonic game in partition function form (PFF-HG) G is given by a pair* $\langle N, \succsim \rangle$*, where N is a set of players;* $\succsim = \{\succsim^1, \cdots, \succsim^n\}$ *, and each* $\succsim_i \subseteq \mathscr{E}_N(i) \times \mathscr{E}_N(i)$ *is a complete, reflexive and transitive preference relation that captures agent i's preferences over the embedded coalitions that contains i.*

In words, we expand the classic hedonic games into hedonic games in partition function form by widening the coalitional space. Now the coalitional space does not simply contain coalitions with different compositions, but it contains all the embedded coalitions, i.e., all the different coalition structures that can result from a set of players N. That is, given an PFF-HG $G = \langle N, \succsim \rangle$, each agent defines her preferences for each pair of embedded coalitions as follows: let $\langle S, CS_1 \rangle \in \mathscr{E}_N(i)$, $\langle T, CS_2 \rangle \in \mathscr{E}_N(i)$, and $\langle U, CS_3 \rangle \in \mathscr{E}_N(i)$ be three embedded coalitions; then agent i prefers $\langle S, CS_1 \rangle$ over $\langle T, CS_2 \rangle$ and $\langle T, CS_2 \rangle$ over $\langle U, CS_3 \rangle$ if and only if $\langle S, CS_1 \rangle \succsim_i \langle T, CS_2 \rangle \succsim_i \langle U, CS_3 \rangle$; what it is to be noted is that coalitions S and U may be exactly the same, however the grouping of other agents affect agent i's preferences.

Example 4. *Let an PFF-HG* $G = \langle N, \succsim \rangle$*, with* $N = \{1, 2, 3\}$*. The space of embedded coalitions is the following*

$$\mathscr{E}_N = \Big\{ \big\langle \{1\}, \big\{\{1\}, \{2\}, \{3\}\big\} \big\rangle, \big\langle \{2\}, \big\{\{1\}, \{2\}, \{3\}\big\} \big\rangle, \big\langle \{3\}, \big\{\{1\}, \{2\}, \{3\}\big\} \big\rangle, \big\langle \{1\}, \big\{\{1\}, \{2, 3\}\big\} \big\rangle,$$
$$\big\langle \{2, 3\}, \big\{\{1\}, \{2, 3\}\big\} \big\rangle, \big\langle \{2\}, \big\{\{2\}, \{1, 3\}\big\} \big\rangle, \big\langle \{1, 3\}, \big\{\{2\}, \{1, 3\}\big\} \big\rangle,$$
$$\big\langle \{3\}, \big\{\{3\}, \{1, 2\}\big\} \big\rangle, \big\langle \{1, 2\}, \big\{\{3\}, \{1, 2\}\big\} \big\rangle, \big\langle \{1, 2, 3\}, \big\{\{1, 2, 3\}\big\} \big\rangle \Big\};$$

and the preference relation of agent 1 *is:*

$$\big\langle \{1, 2, 3\}, \big\{\{1, 2, 3\}\big\} \big\rangle \succsim_1 \big\langle \{1, 2\}, \big\{\{3\}, \{1, 2\}\big\} \big\rangle \succsim_1$$
$$\big\langle \{1\}, \big\{\{1\}, \{2\}, \{3\}\big\} \big\rangle \succsim_1 \big\langle \{1, 3\}, \big\{\{2\}, \{1, 3\}\big\} \big\rangle \succsim_1 \big\langle \{1\}, \big\{\{1\}, \{2, 3\}\big\} \big\rangle.$$

As we can see here, agent 1 *prefers being alone, i.e.,* $\{1\}$*, rather be in a coalition with agent* 3 *when agent* 2 *is singleton as well, i.e.,* $\langle \{1\}, \big\{\{1\}, \{2\}, \{3\}\big\} \rangle \succsim_1 \langle \{1, 3\}, \big\{\{2\}, \{1, 3\}\big\} \rangle$*; at the same time, however, agent* 1 *prefers a coalition with agent* 3 *than being alone when a collaboration between agents* 3 *and* 2 *is in place, i.e.,* $\langle \{1, 3\}, \big\{\{2\}, \{1, 3\}\big\} \rangle \succsim_1 \langle \{1\}, \big\{\{1\}, \{2, 3\}\big\} \rangle$*.*

By now, it must be clear enough that in a PFF-HG the very same coalition can be more or less preferable to some agent regarding the collaborations of the other agents taking place.

In fact, the hedonic games in partition function form can be though of as a *generalization* of the classic hedonic games. It is not hard to see that if we let each agent i be indifferent among embedded coalitions containing the same coalition $S \in N_i$, then the resulted game is a classic hedonic game. That is, let an Pff-HG $G = \langle N, \succsim \rangle$, if for each pair $CS, CS' \in \mathscr{C}\mathscr{S}_N$ and for each agent i we have that

- $\langle S, CS \rangle \sim_i \langle T, CS' \rangle$ if and only if $S \equiv T$; and
- $\langle S, CS \rangle \succsim_i \langle T, CS' \rangle$ or $\langle S, CS \rangle \precsim_i \langle T, CS' \rangle$ if and only if $S \neq T$

then the game G from an PFF-HG is reduced into a classic hedonic game.

3.3 Classes of Hedonic Games in Partition Function Form

In Section 2.1.3 we presented some well-studied classes of hedonic games, in this section we introduce the partition function form of these classes.

3.3.1 Additively Separable Hedonic Games

First we will examine the class of Additively Separable Hedonic Games (ASHG-PFF). As we already have mentioned, in such games agents assign a value $M_{i,j}$ to each other, which are then used to define the preferences over coalitions. The transition to the partition function form is requires each agent i to assign a value $M_{i,j}$ to any other agent j within each different coalition structure. Specifically, for each $CS \in \mathscr{CS}_N$, any agent i declares a vector:

$$M_i(CS) = \begin{bmatrix} M_{i,1}(CS) \\ M_{i,2}(CS) \\ \vdots \\ M_{i,n}(CS) \end{bmatrix}$$

therefore, the value of the embedded coalition $\langle S, CS \rangle$ for agent i is $v_i(S, CS) = \sum_{j \in S} M_{i,j}(CS)$. Now, for the preference relation it holds that agent i prefers the embedded coalition $\langle S, CS \rangle \in \mathscr{E}_N(i)$ over $\langle T, CS' \rangle \in \mathscr{E}_N(i)$, $\langle S, CS \rangle \succsim_i \langle T, CS' \rangle$, if and only if $v_i(S, CS) = \sum_{j \in S} M_{i,j}(CS) \geq v_i(T, CS') = \sum_{j \in T} M_{i,j}(CS')$. Note that for two embedded coalitions $\langle S, CS_1 \rangle \in \mathscr{E}_N(i)$ and $\langle S, CS_2 \rangle \in \mathscr{E}_N(i)$ agent i corresponds two values $v_i(S, CS_1) = \sum_{j \in S} M_{i,j}(CS_1)$ and $v_i(S, CS_2) = \sum_{j \in S} M_{i,j}(CS_2)$; here, even though the composition of coalition S in both embedded coalitions is the same, their values differ.

3.3.2 The classes $\mathscr{B}/\mathscr{W}-$ and Fractional Hedonic Games

Following the same pattern, in the classes $\mathscr{B}/\mathscr{W}-$ and fractional hedonic games ($\mathscr{B}/\mathscr{W}-$HG-PFF & FHG-PFF), each agent i assigns to each other agent j a value for each different coalition structure $CS \in \mathscr{CS}_N$. Therefore, the transition to partition function form is straightforward.

In particular, for a $\mathscr{B}-$ hedonic game in partition function form, for a given agent $i \in N$ an embedded coalition $\langle S, CS \rangle \in \mathscr{E}_N(i)$ is more preferable than some other embedded coalition $\langle T, CS' \rangle \in \mathscr{E}_N(i)$ if the *most preferable agent* in $\langle S, CS \rangle$ is more preferable than the most preferable agent in $\langle T, CS' \rangle$. That is, $\langle S, CS \rangle \succsim_i \langle T, CS' \rangle$ if and only if $v_i(S, CS) = \max_{j \in S}\{M_{i,j}(CS)\} \geq v_i(T, CS') = \max_{j \in T}\{M_{i,j}(CS')\}$. Once again note that coalitions S and T may be exactly the same, however the most preferable agent may be different, or even if the most preferable agent remains the same the value assigned to this agent may be different depending on the overall coalition structure.

Respectively, for a $\mathscr{W}-$ hedonic game in partition function form, for a given agent $i \in N$ an embedded coalition $\langle S, CS\rangle \in \mathscr{E}_N(i)$ is more preferable than some other embedded coalition $\langle T, CS'\rangle \in \mathscr{E}_N(i)$ if the *least preferable agent* in $\langle S, CS\rangle$ is more preferable than the lest preferable agent in $\langle T, CS'\rangle$. That is, $\langle S, CS\rangle \succsim_i \langle T, CS'\rangle$ if and only if $v_i(S, CS) = \min_{j\in S}\{\mathcal{M}_{i,j}(CS)\} \geq v_i(T, CS') = \min_{j\in T}\{\mathcal{M}_{i,j}(CS')\}$. In the case the coalitions S and T coincide, the least preferable agent may be different, or even if the least preferable agent remains the same the value assigned to this agent may differ.

Last but not least, in order to generalize fractional hedonic games to their partition function form, for a given agent $i \in N$ an embedded coalition $\langle S, CS\rangle \in \mathscr{E}_N(i)$ is more preferable than some other embedded coalition $\langle T, CS'\rangle \in \mathscr{E}_N(i)$ if the *i averagely* prefers the players in $\langle S, CS\rangle$ than the players in $\langle T, CS'\rangle$. That is, $\langle S, CS\rangle \succsim_i \langle T, CS'\rangle$ if and only if $v_i(S, CS) = \frac{\sum_{j\in S}\{\mathcal{M}_{i,j}(CS)\}}{|S|} \geq v_i(T, CS') = \frac{\sum_{j\in T}\{\mathcal{M}_{i,j}(CS')\}}{|T|}$.

3.3.3 Boolean Hedonic Games

For the expansion of Boolean Hedonic Games into their partition function form (BHG-PFF) the key idea is for each agent to distinguish the embedded coalitions into good and bad ones. That is, each agent labels each embedded coalition in $\mathscr{E}_N(i)$ as *preferable* or *non-preferable*. Formally, let us denote with P_i^+ the collection with all preferable embedded coalitions for agent i, and, respectively, with P_i^- all non-preferable embedded coalitions. Therefore, in a boolean hedonic game $G = \langle N, \succsim\rangle$ in partition function form, it hold that the embedded coalition $\langle S, CS\rangle \in \mathscr{E}_N(i)$ is strictly preferable by agent i over the embedded coalition $\langle T, CS'\rangle \in \mathscr{E}_N(i)$, i.e., $\langle S, CS\rangle \succ_i \langle T, CS'\rangle$ if and only if $\langle S, CS\rangle \in P_i^+$ and $\langle T, CS'\rangle \in P_i^-$. Moreover, it holds that $\langle S, CS\rangle \sim_i \langle T, CS'\rangle$ if and only if $\langle S, CS\rangle, \langle T, CS'\rangle \in P_i^+$ *or* $\langle S, CS\rangle, \langle T, CS'\rangle \in P_i^-$.

Now let us describe a propositional formula that can provide us with a compact representation of each agent's $i \in N$ preference relation. In fact in [Aziz et al., 2016a] the authors by introducing a propositional logic for preference relation representation, jump to a description of preference relation over partition. They do so by lifting preference over coalitions into preferences over partitions and by being indifferent about partitions where the agent belongs to the same coalition. Consequently, this can actually be thought of as a prime attempt to express a boolean hedonic game in partition function form. First, we build on the idea of the logic developed in [Aziz et al., 2016a], and adjust this logic so that it anticipates to the needs of a BHG in partition function form in general.

A propositional language for BHG-PFF preference representation

As in [Aziz et al., 2016a], let L_N be a propositional language containing all the classical connectives, $\bot, \top, \wedge, \rightarrow, \leftrightarrow$, and propositional variables $p_{i,j}$ for each pair of players where $i < j$. Therefore, a coalition structure can be described by these propositional variables, where intuitively each $p_{i,j}$ answers the question 'is agent i and agent j in the same coalition?'. [2]

As we have already discussed in Section 2.1.3, each agent uses a logic formula to declare her goal, i.e., which collaborative patterns a desirable coalition structure should satisfy. Azis et al, let a propositional formula ϕ_i that describes agents i's goal contain only propositional variables related to i; that is ϕ_i can be constructed only by literals of the form $p_{i,j}$ where $j \in N$. Moreover, we remind the reader that each goal ϕ_i is built from a number of different collaboration patterns expressed as $\phi_i = \phi_{i,1} \vee \cdots \vee \phi_{i,k}$, where each collaboration pattern $\phi_{i.j}$ consists only of variables related to agent i.

Now, in order to describe coalition structures preferences, an agent i needs to declare collaboration patterns not only between herself and the other agents, but also among the other agents in general. Let us denote with $\gamma_{i,j}$ the j^{th} collaboration pattern of agent i's goal–we use the symbol γ to distinguish it from ϕ and avoid any vagueness that may arise. Thus, a collaboration pattern $\gamma_{i,j}$ can now contain variables $p_{k,l}$ where $k, l \in N$ and $k, l \neq i$; that is, $\gamma_{i,j}$ is written as:

$$\gamma_{i,j} = \bigwedge_{\{k,l\} \in \text{Appealing Collaborations}_j} p_{k,l} \bigwedge_{\{m,o\} \in \text{Repellent Collaborations}_j} \neg p_{m,o}$$

which denotes that agent i's goal can be achieved if each pair $\{k, l\}$ in "Appealing Collaborations$_j$" set coexist in some coalition, and no pair $\{m, o\}$ in "Repellent Collaborations$_j$" set coexist in some coalition. Specifically, each coalition structure $CS \in \mathscr{CS}_N$ such that:

- $\forall\, c \in$ Appealing Collaborations$_j$ $\exists\, S \in CS : c \subseteq S$; and
- $\forall\, c \in$ Repellent Collaborations$_j$ $\nexists\, S \in CS : c \subseteq S$

[2] Since we want to indicate if agents i and j are in the same coalition there is no need of having both propositional variables $p_{i,j}$ and $p_{j,i}$ as they both encode the exact same information. This is the reason why we use only variables $p_{i,j}$ where $i < j$, as the variables $p_{j,i}$ are redundant.

satisfies the collaboration pattern $\gamma_{i,j}$ and it is considered as a desirable to agent i coalition structure. In words, if for every desirable pattern $c \in$ Appealing Collaborations$_j$, there is a coalition $S \in CS$ that contains c (i.e, $c \subseteq S$); while for each unwanted pattern $c \in$ Repellent Collaborations$_j$ there is no coalition $S \in CS$ such that c is contained in S, then the coalition structure CS satisfies formula $\gamma_{i,j}$.

Example 5. *Let a BHG-PFF $G = \langle N, \gamma \rangle$, where $N = \{1, 2, 3, 4\}$ is the set of players and $\gamma = \{\gamma_1, \gamma_2, \gamma_3, \gamma_4\}$ is a propositional formula, where each γ_i describes the i's agent goal. Let agent 1's goal be described through the formula*

$$\gamma_1 = \gamma_{1,1} \vee \gamma_{1,2} \vee \gamma_{1,3}$$
$$\underbrace{(p_{1,2} \wedge \neg p_{2,4})}_{\gamma_{1,1}} \vee \underbrace{(p_{2,3} \wedge \neg p_{2,4})}_{\gamma_{1,2}} \vee \underbrace{(p_{1,3} \wedge p_{1,4} \wedge \neg p_{2,3})}_{\gamma_{1,3}}$$

Thus, according to γ_1 the embedded coalitions that are positioned in P_1^+ are the following (we use different colours to indicate which collaboration pattern satisfies each embedded coalition– $\gamma_{1,1}$, $\gamma_{1,2}$,$\gamma_{1,3}$) The space of embedded coalitions related to agent 1 is:

$$\mathscr{E}_N(1) = \Big\{\langle\{1\}, \{\{1\}, \{2\}, \{3\}, \{4\}\}\rangle, \langle\{1\}, \{\{1\}, \{2, 3\}, \{4\}\}\rangle, \langle\{1\}, \{\{1\}, \{2, 4\}, \{3\}\}\rangle, \langle, \langle\{1\}, \{\{1\}, \{2\}, \{3, 4\}\}\rangle,$$
$$\langle\{1\}, \{\{1\}, \{2, 3, 4\}\}\rangle, \langle\{1, 2\}, \{\{1, 2\}, \{3\}, \{4\}\}\rangle, \langle\{1, 2\}, \{\{1, 2\}, \{3, 4\}\}\rangle, \langle\{1, 3\}, \{\{1, 3\}, \{2\}, \{4\}\}\rangle,$$
$$\langle\{1, 3\}, \{\{1, 3\}, \{2, 4\}\}\rangle, \langle\{1, 4\}, \{\{1, 4\}, \{2\}, \{3\}\}\rangle, \langle\{1, 4\}, \{\{1, 4\}, \{2, 3\}\}\rangle, \langle\{1, 2, 3\}, \{\{1, 2, 3\}, \{4\}\}\rangle,$$
$$\langle\{1, 2, 4\}, \{\{1, 2, 4\}, \{3\}\}\rangle, \langle\{1, 3, 4\}, \{\{1, 3, 4\}, \{2\}\}\rangle, \langle\{1, 2, 3, 4\}, \{\{1, 2, 3, 4\}\}\rangle\Big\}$$

thus the preferable embedded coalitions for agent 1 are:

$$P_1^+ = \Big\{\langle\{1\}, \{\{1\}, \{2, 3\}, \{4\}\}\rangle, \langle\{1, 2\}, \{\{1, 2\}, \{3\}, \{4\}\}\rangle, \langle\{1, 2\}, \{\{1, 2\}, \{3, 4\}\}\rangle,$$
$$\langle\{1, 4\}, \{\{1, 4\}, \{2, 3\}\}\rangle, \langle\{1, 2, 3\}, \{\{1, 2, 3\}, \{4\}\}\rangle, \langle\{1, 3, 4\}, \{\{1, 3, 4\}, \{2\}\}\rangle\Big\}$$

Any other embedded coalition $\langle S, CS\rangle \in \mathscr{E}_N(1)$ cannot satisfy any collaboration pattern of $\gamma_{1,1}$,$\gamma_{1,2}$ or $\gamma_{1,3}$. they cannot satisfy formula γ_1, and therefore they are labelled as non-preferable embedded coalitions, i.e they are positioned in P_1^- set.

3.4 Real-life Settings Modelled by PFF-HGs

Let us now go through some real-life settings that can be modelled more accurately by a hedonic game in partition function form; such as the example 7 we presented in Section 3.1.

Example 6. Political Parties *Consider* $n = 8$ *political parties,* $N = \{a, b, c, d, e, f, g, h\}$, *each of which have its own political line, ideas and perspectives. In order to form legal government, the parties need to collaborate with each other and form coalitions. From party's a point of view, parties d and h share some common ideas with a, and therefore they all can establish a satisfying collaboration; at the same time party d is in good amends with parties b and g, so a believes that a synergy between the two,* $\{b, g\}$, *would benefit coalition* $\{a, d, h\}$. *Therefore the coalition structure* $\{\{a, d, h\}, \{b, g\}, \{c\}, \{e\}, \{f\}\}$ *is more preferable to agent a over the coalition structure* $\{\{a, d, h\}, \{b\}, \{g\}, \{c\}, \{e\}, \{f\}\}$, *i.e.,* $\{\{a, d, h\}, \{b, g\}, \{c\}, \{e\}, \{f\}\} \succ_a \{\{a, d, h\}, \{b\}, \{g\}, \{c\}, \{e\}, \{f\}\}$.

Example 7. Companies Synergies *Consider* 3 *companies that establish synergies in order to acquire government contracts. In each competition the companies form different coalitions, and the bundling of the contracts leads to different satisfaction for each company. A collaboration of two specific companies, namely a and b, results a higher satisfaction to company c as this synergy prevents the coalition* $\{a, b\}$ *from acquiring the contract due to some former rivalry. Thus,* $\{\{a, b\}, \{c\}\} \succ_c \{\{a\}, \{b\}, \{c\}\}$.

Example 8. Crowdsourcing Platforms *Another example could be crowdsourcing online platforms, where given a complicated task and a specific time frame, individuals are to form workgroups to solve problems or puzzles. The solutions to the individual problems combine a solution to the overall complicated task. Thus, each participant cares about all groups' composition, as each coalition solves a part of the desired overall task.*

Example 9. Company Domain Structure *Similarly to the concept in the previous example (8), each employer in each domain within a company is interested on the composition of his/her own domain. Yet, they are also interested in other domain's composition as all domains must work in harmony for the company's prosperity.*

Example 10. Term Project Teams *Consider a course in Multi-agent Systems in a Computer Science School. The students, in order to be successful in the course, are to work in groups to fulfil* *a project. Ultimately, all the projects are to compose a larger project that affects the students' grade. Therefore, each student is interesting in the overall grouping instead of solely their own coalition.*

Chapter 4

Learning Hedonic Preferences in Cooperative Games

In real-life it would be frivolous and superficial to believe that one is aware of the complete preference relations among not only other agents, but also among that agent herself and the other agents. For this reason, it is essential for an agent to be able to learn the underlying hidden game. This in fact was the motivation for the authors in [Sliwinski and Zick, 2017] to explore the *Probably A proximately Correct (PAC) learnability* of several classes of hedonic games: it studied how good probabilistic "hedonic" utility function approximations can be derived from a (polynomial) number of samples, and proposed algorithms to do so for specific problem instances in the process.

We let each agent $i \in N$ train and maintain her own learning model. That is, each agent i attempts to learn her own preferences via interacting with others as she observes coalitions and partitions. In this thesis, we studied the problem of uncertainty in hedonic preferences in the essence that each agent is unaware of either the $\mathcal{M}_{i,j}$ values or the logic formulae that she "subconsciously" follow to determine her preferences.

This chapter consists the practical part of this thesis, and as such, our contributions here are as follows:

- we conduct a systematic evaluation of our learning technique;
- we develop two evaluation metrics for the models' performance; and

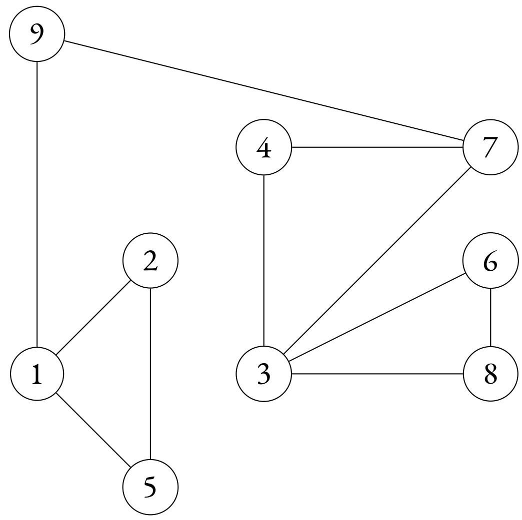


Figure 4.1: Social graph example

- we propose an interpretation method for converting coalition into documents.

The work presented in Sections 4.3.2, 4.4.2, 4.5 also appear in [Georgara et al., 2019b]; while the work in Section 4.6 in [Georgara et al., 2019a].

4.1 A framework for learning hedonic preferences

In order to evaluate the performance of several learning models on the problem of learning preferences within hedonic games, we need to specify the games' framework. We will specify two different frameworks depending on the class of game to be learnt. In particular, the first framework generates hedonic games in the classes of ASHGs, $\mathscr{B}/\mathscr{W}-$ Hedonic Games and Fractional Hedonic Games; while the second framework generates hedonic games with dichotomous preferences.

4.1.1 Preferences based on an underlying social graph

In Additively Separable Hedonic Games, $\mathscr{B}/\mathscr{W}-$ Hedonic Games and Fractional Hedonic Games the preferences over coalitions of an agent i are the lifted preferences over agents; that is, agent i develops preferences over the rest of the agents, expressed through the values $M_{i,j}$ $\forall j \in$

N, this values $M_{i,j}$ are then go through some known function building the preferences over coalitions (or over partitions). These values may be arbitrary, or follow some a priori known distribution. Here we suggest that this values derive from a graph.

Suppose we have a network as the one in Figure 4.1. Each node represents an agent while the edges represents the a social link between the agents. That is, if two nodes n_i and n_j are connected via an edge $e_{i,j}$ it means that agents i and j have a communication link, and intuitively this communication link is interpreted as motive for collaboration. For instance in [Igarashi and Elkind, 2016] the authors considered a environment where agents should be connected according to some graph structure in order to be able to form a coalition.

In this light, let i be an agent and g a social graph, we distinguish the following two distances:

1. the distance $\text{dist}_g(i, j)$ between i and some other agent j, which is given by the minimum path from node n_i to node n_j;

2. the distance $\text{dist}_g(i, S)$ between i and some coalition S, which is given by the average distance between i and the agents $j \in S$, i.e., $\text{dist}_g(i, S) = \frac{\sum_{j \in S} \text{dist}_g(i,j)}{|S|}$.

Intuitively, the distance between two agents answers the question "how much close friends are agent i with agent j?"; thus, given some graph, we can use the distances to form a distribution of $M_{i,j}$ values for each agent. The distance between an agent a coalition, on the other hand, can be interpreted as the tendency of the agent to join this coalition, depending on how much close friends is the agent with the agents within the coalition.

Therefore, if consider a pure hedonic game in which preferences over agents are lifted into preferences over coalitions through $M_{i,j}$ values we can use the following:

$$M_{i,j}\Big|_g = \varkappa_i \cdot \frac{1}{\text{dist}_g(i, j)}$$

where $\varkappa_i$ is some constant related to agent i. Due to the fact that the distances between two agents is given by the shortest path (supposing that the graph g is not weighted, or weighted with positive values), if two agents are close friends, there is a "small" distance between them, and thus, since we are working with ASHG, $\mathscr{B}/\mathscr{W}$−HGs and FHGs, we need the reciprocal of the distance. The constant $\varkappa_i$ is simply a scaling factor.

Now in a hedonic game in partition function form we exploit the distance between an agent and a coalition to set the $\mathcal{M}_{i,j}$ values as follows:

$$\mathcal{M}_{i,j}(\pi \big| g) = \varkappa_i \cdot \frac{1}{\text{dist}_g(i,j)} + \lambda_i \cdot \sum_{S \in \pi} \text{dist}_g(i, S) + \xi_i \cdot \sum_{S \in \pi} \frac{1}{\text{dist}_g(j, S)}$$

with $\varkappa_i$, λ_i and ξ_i be constants related to i. As expected, the composition of $\mathcal{M}_{i,j}$ for a game in PFF is more complex as it takes into consideration all coalitions' constitution within a partition π. In particular, we consider three 'properties' of contribution that influence a value $\mathcal{M}_{i,j}$ within a specific partition π under a specific social graph g:

1. the distance between agents i and j;
2. the distances between agent i and all coalitions within π; and
3. the distances between agent j and all coalitions within π.

Intuitively, the first property of contribution represents the 'direct' relationship that agent i develops towards agent j. The second property captures i's 'cost' to move to a different coalition, and through term $\lambda_i \cdot \sum_{S \in \pi} \text{dist}_g(i, S)$ agent i attributes to j a small/large amount for every coalition close/faraway to her. That is consider the partition $\pi = \Big\{\{1,2,5,9\}, \{3,4,7\}, \{6,8\}\Big\}$ and the social graph g depicted in Figure 4.1. The cost of agent 1 moving to another existing coalition is:

$$\text{dist}_g(1, \{3,4,7\}) + \text{dist}_g(1, \{6,8\}) = \frac{\text{dist}_g(1,3) + \text{dist}_g(1,4) + \text{dist}_g(1,7)}{3} + \frac{\text{dist}_g(1,6) + \text{dist}_g(1,8)}{2} = \frac{20}{3}$$

Therefore, agent 1 attributes to each agent $j \in \{2, 5, 9\}$ an amount of $\lambda_1 \cdot \frac{20}{3}$.

Similarly, the third property captures j's 'cost' to move to a different coalition, and through term $\xi_i \cdot \sum_{S \in \pi} \frac{1}{\text{dist}_g(j,S)}$ agent i expresses a degree of honour towards agent j for collaborating with her. In the above partition π, the cost to move of agent 9 is

$$\text{dist}_g(9, \{3,4,7\}) + \text{dist}_g(9, \{6,8\}) = \frac{\text{dist}_g(9,3) + \text{dist}_g(9,4) + \text{dist}_g(9,7)}{3} + \frac{\text{dist}_g(9,6) + \text{dist}_g(9,8)}{2} = \frac{14}{3}$$

thus, since the cost to move of agent 9 is relatively small, agent 1 'honours' agent 9 for their current collaboration with an amount: $\xi_1 \cdot \left(\frac{1}{\text{dist}_g(9,\{3,4,7\})} + \frac{1}{\text{dist}_g(9,\{6,8\})}\right) = \xi_1 \cdot \frac{14}{15}$.

As a result the value b_1^9 in the partition π under the graph g is:

$$b_1^9(\pi|g) = \varkappa_1 + \frac{20}{3} \cdot \lambda_1 + \frac{14}{15} \cdot \xi_1$$

4.1.2 Logical formulae generation

In the Boolean Hedonic Games the preference relations are defined through logic formulae ϕ and γ; as we have already explicitly shown in Section 2.1.3 for the simple Boolean Hedonic Games model, and in Section 3.3.3 for the Boolean Hedonic Games in Partition Function Form model. In the experimental process we follow in this thesis in order to verify the effectiveness of the various learning models in the area of hedonic preference relations within the scope of hedonic games, we placed no assumptions or restrictions on the model of dichotomous preferences (opposed to the graph restriction we used for the ASHGs, $\mathscr{B}/\mathscr{W}$/FHGs).

Therefore, the construction of the logic formulae was based on uniformly selection of the agents in the 'Appealing Partners' and the 'Repellent Partners', in the case of classic boolean hedonic games; and, respectively, uniformly selection of the patterns in the 'Appealing Collaborations' and the 'Repellent Collaborations', in the case of boolean hedonic games in patterns function form. What is to be noted here, is that even if we have no restrictions on the formulae, we carefully constructed each one so that every formula is feasible, i.e., there are no conflicts between the sets 'Appealing Partners'-'Repellent Partners' or the sets 'Appealing Collaborations'-'Repellent Collaborations' that would result an infeasible formula.

4.2 The QP and OPA evaluation metrics

The learning models we use in this thesis, as we have described them in Section 2.3, and specifically the supervised ones, are tools for function approximation. That is, given some points in the form of observations we build a function such that if we come upon a previously unseen observation we will be able to determine a value that represents good enough the feature our model attempts to learn. In hedonic games however, we deal with a particular property that distinguish out problem from the classical function approximation. That is, in hedonic games we care little about the actual coalitions' utilities, since our interest lies mainly on the preference relations formed. In other words, working with any class of hedonic games presented so far, if we approximately learn a function $\hat{u}_i$ for some agent i, we have to answer the following two questions:

1. Is the approximation $\hat{u}_i$ close to the real u_i? *and*

2. Does the preference relation encoded by $\hat{u}_i$ match the one encoded by u_i?

The first question actually focus on the learning models' performance under the mathematical function approximation point of view. The second, however, implies that even if our model is not the most accurate function approximator, yet it can actually provide us with a good enough preference relation. To clarify this, think of the following: for agent i the preference relation between coalitions $S, T \in N_i$ is $S \succ_i T$ that derives from the values $u_i(S) = \sum_{j \in S} M_{i,j} = 10$ and $u_i(S) = \sum_{k \in T} b_i^k = 5$. Now, our learning model approximately learn a function $\hat{u}_i$ as $\hat{u}_i(S) = 25$ and $\hat{u}_i(T) = 19$; it has actually 'failed' to accurately approximate u_i, however since $S \succ_i T$ if and only if $u_i(S) > u_i(T)$, our approximation $\hat{u}_i$ correctly encodes this relation since $\hat{u}_i(S) > \hat{u}_i(T)$.

Consequently, we distinguish the following two evaluation metrics:

- the *Root Mean Square Error* (RMSE) between the predicted function $\hat{u}_i(S, CS)$, and the true function $u_i(S, CS)$

- the *Qualitative Proximity* (QP) between the ordering described by $\hat{u}_i(S, CS)$ and the true preference relation.

The first metric is straightforward, $\text{RMSE}(u_i, \hat{u}_i) = \sqrt{\frac{1}{|D|} \sum_{(S,CS) \in D} \left(u_i(S, CS) - \hat{u}_i(S, CS)\right)^2}$, where D is the collection of testing data used in the model, and $|D|$ is the size of the collection D. The second metric, however, is more interesting. This metric indicates that even if the predicted $\hat{u}_i$ differs significantly from the true u_i in actual values, they are still *equivalent*. That is, functions $\hat{u}_i$ and u_i encode in the same way the preference relation between any two coalitions S and T, e.g. both $S \succsim_i^{u_i} T$ and $S \succsim_i^{\hat{u}_i} T$ hold. QP, in fact, could be thought of as a variant of *Kendall Tau* metric [Kendall, 1948].

Therefore, when a function u_i of an agent i is "learned" based on the training data, we extract a preference relation, $\succsim_i^{\hat{u}_i}$. Then we can measure the *percentage of equivalence* between u_i and $\hat{u}_i$ by counting the average of pairwise relations that are identically encoded by u_i and $\hat{u}_i$.

Thus, we define Qualitative Proximity (QP) as follows:

$$\mathrm{QP}(u_i, \hat{u}_i) = \frac{\sum_{(S,CS),(T,CS')\in D} \mathrm{CHK}\big(u_i, \hat{u}_i, (S, CS), (T, CS')\big)}{|D|(|D|-1)/2}, \text{ where}$$

$$\mathrm{CHK}\big(u_i, \hat{u}_i, (S, CS), (T, CS')\big) = \begin{cases} 1 & \text{if } \big((S, CS) \succsim_i^{u_i} (T, CS') \\ & \quad \wedge (S, CS) \succsim_i^{\hat{u}_i} (T, CS')\big) \\ 0 & \text{otherwise} \end{cases}.$$

One step further, we introduce a single metric that combines both RMSE and QP. According to this metric:

- the lower the error $\mathrm{RMSE}(u_i, \hat{u}_i)$, the better $\hat{u}_i$ fits u_i
- the higher the $\mathrm{QP}(u_i, \hat{u}_i)$, the better $\succsim_i^{\hat{u}_i}$ matches $\succsim_i^{u_i}$.

Thus, our Overall Preference Accuracy (OPA) metric can be defined as follows:

$$\mathrm{OPA}(u_i, \hat{u}_i) = \frac{\mathrm{QP}(u_i, \hat{u}_i)}{\varepsilon + \mathrm{RMSE}_{\mathrm{nrm}}(u_i, \hat{u}_i)},$$

where $\mathrm{RMSE}_{\mathrm{nrm}}$ is the normalized RMSE regarding the test samples values' range. This normalized RMSE allows us to compare the performance in games with different value ranges. The intuitive interpretation of the OPA metric is that, we highly value the contribution of QP metric–since this actually measures the similarity of the original preference relation to the approximated one–but also take into some consideration the RMSE between the original and the approximated function. In order to avoid division with zero, we add a small positive $\varepsilon = 10^{-5}$ to the denominator.

4.3 Linear Regression Model

In this section we will present the implementation of the *Linear Regression Model* in the problem of learning preferences in both *Classic Hedonic Games* and *Hedonic Games in Partition Function From*.

We remind the reader that an LRM computes a weight vector w based on the input data, and according to this weight vector it builds a function of the form:

$$y(\mathrm{x}_k) = w_0 + \sum_{i=1}^{b} w_i \cdot x_{k,i}.$$

Having computed the function y, the LRM is in place to give an answer for any observation in the form of x_k. Next, we will present the nature of the observations x_k used in any case, and the intuition of their selection.

4.3.1 LRM for Classic Hedonic Games

We employed the linear regression model in classic hedonic games that follow the model of Additively Separable, $\mathcal{B}/\mathcal{W}-$, and Fractional Hedonic Games.

As already mentioned in Section 2.3.1, the input of the LRM consist of pairs $\langle \mathrm{x}_k, t_k \rangle$, where x_k is the k^{th} observation and t_k is the target value, i.e., the actual value, that corresponds to observation x_k. In the problem of learning preferences within classic HGs, the observation x_k represents a coalition $S_k \subseteq N$, while the target value t_k represents the value *(a)* $u_i(S_k) = \sum_{j \in S_k} \mathcal{M}_{i,j}$, *(b)* $u_i(S_k) = \max_{j \in S_k} \mathcal{M}_{i,j}$, *(c)* $u_i(S_k) = \min_{j \in S_k} \mathcal{M}_{i,j}$, or *(d)* $u_i(S_k) = \mathrm{avg}_{j \in S_k} \mathcal{M}_{i,j}$, for *(a)* ASHGs, *(b)* $\mathcal{B}-$HGs, *(c)* $\mathcal{W}-$HGs, or *(d)* FHGs, respectively.

For now on let us focus on the LRM that agent 1 trains and maintain (denoted as LRM_1)– by altering agent 1 to agent i we will obtain the LRM_i that agent i owns. An observation x_k that is imported in the LRM_1 is an encoded coalition $S_k \in N_1$, i.e., agent $1 \in S_k$. The observation x_k encodes the coalition S_k as follows: we use n boolean variables, with each one indicating whether an agent is member of coalition S_k. That is,

$$\mathrm{x}_k = \begin{bmatrix} \mathbb{1}_{1 \in S_k}, & \mathbb{1}_{2 \in S_k}, & \mathbb{1}_{3 \in S_k}, & \cdots, & \mathbb{1}_{n \in S_k} \end{bmatrix}$$

where $\mathbb{1}_{i \in S_k}$ is the indicator function that answers if agent $i \in S_k$, i.e.,:

$$\mathbb{1}_{i \in S_k} = \begin{cases} 1, & \text{if } i \in S_k, \\ 0, & \text{otherwise} \end{cases}.$$

Clearly, since we are in the LRM_1 and $S_k \in N_1$, we always have $\mathbb{1}_{1 \in S_k} = 1$, and thus the observations is:

$$\mathrm{x}_k = \begin{bmatrix} 1, & \mathbb{1}_{2 \in S_k}, & \mathbb{1}_{3 \in S_k}, & \cdots, & \mathbb{1}_{n \in S_k} \end{bmatrix}.$$

n	$\|edges\|$	$\|edge_{weight}\|$
20	$U(19, 190)$	$U(1, 10)$
50	$U(49, 1225)$	$U(1, 10)$
100	$U(99, 4950)$	$U(1, 10)$

Table 4.1: Social graphs' specifications.

The corresponding target value t_k is one of the follow:

$$t_k = \begin{cases} \sum_{i \in S_k} \mathcal{M}_{1,i}, & \text{if game } G \text{ is ASHG} \\ \max_{i \in S_k} \mathcal{M}_{1,i}, & \text{if game } G \text{ is } \mathcal{B} - \text{HG} \\ \min_{i \in S_k} \mathcal{M}_{1,i}, & \text{if game } G \text{ is } \mathcal{W} - \text{HG} \\ \text{avg}_{i \in S_k} \mathcal{M}_{1,i}, & \text{if game } G \text{ is FHG} \end{cases}$$

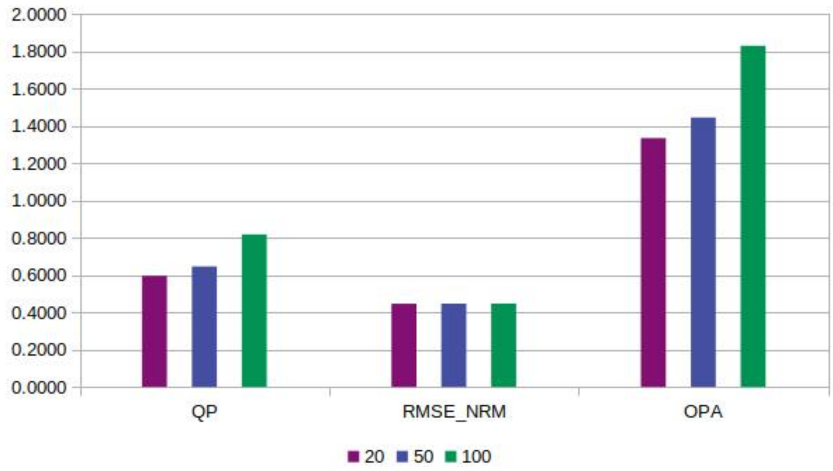
Experimental Results

We conducted a number of experimental simulations in order to evaluate each learning model. Specifically, in each simulation we ran 5 game 'instances' for each class of games, with each game instance in these games be defined by a social graph. So we generated 5 different social graphs for each game class:

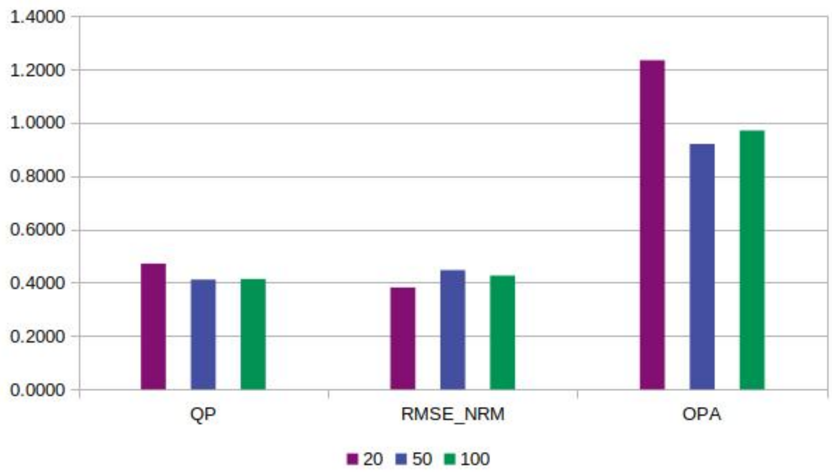
- Additively Section HGs;
- $\mathcal{B}-$HGs; and
- $\mathcal{W}-$HGs.

Moreover, we evaluated the scalability of LRM in the problem of learning preferences as the number of agents increase in the game. In words, we trained and test LRMs for $n = 20, 50$ and 100 agents, in Table 4.1 you can see the specifications of each social graph; here the constant $\varkappa_i$ is set to 1. In Table 4.2 we show the average values over the 5 games of the QP, RMSE, and OPA metrics per hedonic game class and per number of agents. While Figure 4.2 is the graphical representation of the above table.

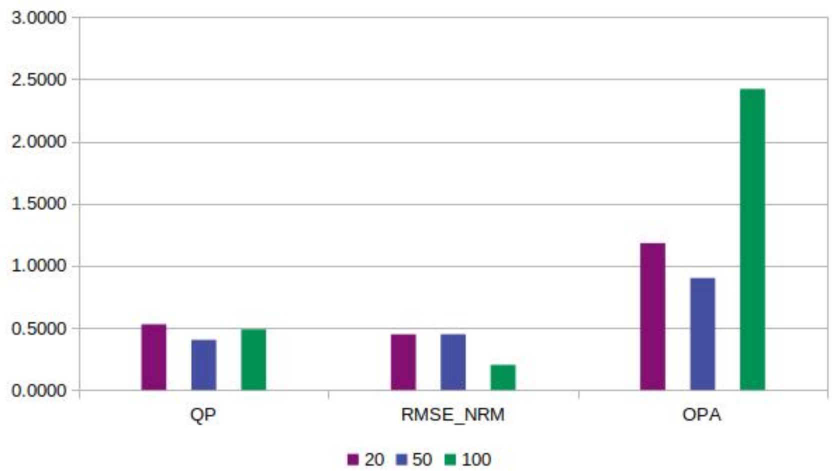
As we can see, the LRMs used for the ASHGs have a better performance than the ones used for $\mathcal{B}-$HGs and $\mathcal{W}-$HGs. This in fact, is an anticipating result, as ASHGs are based on summation which is linear. $\mathcal{B}-$HGs and $\mathcal{W}-$HGs, on the other hand, are based on maximization



(a) Classic ASHGs



(b) Classic $\mathscr{B}$−HGs



(c) Classic $\mathscr{W}$−HGs

Figure 4.2: Linear Regression Model: average on QP, RMSE_{nrm}, and OPA metrics over 5 game-instances per class of Hedonic Games.

n	Number of Samples		ASHG			$\mathscr{B}$−HG			$\mathscr{W}$−HG		
	Training	Testing	RMSE	QP	OPA	RMSE	QP	OPA	RMSE	QP	OPA
20	400	900	0.2683	0.5953	1.3345	$26.366 \cdot 10^3$	0.4715	1.2342	$18.275 \cdot 10^3$	0.5282	1.1811
50	800	1200	0.2913	0.6456	1.4448	$14.752 \cdot 10^3$	0.4115	0.9200	$14.793 \cdot 10^3$	0.4030	0.9001
100	1500	2500	0.2987	0.8132	1.8293	$8.726 \cdot 10^3$	0.4138	10.9707	$2.02753 \cdot 10^6$	0.4861	2.4209

Table 4.2: Experimental results for LRM in classic hedonic games.

n	Total Space	Training Samples (%)	Testing Samples (%)
20	524, 288	0.076%	0.1716%
50	562, 949, 953, 421, 312	$1.421 \cdot 10^{-10}$%	$2.131 \cdot 10^{-10}$%
100	$6.338253001141147 \cdot 10^{29}$	$2.366 \cdot 10^{-25}$%	$3.944 \cdot 10^{-25}$%

Table 4.3: Number of samples as a proportion of the total space.

and minimization, respectively, which are non-linear. Also, notice that we fed each LRM with an very small portion of the total space of the function to be learnt, as the reader can see in Table 4.3.

4.3.2 LRM for Hedonic Games in Partition Function Form

In this section we evaluate the effectiveness of linear regression models on the problem of hedonic games in partition function form. Specifically, we used LRMs to learn preferences over partitions for ASHG-PFF and Boolean Hedonic Games in PFF. Here each observation x_k corresponds to an embedded coalition. In order to encode an embedded coalitions, i.e., a coalition structure, as an observation we use the following structure: we exploit $\binom{n}{2}$ boolean variables, with each one indicating whether an unordered pair of agents co-exist in the same coalition within the given partition. That is, for each unordered pair (i, j) we use the indicator function:

$$\mathbb{1}_{i \in CS(j)} = \begin{cases} 1, & \text{if agent } i \text{ is in coalition } CS(j), \\ 0, & \text{otherwise} \end{cases}.$$

Thus, a coalition structure CS_k is encoded as x_k having $\binom{n}{2}$ elements as:

$$\begin{aligned}\mathrm{x_k} = [\ &\underbrace{\mathbb{1}_{2\in CS(1)}, \mathbb{1}_{3\in CS(1)}, \cdots, \mathbb{1}_{n\in CS(1)}}_{n-1 \text{ elements}}, \underbrace{\mathbb{1}_{3\in CS(2)}, \mathbb{1}_{4\in CS(2)}, \cdots, \mathbb{1}_{n\in CS(2)}}_{n-2 \text{ elements}},\\ &\underbrace{\mathbb{1}_{4\in CS(3)}, \mathbb{1}_{5\in CS(3)}, \cdots, \mathbb{1}_{n\in CS(3)}}_{n-3 \text{ elements}}, \qquad \cdots\cdots\cdots\cdots,\\ &\underbrace{\mathbb{1}_{i+1\in CS(i)}, \mathbb{1}_{i+2\in CS(i)}, \cdots, \mathbb{1}_{n\in CS(i)}}_{n-i \text{ elements}}, \qquad \cdots\cdots\cdots\cdots,\\ &\underbrace{\mathbb{1}_{n-1\in CS(n-2)}, \mathbb{1}_{n\in CS(n-2)}}_{2 \text{ elements}}, \mathbb{1}_{n\in CS(n-1)}]\end{aligned}$$

Note that regardless which agent's LRM we think of, the observation remains the same. That is, x_k carries the information about the whole partition, and not about a single coalitions. Thus, no matter if we are in LRM_1 or LRM_2 or LRM_i, all models will receive as an input the very same x_k, *but* each model sees a different target value which depends on the corresponding agent. As such, the corresponding target value t_k for some specific observation x_k which encodes the embedded coalition $\langle S, CS_k\rangle$ for agent 1 in LRM_1 is:

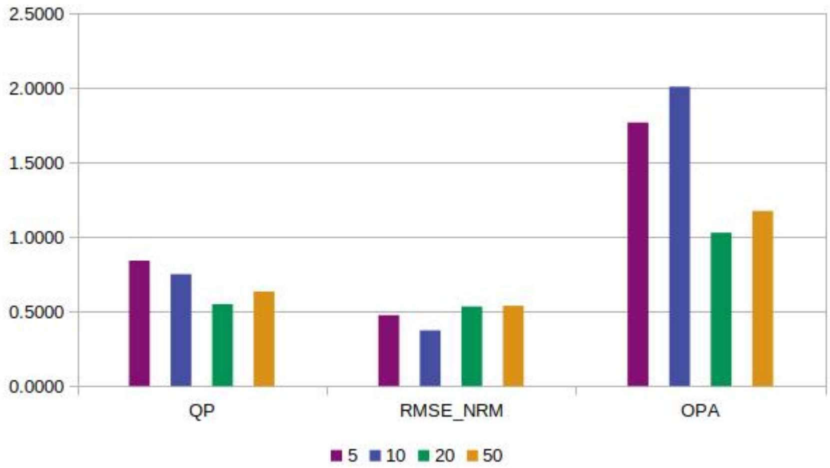
$$t_k = \begin{cases} \sum_{i\in S} \mathcal{M}_{1,i}(CS), & \text{if game } G \text{ is ASHG} \\ +c, & \text{if game } G \text{ is BHG and } CS_k \in P_1^+ \\ -c, & \text{if game } G \text{ is BHG and } CS_k \in P_1^- \end{cases}$$

where c is some positive constant.

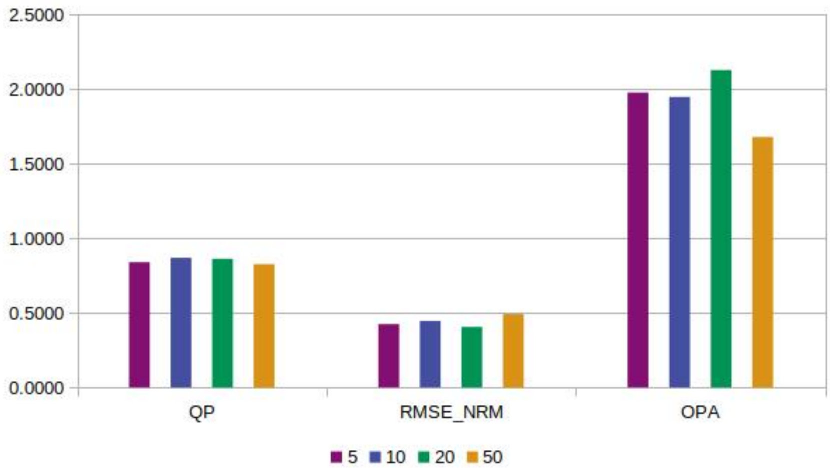
Experimental Results

Here we generated 5 game instance of Additively Separable HGs and 5 game instance of Boolean HGs. For the ASHGs each game instance is defined by the social graph, while each Boolean HGs' game instance is defined through the logic formulae γ. In Table 4.4, the reader can see the specifications for each social graph, and each logic formulae.

We trained and tested LRMs for $n = 5, 10, 20$ and 50 agents to evaluate the models' performance as the number of agents increase. In Table 4.5 we show the average values over the 5



(a) BHGs in PFF



(b) ASHGs in PFF

Figure 4.3: Linear Regression Model: average on QP, $RMSE_{nrm}$, and OPA metrics over 5 game-instances per class of Hedonic Games in PFF.

n	Graph			Formulae	
	$\varkappa, \lambda, \xi$	$\lvert edges \rvert$	$edge_{weight}$	$\#\langle Incl_i, \overline{Incl_i}\rangle / agent$	$\#agents / \langle Incl_i, \overline{Incl_i}\rangle$
5	$\sim U(0, 10)$	$\sim U(4, 10)$	$U(1, 5)$	$U(2, 3)$	$U(1, 4)$
10	$\sim U(0, 10)$	$\sim U(9, 45)$	$U(1, 5)$	$U(4, 7)$	$U(4, 6)$
20	$\sim U(0, 10)$	$\sim U(19, 190)$	$U(1, 5)$	$U(9, 11)$	$U(5, 7)$
50	$\sim U(0, 10)$	$\sim U(49, 1225)$	$U(1, 5)$	$U(12, 15)$	$U(8, 12)$

Table 4.4: Social graph's, and logic formulae specifications.

n	Number of Samples		BHG			ASHG		
	Training	Testing	RMSE	QP	OPA	RMSE	QP	OPA
5	200	500	0.5641	0.8416	3.2598	39.789	0.8390	3.7842
10	2000	5000	0.5878	0.7510	3.3687	78.336	0.8685	2.9331
20	5000	10000	1.0283	0.5496	1.8440	165.57	0.8619	3.1564
50	20000	40000	0.9372	0.6336	2.1193	696.89	0.8258	2.7457

Table 4.5: Experimental results for LRM in hedonic games in partition function form. The values are average on QP, RMSE, and OPA metrics over 5 Hedonic Games in PFF.

n	Total Space	Training Samples (%)	Testing Samples (%)
5	$10^{1.71600}$	384.6%	1346.16%
10	$10^{5.06437}$	1.724%	$6.04 \cdot 10^{-10}$%
20	$10^{13.71372}$	$9.66 \cdot 10^{-9}$%	$1.16 \cdot 10^{-9}$%
50	$10^{47.26897}$	$1.07 \cdot 10^{-41}$%	$2.68 \cdot 10^{-41}$%

Table 4.6: Number of samples as a proportion of the total space.

game instances of the QP, RMSE, and OPA metrics per hedonic game class and per number of agents. While Figure 4.3 is the graphical representation of the above table.

Again, the LRMs used for the ASHGs have a better performance than the ones used Boolean HGs; which is an anticipating result, since learning preference for Boolean HGs can be thought of as a classification problem. Therefore, for Boolean HGs a learning model that assume the input data to be linear is no very accurate. Nonetheless, if we consider the portion of the total space of the functions to be learnt, the performance of the models are more than antiquate (see in Table 4.6). That is, apart form the environment with 5 agents, in all the other settings ($n = 10, 20, 50$) the LRM sees an extremely small proportion of a vast space.

4.4 Regression with Basis Functions

Here we discuss the linear regression model with basis function used for evaluating the model's performance on the problem of learning preferences in classic hedonic games and in hedonic games in partition function form.

As we have discussed in Section 2.3.1 in the regression model with basis functions, we use $\mathcal{M}$ basis functions, and every basis function is of the form:

$$\phi_i(\mathrm{x}_k) = e^{\left\{-\frac{||\mathrm{x}_k-\mu_i||}{2\sigma_i^2}\right\}},$$

where μ_i reflects the location of each ϕ_i in the L-dimensional space, and σ the scale of each value, which is common for every ϕ_i. To clarify this, with L in the aforementioned dimensional space we refer to the number of different features that an observation x_k has. That is, the number of elements in each observation corresponds to a feature, and therefore L coincides with the size of the observation x_k. As a result μ_i is a vector with L elements that 'positions' function ϕ_i in the L-dimensional space. In order to compute the center vector μ_i of each ϕ_i, we use the well-known *k-means* algorithm [MacQueen et al., 1967] K-Means is an unsupervised learning method that partitions data into a specific number of clusters, and calculates the center values μ_i for every cluster. The scale of each value σ is same for all basis functions ϕ_i, and is computed as:

$$\sigma = \frac{\max_{i,j} |\mu_i - \mu_j|}{\sqrt{2 \cdot (\mathcal{M} - 1)}}, \quad \forall\, i, j = 1, 2, \cdots, \mathcal{M}$$

The total number $\mathcal{M}$ of the basis functions depends essentially on the problem at hand; here we determine $\mathcal{M}$ with the TPE method described in Section 4.5. That is, we optimize the number of the basis functions in every learning model by taking into consideration the available input data, i.e, the training set of the RMBF.

4.4.1 RMBF for Classic Hedonic Games

Here we present the performance of regression models with basis functions for classic hedonic games; and in particular, for ASHGs, $\mathscr{B}-$HGs, and $\mathscr{W}-$HGs. Since we are working with classic hedonic games, the input observations x_k of an RMBF encodes a single coalition, and follows the representation described in Section 4.3.1. That is, if S_k is the coalition that corresponds to

n	Number of Samples		ASHG			$\mathscr{B}$−HG			$\mathscr{W}$−HG		
	Training	Testing	RMSE	QP	OPA	RMSE	QP	OPA	RMSE	QP	OPA
20	400	900	0.4406	0.8021	1.8427	$16.042 \cdot 10^3$	0. − −	1.6593	$13.955 \cdot 10^3$	0.7545	1.6870
50	800	1200	0.3492	0.7105	1.7090	$14.219 \cdot 10^3$	0. − −	1.6755	$14.190 \cdot 10^3$	0.6725	1.4918
100	1500	2500	0.3554	0.8132	1.8770	$8.607 \cdot 10^3$	0. − −	1.6882	$10.037 \cdot 10^3$	0.7769	1.7371

Table 4.7: Experimental results for RMBF in classic hedonic games.

observation x_k, then the x_k has the form:

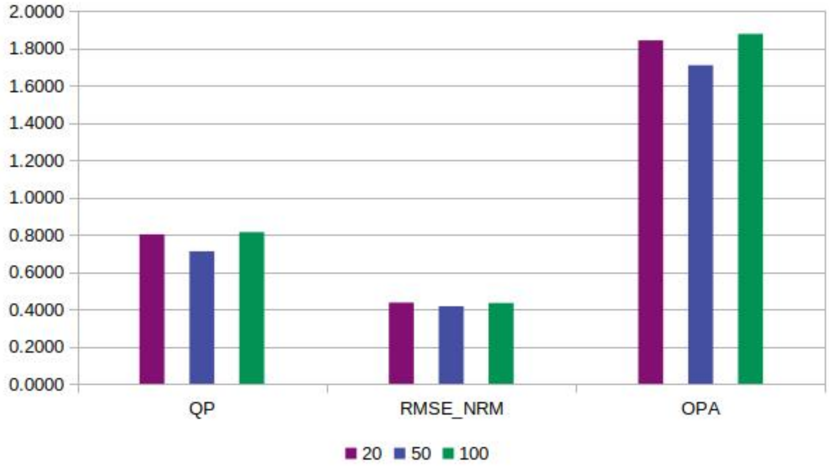
$$\mathrm{x}_k = \left[\mathbb{1}_{1\in S_k}, \quad \mathbb{1}_{2\in S_k}, \quad \mathbb{1}_{3\in S_k}, \quad \cdots, \quad \mathbb{1}_{n\in S_k}\right]$$

For each class of games (ASHGs, $\mathscr{B}$−HGs, and $\mathscr{W}$−HGs) we generated 5 game instances following the specifications in Table 4.1. Once again, we trained and tested the RMBFs for $n = 20, 50$ and 100 agents.

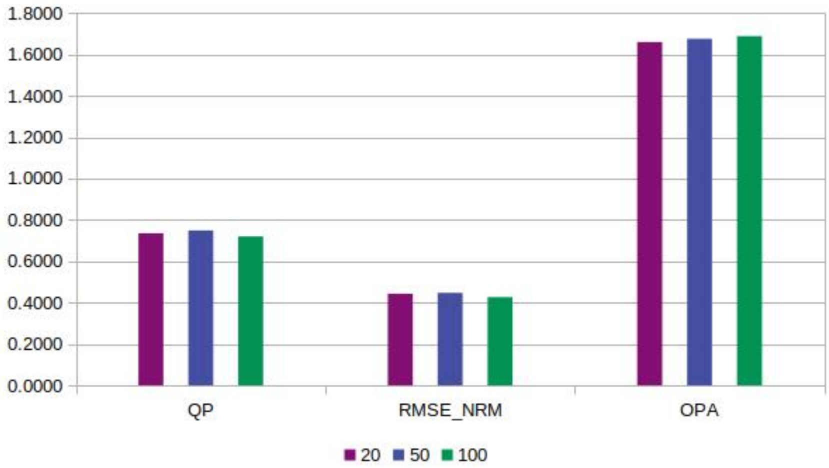
In Table 4.7 we show the average values over the 5 games of the QP, RMSE, and OPA metrics per hedonic game class and per number of agents; while Figure 4.4 is the graphical representation of this table. Looking at Table 4.7, we can observe that in all classes of games we achieve a value of *QP* metric that is above 0.7. This confirms that by using the regression model with non-linear basis function we approximate more accurately non-linear functions, such as the one representing $\mathscr{B}$−HGs and $\mathscr{W}$−HGs; and in the same time gain performance and in the function representing ASHGs. The latter, does not arises any questions due to the nature of the function we have used so far in order to describe an ASHG; that is, each coalition is mapped to a value that is summation over non linear terms:

$$v_i^{\mathrm{ASHG}}(S) = \sum_{j\in S} \frac{1}{\mathrm{dist}_g(i,j)}.$$

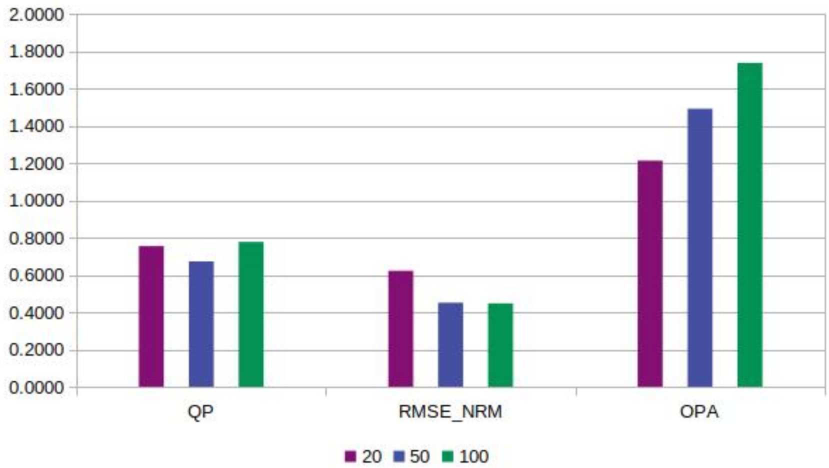
Last but not least, in Figure 4.5 we show the graphical representation of OPA metric per game class and per number of agents for both LRM and RMBF learning models. In order to do so, we used the same training sets and the same testing sets in both types of learning models. As a result, we can observe that the Regression Model with Basis Functions *outperforms* the Linear Regression Model in all game-classes.



(a) classic ASHGs

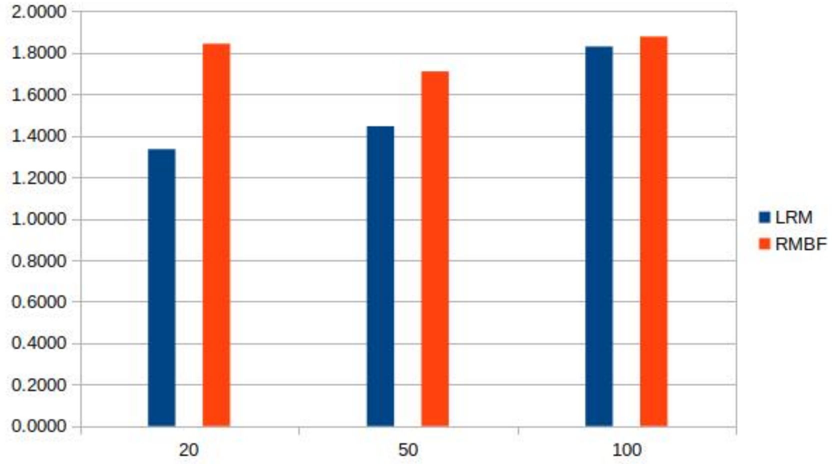


(b) classic $\mathscr{B}$−HGs

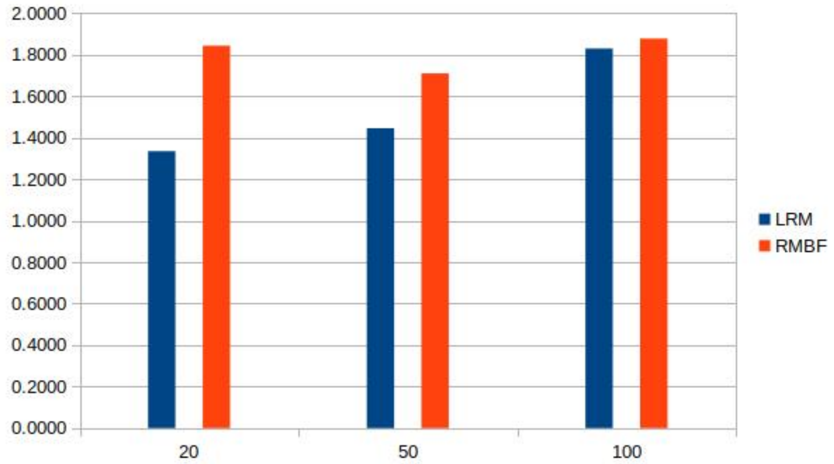


(c) classic $\mathscr{W}$−HGs

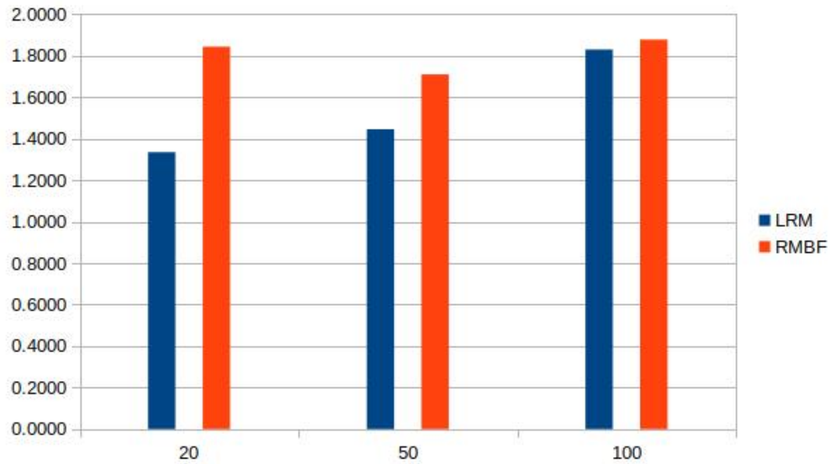
Figure 4.4: Regression Model with Basis Functions: average on metric OPA over 5 classic game-instances per class of Hedonic Games.



(a) classic ASHGs



(b) classic $\mathscr{B}$−HGs



(c) classic $\mathscr{W}$−HGs

Figure 4.5: Performance of LRM vs RMBF in classic HGs according to the OPA metric. The values are average over 5 game-instances per class of Hedonic Games.

n	Number of Samples		BHG			ASHG		
	Training	Testing	RMSE	QP	OPA	RMSE	QP	OPA
5	200	500	0.4072	0.9059	3.8592	30.347	0.8926	4.2244
10	2000	5000	0.8593	0.6887	2.833	106.60	0.8071	2.9914
20	5000	10000	0.7816	0.7348	2.4577	242.29	0.7443	2.5969
50	20000	40000	0.6561	0.8035	2.7041	885.76	0.7174	2.4304

Table 4.8: Experimental results for RMBF in hedonic games in partition function form. The values are average on metrics RMSE, QP and OPA over 5 Hedonic Games in PFF.

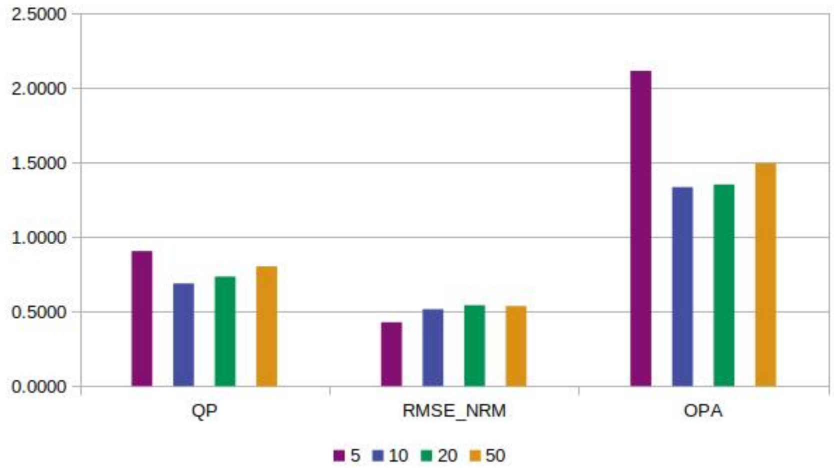
4.4.2 RMBF for Hedonic Games in PFF

Here we evaluate the performance of regression models with basis functions for hedonic games in partition function form; and in particular, for ASHGs-PFF, and Boolean HGs-PFF. Since we are working with hedonic games in PFF, the input observations x_k of an RMBF encodes a single coalition, and follows the representation described in Section 4.3.2. That is, if $\langle S_k, CS_k \rangle$ is the embedded coalition that corresponds to observation x_k, then the x_k has the form:

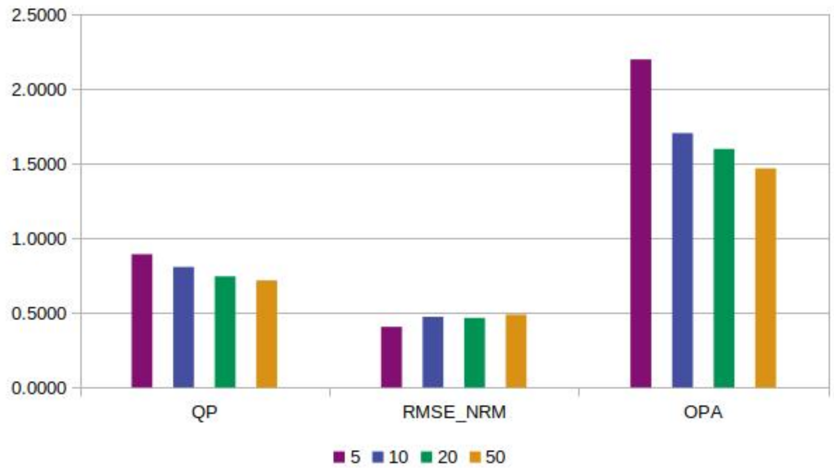
$$
\begin{aligned}
\mathrm{x_k} = [\ & \underbrace{\mathbb{1}_{2 \in CS(1)}, \mathbb{1}_{3 \in CS(1)}, \cdots, \mathbb{1}_{n \in CS(1)}}_{n-1 \text{ elements}}, \underbrace{\mathbb{1}_{3 \in CS(2)}, \mathbb{1}_{4 \in CS(2)}, \cdots, \mathbb{1}_{n \in CS(2)}}_{n-2 \text{ elements}}, \\
& \underbrace{\mathbb{1}_{4 \in CS(3)}, \mathbb{1}_{5 \in CS(3)}, \cdots, \mathbb{1}_{n \in CS(3)}}_{n-3 \text{ elements}}, \qquad \cdots\cdots\cdots\cdots, \\
& \underbrace{\mathbb{1}_{i+1 \in CS(i)}, \mathbb{1}_{i+2 \in CS(i)}, \cdots, \mathbb{1}_{n \in CS(i)}}_{n-i \text{ elements}}, \qquad \cdots\cdots\cdots\cdots, \\
& \underbrace{\mathbb{1}_{n-1 \in CS(n-2)}, \mathbb{1}_{n \in CS(n-2)}}_{2 \text{ elements}}, \mathbb{1}_{n \in CS(n-1)}]
\end{aligned}
$$

For each class of games (ASHGs-PFF, BHGs-PFF) we generated 5 game instances following the specifications in Table 4.4. Once again, we trained and tested the RMBFs for $n = 5, 10, 20$ and 50 agents.

In Table 4.8 we show the average values over the 5 games of the QP, RMSE, and OPA metrics per hedonic game class and per number of agents; while Figure 4.6 is the graphical representation of this table. Looking at Table 4.8, we can observe that Regression Model with Gaussian basis function can approximate Boolean HGs in PFF achieving accuracy (regarding to QP



(a) BHGs in PFF



(b) ASHGs in PFF

Figure 4.6: Regression Model with Basis Functions: average on metric OPA over 5 classic game-instances per class Hedonic Games in PFF.

	5			10			20			50		
	LRM	LRMRBF	NN	LRM	LRMRBF	NN	LRM	LRMRBF	NN	LRM	LRMRBF	NN
ASHGs-PFF	0.05sec	1.2sec	14sec	2.4sec	17sec	1.7min	10sec	1.7min	8.3min	3.2min	25.3min	5.5hr
BHGs-PFF	0.04sec	1sec	45sec	1.3sec	12sec	2.9min	5.5sec	1.2min	8.4min	1.7min	30.6min	2.5hr

Table 4.9: Approximate time needed per game for training & testing.

metric) up to 0.8 for 50 agents. In the same time, the RMBFs achieve a very good Qualitative Proximity (QP metric) in ASHGs-PFF as well (above 0.7).

4.5 Feed Forward Neural Networks

The model of neural networks is a powerful, yet complex learning tool. Neural networks, even in their simplest form, require a significant amount of computational power and they are time consuming; indicatively in Table 4.9 we show the average time required to train and test LRMs, RMBFs and FFNNs for hedonic games in partition function form. For this reason, we used the model of *Feed Forward Neural Network* only for learning preferences in hedonic games in partition function form; since such games are by nature more complex than classic hedonic games, and the preferences in HGs-PFF lies in a vast space even for a small number of agents: for instance, for 10 agents we have 115975 elements for which elements an agents defines a preference relation per unordered pair. Specifically, we trained and tested FFNNs for Adaptive Separable Hedonic Games in PFF and for Boolean Hedonic Games in PFF.

We remind the reader that in FFNN we have several hidden layers, and in each hidden layer, a non-linear activator is used, otherwise it would act similarly to an simple LR model. Depending on the PFF-HG class, we let the output layer have a different activator. That is, in ASHGs-PFF, a regression model is needed to approximate the function $t_k = v_i(S, CS_k) \in \mathbb{R}$, so the output layer activator must be linear. In the case of BHGs-PFF, essentially we have a classification problem, thus we use a sigmoid activator at the output layer. By using a sigmoid function, the resulting target values lie in $\{0, 1\}$; thus we use the convention that when the target value of a given sample CS_k is $t_k = 1$, it means that $CS_k \in P_i^+$, and when $t_k = 0$ we have that $CS_k \in P_i^-$.

We have already mentioned that the performance of neural networks depends on the choice of some hyperparameters such as the optimization method, the number of nodes per layer, the activator function in each layer, etc.; while the chosen set of hyperparameters is highly dependent on the complexity of the to-be-learned preferences, i.e., the problem at hand. In this work, we use as an optimization method the *ADAM optimization* algorithm [Ba and Kingma, 2015]. ADAM attempts to combine the benefits of two different variations of the stochastic gradient descent method, namely Adaptive Gradient Descent (AdaGrad) and the Root Mean Square Propagation (RMSprop) [Ruder, 2016]. AdaGrad uses a different learning rate for each parameter to improve performance for sparse gradients, while in RMSprop and AdaDelta the learning rate of each parameter relies on a decaying average of recent gradients. The choice of the ADAM optimizer was made empirically, following a series of experiments with instances of our problem. This showed that ADAM outperformed AdaGrad, RMSprop and AdaDelta [Ruder, 2016]. We focused on the above methods, since adaptive learning systems are generally suggested for sparse data, as discussed in [Ruder, 2016]. The sparsity of the data is clearly indicated by the form of our observations' representation, which is thoroughly described in Section 4.3.2.

Having selected the optimization method, we let our architecture self-tune other hyperparameters of the network by using the Tree-structured Parzen Estimator (TPE) [Bergstra et al., 2011], an algorithm based on Bayesian optimization with the use of *hyperopt library* for python [Bergstra et al., 2013]. In general, hyperparameter optimizers attempt to find a value h_i that minimizes a function $f(h)$, i.e. the optimizer seeks the value $\arg\min_h(f(h))$. In our apporach, f represents the loss function of the neural network, since this is the quantity we want to minimize. For a predefined number of steps, the algorithm produces sets of hyperparameters h_i, that are used to construct a neural network, which is then trained and tested given the input data. At the end of this process, our architecture yields the best set of hyperparameters h^*, along with the trained neural network that uses these h^*. The TPE allows us to optimize hyperparameters that are structurally dependent: for instance, if we set as a hyperparameter the number of hidden layers, we first optimize the number of layers, and then the hyperparameters for each layer. Figure 4.7 shows the architecture of the model for selecting the best hyperparameters for the neural network. In Section 4.2 we introduced the QP evaluation metric, which is used as the function f.

4.5.1 Experimental Results

In Table 4.10 we shows the average values over the 5 games of the QP, RMSE, and OPA metrics per hedonic game class and per number of agents. As we can see, NN models outperform both LRM and LRMRBF, both in ASHGs and BHGs, especially as the number of agents increases.

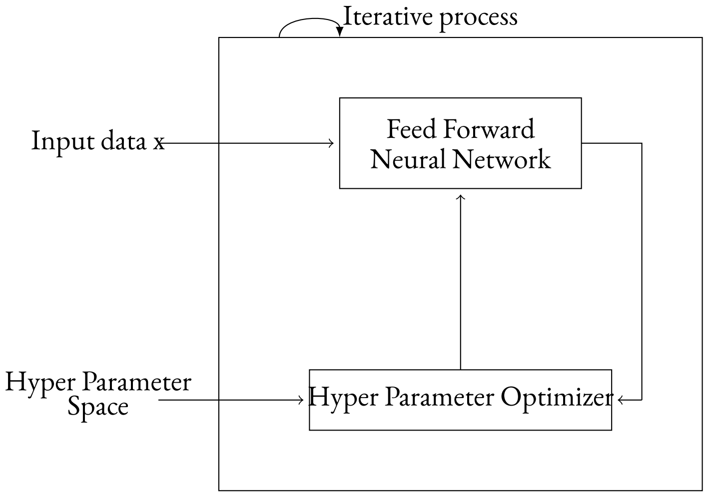


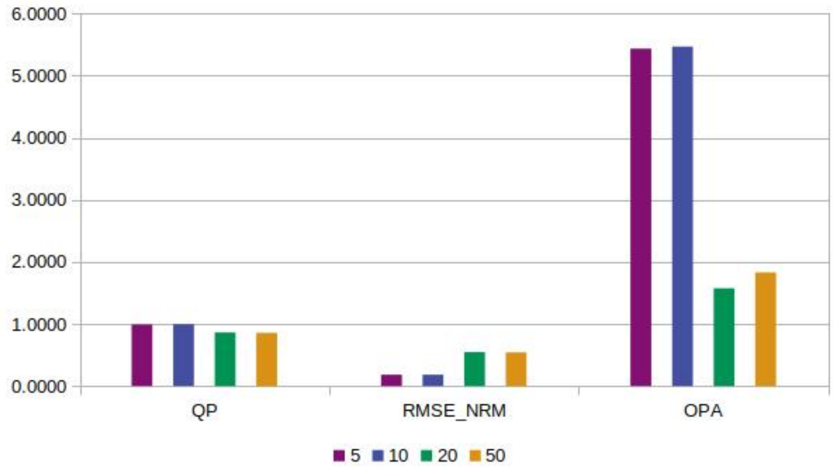
Figure 4.7: Architecture for selecting network hyperparameters.

n	Number of Samples		BHG			ASHG		
	Training	Testing	RMSE	QP	OPA	RMSE	QP	OPA
5	200	500	0.0283	0.9921	5.9525	13.887	0.9188	1.9925
10	2000	5000	0.0123	0.9985	5.9905	14.604	0.9678	2.1362
20	5000	10000	0.2784	0.8625	2.8668	71.354	0.9377	2.2991
50	20000	40000	0.2824	0.8559	3.3192	404.52	0.9032	1.8285

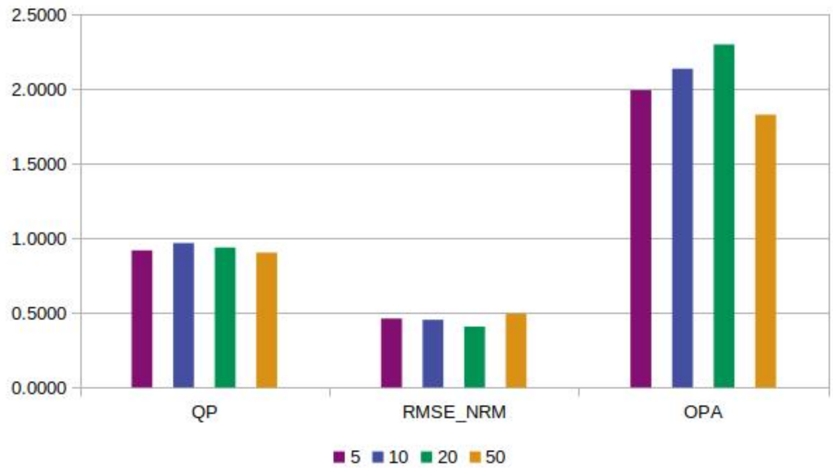
Table 4.10: Experimental results for FFNN in hedonic games in partition function form. The values are average on metrics RMSE, QP and OPA over 5 game instances per class of Hedonic Games in PFF.

This is, in fact, an anticipated result since the hyper-parameters optimization makes our NN models more adaptive to the problem. However, the larger the number of agents is, the more computationally expensive the NN model is. This results from the hidden layers having many fully connected nodes. The number of layers is 1 or 2, while the nodes per layer are in $[\frac{n}{2}, \frac{n\cdot(n-1)}{2}]$, selected by TPE optimizer.

In Figure 4.8 we depict the graphical representation of the QP, RMSE, and OPA described in Table 4.10. While in Figure 4.9 we show the juxtaposed values of OPA metric for all learning models used to approximate preferences of hedonic games in partition function form. Again, in order to do so, we used the same training sets and testing sets for all models per game-class.

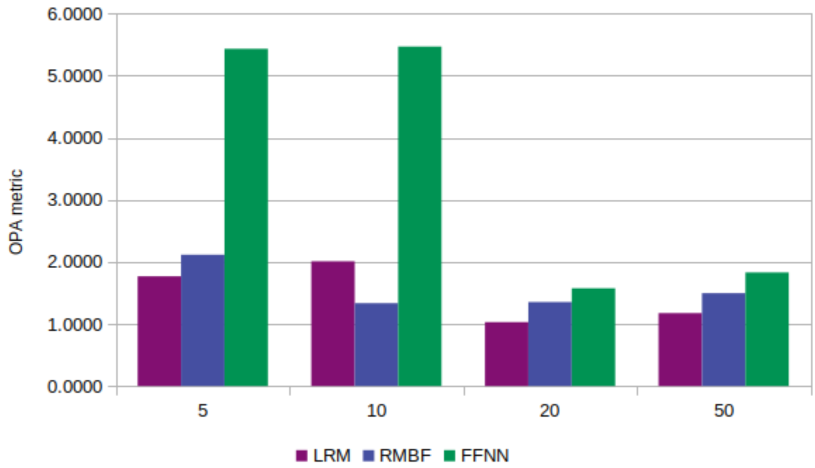


(a) BHGs in PFF

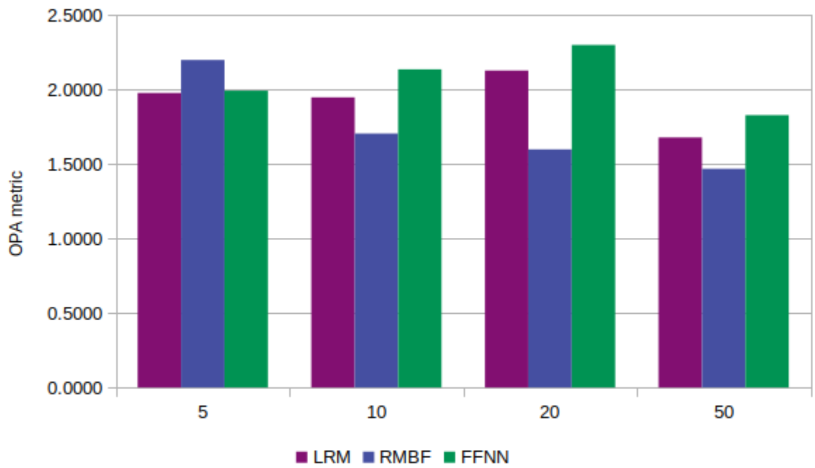


(b) ASHGs in PFF

Figure 4.8: Feed Forward Neural Networks: average on metric OPA over 5 Hedonic Games in PFF.



(a) BHGs in PFF



(b) ASHGs in PFF

Figure 4.9: Performance of LRM vs RMBF vs FFNN Hedonic Games in PFF according OPA metric. The values are average over 5 game-instances per class of Hedonic Games in PFF.

4.6 Probabilistic Topic Modelling

Now, let us move to the unsupervised learning technique we used in order to *discover* hidden collaboration patterns within hedonic games. Specifically, we employed Probabilistic Topic Modelling in classic Boolean Hedonic Games. To avoid any faugness we clarify that in this section we use the symbol ϕ to refer to logic formulae – opposed to previous sections in this chapter where ϕ was used to represent basis functions.

As we have already mentioned Probabilistic topic models (PTMs) is a statistical approach used in analyzing words of documents that was originally used in data mining to discover a distribution over topics related to a given text document. Firstly, in order to use PTMs so that we can extract hidden information regarding the preferences of the underlying hedonic game we need to prescribe a way to represent coalitions and preferences orderings into text documents. As such, right below in Section 4.6.1 we present a novel procedure(inspired by the recent work of [Mamakos and Chalkiadakis, 2018]) to interpret a pair of coalition and preference order into a 'bag-of-words', which can then be channeled into the PTM algorithm.

4.6.1 Game interpretation into documents

Let $G = \langle N, \phi_1, \cdots, \phi_n \rangle$ be a hedonic game with dichotomous preferences, where N is a set of players and $n = |N|$. ϕ_i represents a logic formula correlated to agent $i \in N$, which allows agent i to 'approve' or 'disapprove' a given coalition. The formula ϕ_i consists a concise representation of the preference relation $\succsim_i$ (the preference relation $\succsim_i$ may be of size exponential in n, as opposed to formula ϕ_i which may be significantly shorter). In hedonic games with dichotomous preferences, each agent classifies the coalitions related to her into satisfactory coalitions and dissatisfactory ones. Intuitively, formula ϕ_i expresses agent i's goal, and agent i is satisfied if her goal is achieved, or dissatisfied otherwise.

We define an instance $\varkappa$ of game G as a tuple $\langle CS_\varkappa, satisfied_1, \cdots, satisfied_n \rangle$, where $CS_\varkappa$ is a coalition structure of N, and $satisfied_i$ is a auxiliary boolean variable that indicates whether agents i's goal is achieved or not. We let each instance $\varkappa$ produce n documents, one document per agent. Since preferences are assumed to be exclusively personal, it is natural to consider n formulae to be independent. Under this assumption, it is natural to train and maintain n different LDA models each of which learns a single formula. Therefore, we assign a single model to each agent, which corresponds to the formula related to that agent; i.e. agent i is responsible for the i^{th} model, which is used to discover ϕ_i.

We sample m instances of the game G. Every instance produces n documents, where each document refers to exactly one's agent formula. Thus, in total we have $m \cdot n$ documents to train n different models. That is, each agent i uses in her own probabilistic topic model exactly m documents, which correspond to her own formula ϕ_i. Thus, the corpus of each model is of size $(1/n)$% of the total number of produced documents.

This approach is similar to the one used in [Mamakos and Chalkiadakis, 2018]. However, in the work of Mamakos and Chalkiadakis, agents belonging in the same coalition process identical documents describing this coalition; while in our case, agents belonging to the same coalition process *different* documents describing the same coalition. This divergences from the previous work makes our representation distinct, and is due to the nature of the games: the work of [Mamakos and Chalkiadakis, 2018] concentrates in transferable utility games, where a coalition is related to a single utility; by contrast, we work with hedonic games with dichotomous preferences, where each agent participating in a coalition S characterizes S as *satisfactory* or *disatisfactory*, regardless of the corresponding characterization by her partners.

A document $\mathrm{d}_{i,\varkappa}$ is related to agent i in instance $\varkappa$, and contains the following: an indicative word for each agent in coalition $CS_\varkappa(i)$, the indicative word 'gain' if the coalition $CS_\varkappa(i)$ is satisfactory (i.e. if $CS_\varkappa(i) \in N_i^+$), or the indicative word 'loss' if $CS_\varkappa(i)$ is disatisfactory ($CS_\varkappa(i) \in N_i^-$). For example, with $N = \{1, 2, 3, 4, 5\}$, and an instance

$$\mathrm{inst}_\varkappa = \langle \{1, 2\}, \{3, 5\}, \{4\}, \mathit{True, False, True, False, True} \rangle$$

the produced documents are of the form:

$$\begin{aligned}
\mathrm{d}_{1,\varkappa} &= (ag_1, ag_2, gain)\\
\mathrm{d}_{2,\varkappa} &= (ag_1, ag_2, loss)\\
\mathrm{d}_{3,\varkappa} &= (ag_3, ag_5, gain)\\
\mathrm{d}_{4,\varkappa} &= (ag_4, loss)\\
\mathrm{d}_{5,\varkappa} &= (ag_3, ag_5, gain)
\end{aligned}$$

As we can see, agents belonging in the same coalition may have identical documents, $d_{3,\varkappa} \equiv d_{5,\varkappa}$; while other agents may have different documents, $d_{1,\varkappa} \not\equiv d_{2,\varkappa}$.

4.6.2 Significant agents

In order to evaluate our experimental results, we first need to determine the coalition that is primarily described by each topic. For this reason, we define *significant agents* within each topic. That is, given a topic k, all the agents that appear with probability greater than a small number ε are considered to be significant. Formally:

Definition 18. Significant Agents *An agent $i \in N$ is considered to be significant with respect to a topic k if and only if $Pr(i|topic = k) \geq \varepsilon$, where $\varepsilon \in \mathbb{R}^+$ is a small, positive, real number.*

Therefore, we say that any agent, which is significant with respect to a topic, belongs to the coalition determined by this very topic. That is, topic k describes the coalition $S = \{j \in N : Pr(j|topic = k) \geq \varepsilon\}$.

4.6.3 Valid topics

After establishing the coalition described in each topic, we move to assessing the *validity* of the topic. Intuitively, the validity of a topic signifies whether the topic reflects a sub-formula describing the agent's hedonic preferences. Thus, given the significant agents within a topic, we characterize it as valid or invalid for agent i. A topic k is valid when:

- $Pr(gain|topic = k) \geq Pr(loss|topic = k)$ and $S = \{j : Pr(j|topic = k) \geq \varepsilon\} \in N_i^+$, or
- $Pr(loss|topic = k) \geq Pr(gain|topic = k)$ and $S = \{j : Pr(j|topic = k) \geq \varepsilon\} \in N_i^-$

otherwise the topic is invalid. In words, the meaning of the latter characterization is essentially the actual cross-validation of the topics result with the corresponding formula ϕ_i.

Given these definitions, we can then adopt as an evaluation metric the percentages of valid and invalid topics found by the algorithm. Intuitively, we would like the algorithm to discover topics that are valid, that is, they reflect preferences sub-formulae that correspond to satisfactory or dissatisfactory collaboration patterns.

4.6.4 Topic Significance constraint

In a more realistic scenario, knowing that the probability of 'gain' is greater than the one of 'loss', and vice versa, within a topic, may not be enough. Intuitively, we would like to be *confident* whether the coalition described within a topic is satisfactory or dissatisfactory. For this reason, we introduce a *topic significance constraint*, according to which a topic is labeled as significant if the absolute difference of the probability of term 'gain' and the probability of term 'loss', exceeds some small number δ, i.e. $|Pr(gain|topic = k) - Pr(loss|topic = k)| \geq \delta$. Thus, each topic is assessed as 'significant' with confidence level δ; and if a topic is 'significant' it can therefore assessed as 'valid' or 'invalid'. Now the definition of a valid topic k becomes:

- $Pr(gain|topic = k) \geq Pr(loss|topic = k) + \delta$ and $S = \{j : Pr(j|topic = k) \geq \} \in N_i^+$, or
- $Pr(loss|topic = k) \geq Pr(gain|topic = k) + \delta$ and $S = \{j : Pr(j|topic = k) \geq \} \in N_i^-$

4.6.5 Dataset and Setting Escalation

For the simulations we created several game instances of hedonic games with dichotomous preferences, according to the following procedure:

1. for each game G define the preference relations through ϕ formulae
2. generate randomly partitions of N, π
3. for each agent i decide whether $\pi(i)$ is satisfactory regarding formula ϕ_i
4. interpret and log the instance information $\langle \pi(i), satisfied_i \rangle$ into documents

Formulae Construction

A formula ϕ_i expresses agent i's goal in a concise and short representation. That is, each ϕ_i consists of two subsets of agents: *(a)* the "Appealing Partners" set and *(b)* the "Repellent Partners" set. In its simplest form, agent i is satisfied within a coalition where all agents in the "Appealing Partners" set are members of this coalition, and no agent in the "Repellent Partners" set is participating. However, an agent i may have several pairs of ⟨*Repellent Partners, Repellent Partners*⟩ subsets of agents, and be satisfied with a coalition if this coalition is consistent with at least one such pair. Therefore, in the general form we have that: $\phi_i = \bigvee_{l=1:L} \phi_{i,l} = \phi_{i,1} \vee \phi_{i,2} \vee \cdots \vee \phi_{i,L}$ where $\phi_{i,l} = Repellent\ Partners_{i,l} \wedge, Repellent\ Partners_{i,l}$. The complexity of a formula ϕ_i depends on (a) the number of $\phi_{i,l}$ and (b) the number of agents each pair

$$\langle Appealing\ Partners, Repellent\ Partners \rangle$$

contains. We ran our experiments on settings with escalated complexity of the structure of the formulae ϕ.

- Low Complexity: ϕ_is with low complexity consist of a single ⟨*Appealing Partners*, *Repellent Partners*⟩ pair, and the total number of agents appearing in this pair of subsets is fixed to $n/10$. That is, each agent has a must include or exclude demand on 10% of the agents.
- High Complexity: we escalate the complexity by increasing the number of ⟨*Appealing Partners*, *Repellent Partners*⟩ pairs to 3. The number of agents appearing in each pair uniformly ranges in $[n/10, n/5]$.[1,2]

Instance Generation & Information Logging

For a given hedonic game G, i.e. for a given set of formulae ϕ, we generate a number of game instances. In each instance we randomly partition agents into coalitions. Each coalition in the formed coalition structure is characterized as satisfactory or dissatisfactory according to formula ϕ_i, for every agent i within the coalition. After characterizing the coalition of a certain agent, we log the contained information into a text document using the interpretation described in Section 4.6.1.[3] The total number of documents produced per game varies depending on the game's complexity, and ranges between $[500K, 2.5M]$. However, the size of the corpus each online LDA model is fed with, corresponds only to the 2% of the total number of produced documents. That is, we use $10,000$ and $50,000$ game instances for the low and the high complexity environments respectively.

Note that the game instances do not have stochasticity. That is, for each generated sample (document) of a game instance, it is deterministically guaranteed that if the sample contains the word 'gain'('loss') then the respective coalition is satisfactory (dissatisfactory). In a real life situation of a boolean hedonic game formation, we would sample instances of the game by

[1] The number of agents participating in each ϕ_i, along with the total number of formulae per agent within each level of complexity environment were chosen so that the required dataset could be generated within a reasonable time frame; these numbers do not impose any burden on the LDA algorithm itself.

[2] We have preliminary results showing our approach can be quite effective in even more complex settings.

[3] For practical reasons, the logged information is repeated more than just once within a document. That is, we boost the term frequency of the agents' indicative words, along with the characterization 'gain' or 'loss', in order to avoid misleading words with low frequencies.

Complexity	$\|corpus\|$	$\|\phi_i\|$	number of Topics	batch size	iterations
Low	10000	1	5-12	5-25 (step 5)	50
High	50000	3	5-19(step 2)	5-25 (step 5)	50

Table 4.11: Simulation parameters

letting participants form groups, and then receive by each one a feedback on whether or not they were satisfied in their coalition.

LDA Model

For the implementation, we used the *scikit-learn* Python 3.5 library [Pedregosa et al., 2011]. As mentioned, the online version of LDA was used for a range of topics, iterations and batch sizes related to the formulae complexity. In Table 4.11 we present the different parameters used in the simulations.

The number of topics K, is a parameter that the (online) LDA model needs to be provided with. In situations such as the ones we are studying, the exact number of topics cannot be known a priori; however, it can systematically be chosen depending on the problem at hand. Moreover, there exist other LDA variations, such as HDP [Teh et al., 2004], that are non-parametric on the number of topics.

4.6.6 Experimental results

Before presenting our results, in Figure 4.10 we show two examples of topics (distributions over the "words"), in the low complexity environment. That is, each bar depicts the probability of a specific word belonging to the given topic. To clarify, the x-axis of the graphs depicts the vocabulary of our corpus, i.e. the bars $0 - 49$ correspond to the agent names; and the last two bars $50, 51$ represent the words 'gain' and 'loss', respectively. In the left graph, the distribution over the vocabulary is exhibited for a valid topic, which infers a satisfactory coalition–due to the relatively high probability of the word 'gain' (the 50^{th} word in the axis shown with green bar). Similarly, in the plot we see the graph for a dissatisfactory coalition scenario (the red bar there corresponds to the word 'loss').

We now present our actual results. First, we conducted a series of game simulations for the low complexity environment. Specifically, we constructed 4 different games that are subject

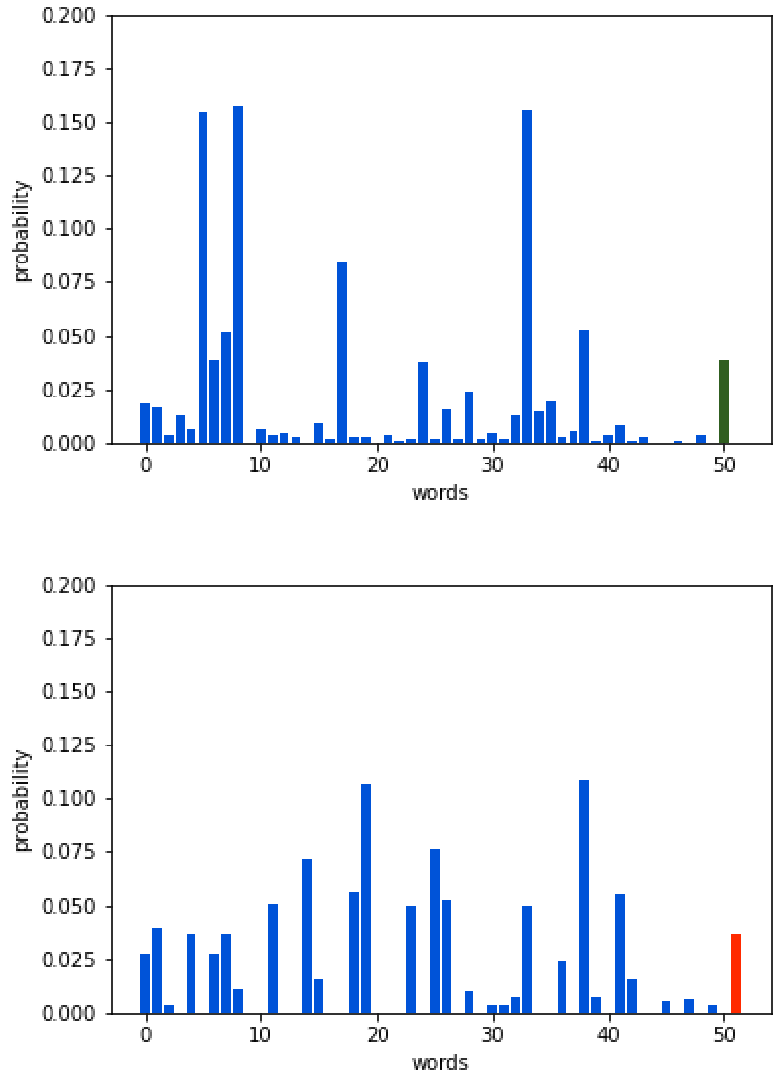


Figure 4.10: Environment Complexity: *low*. Example of a topic describing 'satisfactory' coalition (left), and a topic corresponding to a 'dissatisfactory' coalition (right).

Low Complexity			High Complexity		
Number of Topics	(%) valid	(%) invalid	Number of Topics	(%) valid	(%) invalid
5	89.83%	10.17%	5	84%	16%
6	91.81%	8.16%	7	87%	13%
7	96.79%	3.21%	9	90%	10%
8	92.08%	7.92%	11	90%	10%
9	93.98%	6.02%	13	86.92%	13.08%
10	92.67%	7.33%	15	84.64%	15.36%
11	95.53%	4.47%	17	80.59%	19.41%
12	88.89%	11.11%	19	78.16%	21.84%

Table 4.12: Percentage of *valid* and *invalid* topics over 4 different hedonic games with dichotomous preferences.

to the characteristics of the low complexity environment. For each one of these games we ran the learning process 5 times, using batches of different size (of the same documents) channeled to the LDA model.[4] The resulting topics were evaluated using the metrics described above, and we computed the average percentage of *valid* and *invalid* topics per game per number of topics. A similar evaluation process was followed for the high complexity environment. Again, we constructed 4 different games that are subject to the characteristics of the high complexity environment.

Table 4.12 shows our average results in the settings. As we can see, generally, the model in low complexity environment learns correctly at least 88% of the sub-formulae $\hat{\phi}_i$, for different numbers of topics.[5] At the same time, the percentage of incorrectly learned sub-formulae does not exceed 11.5% within this environment. Regarding the high complexity environment, we see that the average percentage of valid topics learned is greater than 78% for various number of topics. The average percentage of incorrectly learned topics does not exceed 22%. As the environment complexity increases along with the number of topics we intend to discover, the accuracy of learned collaboration patterns drops.

By taking into account the topic significance constraint, we re-evaluated the topics arisen from the LDA model for the low and the high complexity environment. The results depicted

[4] As we have already mentioned, there is no stochasticity during the dataset creation. However, by employing this repetition of the learning procedure per game, we ensure the robustness of our results.

[5] A larger number of topics allows for more preferences sub-formulae to be learned, but it naturally increases complexity.

in Table 4.13 show an expected drop in the average percentage of valid topics, since we discard a portion of the topics by assessing them as 'insignificant'. It is worth noting that the difference between the average percentage in the unconstrained and constrained cases reaches up to 15.27 percentage points for the assessment of valid topics. Figure 4.11 is the graphical representation of Table 4.13.

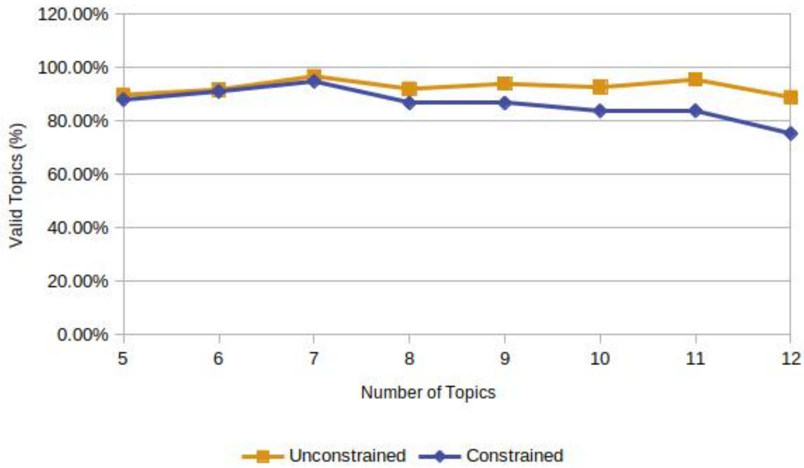
Average (%) valid topics							
Topics	low complexity			Topics	high complexity		
	(%) *unconstr*	(%) *constr*	diff		(%) *unconst*	(%) *constr*	diff
5	89.83%	88%	−1.83	5	84%	82%	−2
6	91.81%	91.11%	−0.7	7	87%	82.86%	−4.14
7	96.79%	94.88%	−1.91	9	90%	85.56%	−4.34
8	92.08%	86.77%	−5.31	11	90%	82.27%	−7.73
9	93.98%	86.76%	−7.22	13	86.92%	73.08%	−13.84
10	92.67%	83.83%	−8.84	15	84.67%	71%	−13.67
11	95.53%	83.86%	−11.67	17	80.59%	67.35%	−13.24
12	88.89%	75.35%	−13.54	19	78.16%	62.89%	−15.27

Table 4.13: Environment Complexity: *low* | *high*. Average percentage of *valid* topics over 4 different games with and without the significance constraint.

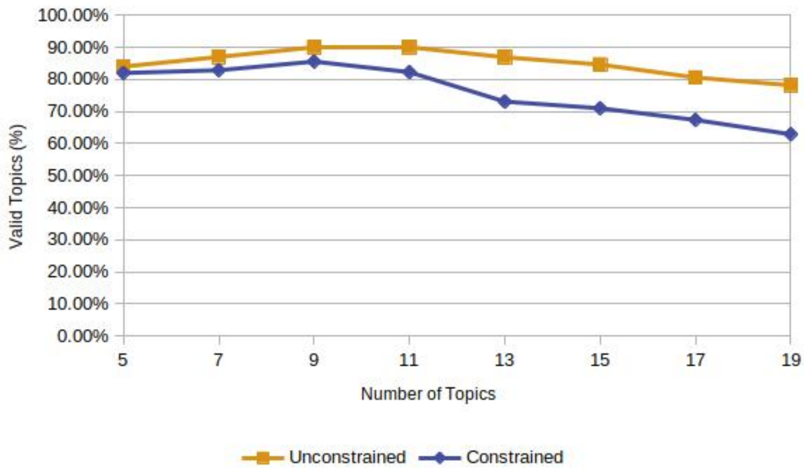
"Anytime" Behaviour

Last but not least, we conducted a simulation experiment to examine how our model behaves during an ongoing learning process. That is, assume that agents at a certain time t_1 have access to a part of the corpus. The agents train their models with the sub-corpus that is available to them at the time. After this first phase training, agents have some beliefs over satisfactory and dissatisfactory coalitions, that they could use in a decision-making process. At time t_2 a second part of the corpus is revealed to the agents, thus the agents update their already partially trained models with the new documents; and so on and so forth. In each of the later phases, the values of parameters of the model regarding the number of batches and iterations are maintained. Equivalently, these later phases correspond to LDA processes with prior distributions over topics and documents. Intuitively, using priors leads to faster convergence of the algorithm.

In Table 4.14 we show the results of this procedure, for games in the high complexity environment and for varying numbers of topics. We let the agents train in 3 phases and recorded



(a) Low Complexity Environment



(b) High Complexity Environment

Figure 4.11: Environment Complexity: *low* | *high*. Average percentage of *valid* topics over 4 different games with and without the significance constraint.

Topics	Phase 0		Phase 1		Phase 2	
	time (sec)	valid (%)	time (sec)	valid (%)	time (sec)	valid (%)
5	43.97*s*	80%	35.66*s*	100%	24.0*s*	100%
7	43.57*s*	77.14%	33.04*s*	97.14%	25.62*s*	100%
9	45.63*s*	71.11%	35.45*s*	93.33%	27.64*s*	97.78%
11	43.91*s*	67.27%	35.48*s*	90.18%	28.31*s*	90.91%

Table 4.14: Environment Complexity: *high*. Anytime behaviour of the training model.

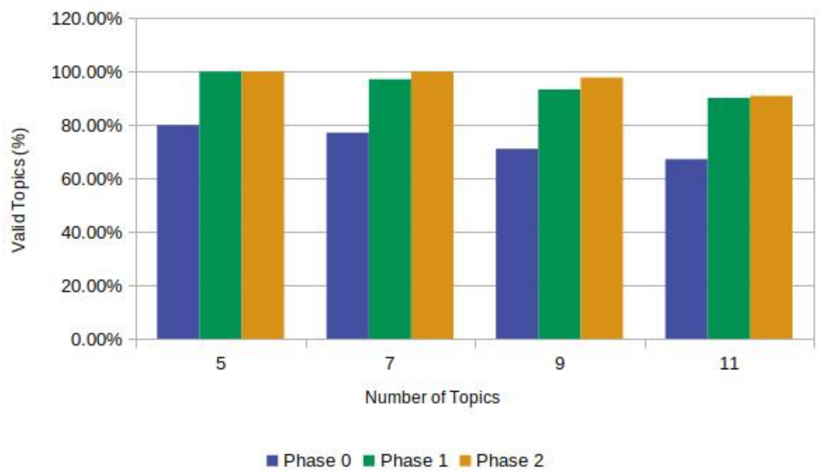


Figure 4.12: Environment Complexity: *high*. Anytime behaviour of the training model.

the time needed and the average percentage of valid topics for each phase. 2500 documents were fed to the model in each phase. As we can see, Phase 0 requires more time, while the average percentages of valid topics are not particularly encouraging; Phase 1 requires approximately $75-80\%$ of time required by Phase 0, and the percentages rise by up to 22 percentage points; similarly, Phase 2 requires approximately 40% of the time required by Phase 0, and the average percentage of valid topics consistently gets close to or even reaches 100%.

In Figure 4.14 we show the result's graphical representation of the anytime behaviour of the training model.

Note: What it is important to be noted here, is that we employed probabilistic topic modeling which yielded with really good results (reaching validity $\sim$ 78% and accuracy 80%) under an extreme scenario. That is, essentially we had only two features ('satisfactory' and 'dissatisfactory'), and a small vocabulary consisting of only 52 words; while PTMs are to process large series of documents, which in their turn consist of large vocabularies, and ultimately extract much more features than just two. As a result, our model performed extremely good in an extreme case.

Chapter 5

Hedonic Utility Games

In this chapter we provide one more theoretical extension of hedonic games. In fact, here we formally introduce a novel class of cooperative games the *Hedonic Utility Games (HUGs)*. We discuss the HUGs model intuition, and study this model under the notion of stability through several solution concepts.

As such, our contributions here are:

- we introduce a novel hybrid class of cooperative games, the HUGs, and provide a generic model;
- we walk through the application of existing solution concepts to HUGs;
- we propose a novel solution concept; and
- we propose an instantiation of hedonic preferences, study its properties within HUGs, and exploit it to obtain a probability bound for pruning the coalitional space.

5.1 Motivation

Cooperative games [Chalkiadakis et al., 2011] can be naturally distinguished into utility-driven games and hedonic games. This reflects agents' motivation during coalition (group) formation. In utility-driven games, an agent seeks to acquire the best possible payoff, and therefore joins the coalition that offers her the highest reward. With the term 'utility-driven' games we refer to classes of games such as Transferable Utility Games, Characteristic Function Games, etc. [Chalkiadakis et al., 2011]; such games, and all their subclasses share a common property: for each coalition there is a unique real value that intuitively represents the worth of the coalition, and this value corresponds to some dividable resource that can be distributed among the coalition members. By contrast, in hedonic games [Aziz et al., 2016b] as we have already thoroughly discussed, each agent is interested to participate in her most preferable coalition. That is, each agent ranks all coalitions depending solely on the identity of the coalitional members.

One could say that in the former case the agents form a preference relation over coalitions which is based on payoffs; while in the latter it is based on coalitional composition. However, in many real life settings, such an absolute demarcation among motives does not exist. On the contrary, people value (maybe in different proportions each) *both* hedonic preferences and payoff shares, when they are to collaborate with others in order to carry out a task. Thus, in the general case, when people are to form coalitions, they take into consideration all motivating aspects.

Although there are classes of games in cooperative game theory literature that can sufficiently model real cooperative problems; these classes so far ignore either their hedonic or their utility aspect. For this reason, here, we give a few examples of settings where hedonic and utility aspects naturally co-exist.

Example 11. Startup Companies *Consider an online platform that promotes startup companies' formation. Startup companies par excellence have a core (a group of friends) that share ideas, passion, way of thinking etc.; a group of friends that most likely will be sceptical about cooperating with individuals that are in rivarly. Despite that, since they constitute a working firm, this core of people also care about the prosperity of the company, i.e. they are interested in the profits of the company.*

Example 12. Social Ridesharing *Consider the problem of* social ridesharing *In ridesharing, a set of commuters form coalitions and arrange one-time rides at short notice. The goal is to* (a) *transport all commuters to their destination; and at the same time* (b) *minimize their expenses. In [Bistaffa et al., 2017], the authors adopt a cooperative game theoretic approach in order to tackle the problem considering only the utility aspect of the game, i.e. they satisfy* (a) *and* (b). *Nevertheless, in many cases a commuter i may prefer to* rideshare with her friends *even if this ride costs more than others. In other words, a commuter may be willing to pay more if she is to spend time with people she is having fun with.*

Example 13. Recommender System *think of a recommender system that is used by a travel agency. The travel agency is interested in creating groups of travellers that will* (a) *have a good time during their holidays,* (b) *meet their constraints/desires, for example a maximum total cost. Thus, the recommender system should form groups of travellers that appear to prefer each other's company, and at the same time plan a holiday package which is within their budget.*

Example 14. Participation in AI coding competition–revisited *Consider n programmers that take part in an AI coding competition. In this competition the programmers form groups to work together in order to build intelligent software that plays a board game, that will eventually compete each other in a tournament. Participating in such compete requires plenty of hours of intimate work with each other, thus each programmer highly values the identity of his/her team-mates. Moreover, each team (group of programmers) will receive a cash prize for participating, which is closely related to the team's performance in the tournament. As a result, each programmer desires to be in a team such that:*

- *he/she highly appreciates the other members of the team; and*
- *the team's performance is good enough, in order to receive a higher cash prize.*

As such, in Section 5.2 we discuss a generic model that combines hedonic and utility aspects, and as such we introduce *hedonic utility games* (HUGs). In Section 5.3 we discuss the application of existing stability concepts into the HUGs setting, while in Section 5.4 we put forward our novel theoretical IRIS solution concept, and study its complexity. In Section 5.5 we first extend the well-known dichotomous hedonic preferences model to a natural trichotomous preferences one, and study HUGs and IRIS in that setting. As part of our work there, we characterize feasible coalitions for HUGs, and exploit this feasibility concept to obtain a probability bound for pruning the coalitional space that ultimately reduce the computational load for obtaining kernel-stable payoff configurations in IRIS partitions.

5.2 Generic Model

As mentioned, in this work we combine hedonic with utility games. That is, players have hedonic preferences over coalitions, but also form utility-based preferences over coalitions. The *hedonic preferences* take into consideration the identity of coalition members, i.e. each player only cares about which players are in her own coalition. The *utility-based preferences* are derived from a utility (characteristic) function and/or the payoff share each player receives. As such, a *hedonic utility game* in its generality is driven by two main components: the *hedonic aspect*, i.e. hedonic preferences over coalitions based solely on each coalition's members composition; and the *utility aspect*, i.e. the utility obtained by a coalition, that eventually leads to a payoff rewarded to each player. In other words, the hedonic preferences are the component that attributes a personalized opinion on a given coalition, while the utility function along with the utility-based preferences are the component that attributes a "generally accepted" quantified opinion on the same coalition.[1]

Definition 19. *(Hedonic Preferences) Let* $N = \{1, \cdots, n\}$ *be a finite set of players, a hedonic preference relation is denoted by* $\succsim^{\text{hed}} = (\succsim_1^{\text{hed}}, \cdots, \succsim_n^{\text{hed}})$*, where* $\succsim_i^{\text{hed}} \subseteq N_i \times N_i$ *is complete, reflexive, and transitive relation, with* $N_i = \{S \subseteq N \text{ such that} i \in S\}$.

Definition 20. *(Utility-based Preferences) Let* $N = \{1, \cdots, n\}$ *be a finite set of players and* $v : 2^N \to \mathbb{R}$ *be a utility function that give rise to utility-driven preference relation* $\succsim^{\text{ut}} = (\succsim_1^{\text{ut}}, \cdots, \succsim_n^{\text{ut}})$*, where* $\succsim_i^{\text{ut}} \subseteq N_i \times N_i$ *is a complete, reflexive and transitive relation, with* $N_i = \{S \subseteq N \text{ such that} i \in S\}$.

The utility-based preferences $\succsim^{\text{ut}}$ can be defined in various ways, which depend on the player and/or the problem at hand. For instance, let v be a utility function, and $\text{x}(C)$ a payoff vector for some coalition $C \subseteq N$, then, $\succsim_i^{\text{ut}}$ could be of the form $C_1 \succsim_i^{\text{ut}} C_2$ if $x_i(C_1) \geq x_i(C_2)$; or, $\succsim_i^{\text{ut}}$ could solely depend on the utility function: $C_1 \succsim_i^{\text{ut}} C_2$ if $v(C_1) \geq v(C_2)$. One step further, we can say that each player forms an overall ordering over coalitions that takes into account both hedonic and utility-based preferences.

Definition 21. *(Overall Preferences) Let* $\succsim_i^{\text{hed}}$ *be a hedonic preference relation, and* $\succsim_i^{\text{ut}}$ *be a utility-based preferences relation, for some player i. Then, a function* $h_i(\succsim_i^{\text{hed}}, \succsim_i^{\text{ut}}) = \succsim_i^{\text{overall}}$*, blends the hedonic and utility-based preferences in order to produce a single overall ordinal preference relation over coalitions.*

[1] A HUG could be reminiscent of Multiple Objective Games [Zhao, 1991] where each player has a multi-dimensional utility vector, and needs to optimize every dimension. However, in HUGs the utility function is common to all agents, while there is also a distinct hedonic dimension in the preferences of each player.

Thus, a hedonic utility game in its generality is:

Definition 22. *(HUGs) A Hedonic Utility Game (HUG) $\mathcal{G}$ is given by a tuple* $\langle N; h_1, \cdots, h_n; \succsim_1^{\text{hed}}, \cdots, \succsim_n^{\text{hed}}; v\rangle$, *where for each* $i \in N$ $h_i(\succsim_i^{\text{hed}}, \succsim_i^{\text{ut}})$ *is a blending function that produces an overall preference relation* $\succsim_i^{\text{overall}}$ *combined by the hedonic preference* $\succsim_i^{\text{hed}}$ *and a utility-based preference* $\succsim_i^{\text{ut}}$ *derived from the utility function* $v : 2^N \rightarrow \mathbb{R}$. *The outcome of a HUG is a pair* $\langle CS, \mathrm{x}\rangle$, *where CS is a coalition structure; and* $\mathrm{x} \in \mathbb{R}^n$ *is a payoff vector related to CS.*

The blending function h_i takes into account player i's both the hedonic and the utility-based preferences, and produces an ordinal preference relation over coalitions. For each player i, function h_i may differ; so, let us examine how h can be formed in some base-line scenarios:

(i) A player i's overall preferences that depend *only* on the *hedonic* preferences, $h_i(\succsim_i^{\text{hed}}, \succsim_i^{\text{ut}}) = \succsim_i^{\text{hed}}$; i.e., even if $u(S) < u(T)$ or $x_i(S) < x_i(T)$ player i still prefers coalition S over T.

(ii) For some player i, function $h_i(\succsim_i^{\text{hed}}, \succsim_i^{\text{ut}}) = \succsim_i^{\text{ut}}$ depends *exclusively* on the *utility-based* preferences, that is player i's overall preferences is $S \succsim_i^{\text{overall}} T$ if and only if $v(S) \geq v(T)$ (or if and only if $x_i(S) \geq x_i(T)$). Or the i's overall preferences can be based on what we later call 'potentially individual rationality', i.e., $S \succ_i^{\text{overall}} T$ if and only if $v(S) \geq \sum_{j \in S} v(\{j\})$ and $v(T) < \sum_{k \in T} v(\{k\})$.

(iii) Another case is that of depending on both *hedonic* and *utility-based* preferences, i.e. let for some player i function $h_i(\succsim_i^{\text{hed}}, \succsim_i^{\text{ut}})$ be defined as "i prefers coalition S over coalition T if and only if $S \succsim_i^{\text{hed}} T$ and $v(S) \geq v(T) - \varepsilon_i$"; where ε_i is a threshold corresponding to an acceptable utility-loss, determined by i. Similarly could be the case of "i prefers coalition S over coalition T if and only if $S \succsim_i^{\text{hed}} T$ and $x_i(S) \geq x_i(T) - \varepsilon_i$", where ε_i now determines an acceptable payoff-loss.

(iv) A quite more complex scenario is that player i's function $h_i(\succsim_i^{\text{hed}}, \succsim_i^{\text{ut}})$ be defined as "i prefers coalition S over coalition T if and only if $\big(S \succsim_i^{\text{hed}} T$ and $v(S) \geq v(T) - \varepsilon_i^{max}\big)$ *or* $\big(v(S) \geq v(T) + \varepsilon_i^{min}$ regardless of the hedonic relation of i on S and $T\big)$"; where ε_i^{max} defines a maximum acceptable utility-loss for player i in order to satisfy her hedonic preferences, and ε_i^{min} defines a minimum desirable utility-gain for player i in order to ignore her hedonic preferences. Respectively, function $h_i(\succsim_i^{\text{hed}}, \succsim_i^{\text{ut}})$ could be also defined as "i prefers coalition S over coalition T if and only if $\big(S \succsim_i^{\text{hed}} T$ and $x_i(S) \geq x_i(T) - \varepsilon_i^{max}\big)$

or $\left(x_i(S) \geq x_i(T) + \varepsilon_i^{min}\right.$ regardless of the hedonic relation of i on S and $\left.T\right)$"; where now ε_i^{max} and ε_i^{min} correspond to a maximum acceptable loss and a minimum desirable gain, respectively.

These are just some scenarios: in general $h_i(\succsim_i^{\text{hed}}, \succsim_i^{\text{ut}})$ can take any form. Thus, each player can value her hedonic and utility-based preferences differently. Note that this blending function h_i may result to a preference relation that is *non-transitive*; in our view, this is a very interesting property that alters the way the concept of 'rationality' is perceived in settings where both the hedonic and the utility aspect affect the outcome. In other words, in real-life settings where we deal with interpersonal relations *and* imminent payoff, rationality stops being straightforward and adapts in such a complex environment.

To help our discussion in the rest of the paper, we now provide an alternative definition equivalent to Definition 22, but which explicitly refers to the $\succsim_i^{\text{hed}}$ and $\succsim_i^{\text{ut}}$ aspects of the problem. As such, HUGs' definition now becomes:

Definition 23. *(HUGs–alternative) A Hedonic Utility Game $\mathcal{G}$ is given by a tuple $\langle N; \succsim^{\text{hed}}; v\rangle$, where $\succsim^{\text{hed}} = (\succsim_1^{\text{hed}}, \cdots, \succsim_n^{\text{hed}})$ is a vector of hedonic preference relations, one for each i; and $v : 2^n \to \mathbb{R}$ is a utility function. The outcome of a HUG is a pair $\langle CS, \mathrm{x}\rangle$, where CS is a coalition structure; and $\mathrm{x} \in \mathbb{R}^n$ is a payoff vector related to CS.*

To ease notation, from now on we use $\succsim_i$ to refer to $\succsim_i^{\text{hed}}$, unless explicitly stated otherwise. Also, henceforth, we use $S \succ_i T$ to denote that i strictly prefers coalition S to T; and $S \sim_i T$ to denote that i is indifferent between coalitions S, T. Moreover, in the rest of the paper we use the following notation: N is a set of players and $N_i = \{S \subseteq N : i \in S\}$ is the set of all coalitions that contains agent i; π is a partition (also referred to as coalition structure CS) of N, while $\pi(i) \equiv S \in \pi : i \in S$, is the single coalition within π that contains agent i.

5.3 Existing Solution Concepts applied in HUGs

In this section we discuss how several existing stability solution concepts can be applied on the HUG model. We focus on solution concepts from the literature of Hedonic Games and the Transferable Utility Games in order to approach HUGs from both aspects.

5.3.1 Individual Rationality

First we discuss the concept of *individual rationality (IR)*. Individual rationality is a notion one finds on both hedonic and TU games [Chalkiadakis et al., 2011]. In hedonic games, a partition π is individually rational if for every agent $i \in N$ it holds that $\pi(i) \succsim_i \{i\}$. In words, π is IR if each i prefers its current coalition, $\pi(i)$, at least as much as the singleton $\{i\}$ [Bogomolnaia and Jackson, 2002, Aziz et al., 2016b]. In TU games, a partition π with respect to a payoff vector x, is individually rational if for every agent i holds that $x_i \geq v(\{i\})$. In words, π is IR if each agent i receives a payoff that is at least as good as what she can earn on her own [Chalkiadakis et al., 2011].

Now, given a hedonic utility game $\mathcal{G} = \langle N; \succsim; v \rangle$, a partition π of N is individually rational if π is IR in both hedonic and TU terms. That is, *wrt* a payoff vector x related to π, for every $i \in N$ it holds that $\pi(i) \succsim_i \{i\}$, and at the same time, $x_i \geq v\big(\{i\}\big)$.

5.3.2 Individual Stability

The concept of individual stability of coalition structures is a key notion in the hedonic games literature [Dréze and Greenberg, 1980b, Bogomolnaia and Jackson, 2002, Aziz et al., 2016b]. In an individually stable (IS) partition, no agent prefers to unilaterally deviate into a new coalition and, at the same time, is welcomed by this new coalition.

Definition 24. *(IS-deviation in HUGs) In a HUG $\mathcal{G} = \langle N; \succsim; v \rangle$, given a partition π, an agent i can IS-deviate into a coalition $S \in \pi \cup \{\emptyset\}$ if it holds that $S \cup \{i\} \succ_i \pi(i)$ and for each $j \in S$ it holds that $S \cup \{i\} \succsim_j S$.*

A partition π is individually stable if no agent can IS-deviate. By its definition, individual stability includes individual rationality, i.e. if a partition satisfies individual stability then it satisfies individual rationality (in the hedonic sense) as well. For pure hedonic games, computing or even deciding the existence of IS partitions is NP-complete [Ballester, 2004]; thus, since considering solely the hedonic aspect of a HUG with arbitrary hedonic preference relations the game coincides with a pure hedonic game, it is straightforward that it is NP-complete to find an individually stable partition in a HUG.

For TU games there is no 'individual stability' solution concept;[2] however, in a utility game

[2] One could, of course, consider the special case of the *core* [Chalkiadakis et al., 2011] where an agent forms a profitable deviating coalition by joining an existing one or by staying alone.

an IS-deviation can be thought of as follows: given a partition π, an agent i can IS-deviate into $S \in \pi \cup \{\emptyset\}$ iff $x_i(S \cup \{i\}) > x_i(\pi(i))$ and for each $j \in S$ we have that $x_j(S \cup \{i\}) \geq x_j(S)$. Thus, we can provide an *enhanced* definition for IS-deviation for HUGs as:

Definition 25. *(Enhanced IS-deviation in HUGs) In a HUG $\mathcal{G} = \langle N; \succsim; v \rangle$, given a partition π, an agent i can IS-deviate into a coalition $S \in \pi \cup \{\emptyset\}$ if it holds that $S \cup \{i\} \succ_i \{i\}$ and $x_i(S \cup \{i\}) > x_i(\pi(i))$; and also, for each $j \in S$ it holds that $S \cup \{i\} \succsim_j S$ and $x_j(S \cup \{i\}) \geq x_j(S)$.*

Note that we said nothing on how to compute the payoff $x_i(S)$; one can arbitrarily set a complete representation $\mathrm{X} : 2^n \to \mathbb{R}^n$, such that for every coalition $S \subseteq N$ there is a payoff $x_i(S) \in \mathrm{X}$ for each $i \in N$.[3]

5.3.3 Core Stability

The strongest cooperative solution concept regarding stability is the *core*, the set of outcomes where no subset of players has an incentive to deviate. The core solution concept is well-defined and well-studied in both hedonic games [Aziz et al., 2016b] and utility-driven games [Chalkiadakis et al., 2011]. For interest and completeness, we define the core of HUGs in a straightforward manner.

Definition 26. *(Core of HUGs) Given a hedonic utility game $\mathcal{G} = \langle N; \succsim; v \rangle$, a pair $\langle S, \mathrm{y} \rangle$ blocks the outcome $\langle \pi, \mathrm{x} \rangle$ if for every $i \in S$ holds $S \succ_i \pi(i)$ and $y_i > x_i$. The core $C(\mathcal{G})$ of a HUG is the set of all partitions π that admits no blocking pairs $\langle S, \mathrm{y} \rangle$.*

5.3.4 Kernel Stability

The kernel is an exclusively utility-driven games' solution concept, defined *given* a coalition structure. It was introduced in [Davis and Maschler, 1965] as a subset of the bargaining set [Chalkiadakis et al., 2011] of a cooperative game. It consists of all outcomes where any pair of agents are in bilateral *equilibrium*; that is, no player can claim a share of another player's payoff. The kernel is always non-empty [Schmeidler, 1969]. Nonetheless, computing the kernel is itself hard; [Aumann et al., 1965] proposed a set of rules for determining the kernel, but this is inapplicable for large settings.

[3] We make the intuitive assumption that for every $j \notin S$ we have $x_j(S) = 0$, and $x_i(\emptyset) = 0$ for each $i \in N$.

The key element for the kernel computation is the *excess* of a coalition S, defined as $e(S, \mathrm{x}) = v(S) - \sum_{i \in S} x_i$, which expresses the gain / loss to the members of S if they do not accept the payoff distribution x. The maximum *surplus* of player i over j, $s_{i,j}(\mathrm{x}) = \max_{S \in T_{i,j}} e(S, \mathrm{x})$ where $T = \{S \subset N | i \in S \text{ and } j \notin S\}$, represents the maximal amount that i can gain by joining a coalition $D \in T_{i,j}$, with the understanding that the other members of D are satisfied with getting the same with their current payoff. If i outweighs j ($s_{i,j}(\mathrm{x}) > s_{j,i}(\mathrm{x})$ and $x_j > v(\{j\})$), then i can claim a portion of j's payoff. If neither i outweighs j nor j outweighs i, then the pair i, j is in bilateral equilibrium.

The Stearns transfer scheme [Stearns, 1968] described a payoff transfer scheme that converges to the game's kernel. This transfer scheme performs a series of k-transfers that rearrange the payoff configuration such that each pair of agents in a coalition is in bilateral equilibrium. Regardless, this may require an infinite number of steps. Under that realization, [Shehory and Kraus, 1999] provided a modification that allows fast convergence, given a specified error . Both schemes transform a payoff vector to a kernel-stable one with respect to some partition. Thus, in principle one could have a HUG $\mathcal{G} = \langle N; \succsim; v \rangle$, provided along with an IS coalition structure, via the algorithms presented in Section 5.3.2; and an arbitrary initial payoff vector x, which can then be transformed into a kernel-stable one. However, this transformation procedure may result to a payoff configuration that is *neither individually stable* (under Enhanced IS-deviation), nor even *individually rational*; subsequently, that would allow incentives for deviations. We now introduce a new solution concept that helps us overcome this problem.

5.4 The IRIS Solution Concept

In this section, we propose a novel solution concept designed specifically for the hybrid model of HUGs. In Subsection 5.3.2, we presented an extension of the notion of individual stability that considers utility as well. However, this enhanced version of individual stability for HUGs, requires an explicitly 'predefined' payoff configuration space, a payoff configuration per each of all possible partitions. That would be prohibitive, in terms of space, even for a small, let alone for a large number of players. Here, we propose the novel *individually rational - individually stable (IRIS)* solution concept that does not suffer from this problem. In IRIS, we have two requirements:

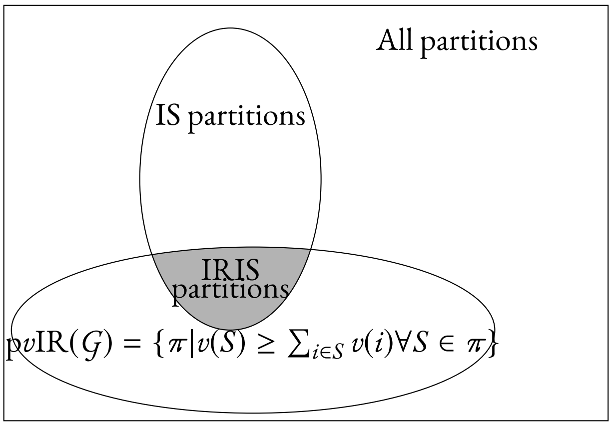


Figure 5.1: IRIS partition space

- the partition is individually stable as far as the hedonic aspect of the game is concerned, and
- the partition is such that an individually rational payoff *can potentially* be provided.

In other words, we seek partitions π that satisfy the hedonic individual stability concept, and at the same time the coalitional values are such that all players can claim a payoff that is at least as good as what they can earn on their own; i.e., partitions that are hedonically individually stable, and partition that provide *imputations* (for instance, kernel-stable partitions). Such (IRIS) partitions (may) exist in the intersection of the set of individually stable partitions and the set of partitions where each agent can at least claim what she can earn on her own $\{\pi : v(S) \geq \sum_{i \in S} v(\{i\}), \forall S \in \pi\}$ as illustrated in Figure 5.1. Let us denote the set of individual stable partitions as IS($\mathcal{G}$), and the set $\{\pi : v(S) \geq \sum_{i \in S} v(\{i\}), \forall S \in \pi\}$ as (potentially v-Individually Rational) pvIR($\mathcal{G}$). To formally define IRIS, let us first define the concept of *v-rationalizing deviation*:

Definition 27. *(v-Rationalizing Deviation) Given a partition π of N, a deviation of agent i from $T \in \pi$ into $S \in \pi \cup \{\emptyset\}$ is called* v-rationalizing *if $v(T) < \sum_{j \in T} v(\{j\})$ and $v(S \cup \{i\}) \geq v(\{i\}) + \sum_{j \in S} v(\{j\})$.*

We can now exploit Definition 24 and Definition 27 to construct the formal expression of *IRIS deviation*, and the characterization of *IRIS partitions*:

Definition 28. *(IRIS-deviation) Given a HUG $\mathcal{G} = \langle N; \succsim; v \rangle$ and a partition π, agent i can* IRIS-deviate *into $S \in \pi \cup \{\emptyset\}$ if*

- *i can IS-deviate into S; or*
- *i can perform a v-*rationalizing *deviation into S.*

Definition 29. *(IRIS partition) Given a HUG $\mathcal{G} = \langle N; \succsim; v \rangle$, a partition π is* individually rational-individually stable *if no agent can IRIS-deviate. That is,*

(1) $\nexists i, S \in \pi \cup \{\emptyset\}$ s.t. $S \cup \{i\} \succ_i \pi(i)$, and $S \cup \{i\} \succsim_j S, \forall j \in S$; and

(2) $\forall i \in N$ it holds that $v(\pi(i)) \geq \sum_{k \in \pi(i)} v(\{k\})$.

The second condition in Definition 29 ensures that in an IRIS partition, there *can* exist a payoff configuration that is individually rational. In utility-driven games the payoff vector results from the distribution of each coalition's utility to its members. In IRIS partitions, by allowing *only* the coalitions that can afford to reward their members with a payoff at least as good as their individual utility, we can guarantee that there exists at least one payoff configuration that is individually rational. We clarify that we slightly abuse the term 'individual rationality', since we do not consider any payoff vector in particular, but coalition structures that can *potentially lead* to an individually rational payoff configuration.

Note that we are able to define and use the IRIS solution concept exactly due to the hybrid nature of HUG settings. That is, we could not have had the IRIS solution concept in a pure utility-driven game, since the notion of individual stability is not defined in such settings; nor could we have had it in pure hedonic games where the notion of utility is not defined. Notice, however, that by dropping the first condition of Definition 29 we would be able to consider a special case of individual rationality in pure utility-driven settings where condition (2) holds. At the same time, by dropping the second condition we would end up with the concept of individual stability in pure hedonic games; as such, Definition 29 generalizes the individual stability concept to HUGs. Notice also that IRIS is a strengthening of individual stability: every IRIS partition is always IS, but the opposite is not necessarily true.

Complexity of IRIS In Algorithm 4 we provide an $O(n)$ algorithm that checks if an agent can IRIS-deviate into a coalition.

Name: isInIRIS
Instance: A HUG $\mathcal{G} = \langle N; \succsim; v \rangle$, and a partition π
Question: Is π in IRIS($\mathcal{G}$)?

Name: ExistsIRISPartition
Instance: A hedonic utility game $\mathcal{G} = \langle N; \succsim; v \rangle$
Question: Is there a π that is an IRIS partition?

Figure 5.2: Two IRIS-related decision problems.

Algorithm 4: CanIRIS-Deviate(i,S,$\succsim$,v,π)

```
if( CanIS-Deviate(i,S,≿,π) ): return True;
current_value ← ∑_{k∈π(i)} v(k);
if( current_value ≥ v(π(i)) ): return False;
new_value ← ∑_{j∈S} v(j) + v(i);
if( new_value ≥ v(S ∪ {i}) ): return True;
return False;
```

Algorithm 5: checkIRISPartition($\mathcal{G}$,π)

```
1 for ( every existing coalition S ∈ π ):
2     for ( every agent i ∈ N \ S ):
3         if( CanIRIS-Deviate(i,S,≿,v,π) ): return False;
4 return True;
```

Proposition 1. *The problem isInIRIS (Fig. 5.2) is decidable in polynomial time.*

Proof. Go through Algorithm 5. The conjunction of the for-loops in lines 1 and 2 executes at most n^2 times, while for checking CanIRIS-Deviate(i,S,$\succsim$,v,π) we need at most $3 \cdot n$ computations ($O(n)$). Thus we can check in polynomial time, $O(n^3)$, whether a given partition of a HUG is IRIS or not. □

Even though it is easy to decide if a given partition is IRIS, it is also essential to solve the decision problem ExistsIRISPartition (see Figure 5.2). To answer this problem we need to either find a partition that satisfies conditions (1) and (2) of Definition 29, or decide that there is no such partition. However, in general, these two conditions are completely unrelated; that is, since they refer to different aspects of a HUG (the former to the hedonic aspect, while the latter to the utility aspect), having information that regards the first condition, provides us with no information regarding the second one, and vice versa. Therefore, a machine that decides the ExistsIRISPartition problem needs to solve *two unrelated, separate problems*, and come with a partition that is an answer to both problems or *halt* if there is none such partition. Since one of the problems is NP-complete [Ballester, 2004], ExistsIRISPartition is NP-hard. We can also explicitly prove Proposition 2 below:

Proposition 2. *It is NP-hard to find an IRIS partition in a HUG $\mathcal{G} = \langle N; \succsim; v \rangle$ with arbitrary hedonic preferences.*

Proof. Suppose we have a hedonic game $\langle N, \succsim \rangle$ with arbitrary preference relations, exactly as the model considered in [Ballester, 2004]. Add to this game a superadditive utility function v to get a HUG $\mathcal{G} = \langle N; \succsim; v \rangle$. Due to superadditivity, condition 2 of Definition 29 always stands; thus a partition π is IRIS if and only if it satisfies condition 1 of Definition 29, i.e. if and only if it is IS. However, Ballester showed in [Ballester, 2004] that it is NP-complete to find an IS partition in a game with arbitrary hedonic preferences. Therefore, it is NP-hard to find an IRIS partition with arbitrary preferences. □

5.4.1 A randomized transitions scheme to find IRIS partition

In Proposition 2 we showed that it is NP-hard to build an IRIS partition. However, suppose we have an oracle which informs us that IRIS($\mathcal{G}$) is non-empty for some HUG $\mathcal{G} = \langle N; \succsim; v \rangle$. Therefore, we can use a transitions scheme to reach a point in IRIS($\mathcal{G}$).

We begin with a random partition; if this partition is IRIS we stop. Otherwise, we successively perform IRIS deviations. In the case we have stuck in a loop, i.e., the partition transit from point A to point B and vice versa, we perform a random deviation. At some point following this procedure we will reach the IRIS partition, however this may take infinite number of steps. Nonetheless, given that the oracle verifies the IRIS non-emptiness, the time needed for the transitions scheme to converge into an IRIS partition is highly dependent on the sizes of the IS($\mathcal{G}$) and pvIR($\mathcal{G}$) sets, and the proportion of their intersection with respect to the total area of partitions (see Figure 5.1). In many settings where IRIS partition exists, we expect that this time will not be prohibitive in practice, though this is to be verified via simulations within specific environments.

1. Get a random partition π
2. If π is IRIS: stop
3. Repeat:
   a) Perform an IRIS-deviation
   b) If π is IRIS: stop
   c) If π has stacked in a loop: perform a random deviation

Figure 5.3: A randomized transitions scheme.

5.5 Instantiation of HUGs with Trichotomous Preferences

So far within the HUGs framework, we considered the hedonic preferences to be arbitrary. However, here we present a modification of the well-known *dichotomous hedonic preferences*[4] that will allow us to obtain certain algorithmic results for solution concepts in HUGs. Dichotomous preferences were introduced in [Aziz et al., 2016a] to suggest that for each player the related coalition space can be partitioned into two disjoint subsets N_i^+ and N_i^-. In our trichotomous preferences modification we now explicitly require that each player i:

- strictly prefers all coalitions in N_i^+ to singleton,
- strictly prefers singleton to all coalitions in N_i^-, and
- is indifferent about coalitions in the same subset.

That is, for some $S, T \in N_i$ we have $S \succ_i \{i\} \succ_i T$ if and only if $S \in N_i^+$ and $T \in N_i^-$, and $S \sim_i T$ if and only if $S, T \in N_i^+$ or $S, T \in N_i^-$. Also, $\forall i\ N_i = N_i^+ \cup \{i\} \cup N_i^-$. The trichotomous preferences model is an intuitive paradigm that holds in most real life settings; for instance, consider work groups for a school project assignment, a student would prefer to be in a group with her friends than being alone, and in the same time would prefer being alone rather in group of people she dislikes.

Therefore, if we adopt trichotomous hedonic preferences, we can build on [Peters, 2016] and obtain an IS partition in $O(n^3)$ using Algorithm 6.

[4]So far, in this thesis we have refer to Games with Dichotomous Hedonic Preferences as Boolean Hedonic games.

Algorithm 6: IS Partition(N,$\succsim$,v)

```
π ← ∅;
∀i ∈ N assign i into {i};
while ( there are agents that can IS-deviate ):
    for ( every agent i that is singleton ):
        for ( every existing coalition S ∈ π ):
            if( CanIS-Deviate(i,S,≿,π) ):
                assign agent i into coalition S;
return π;
```

Algorithm 7: CanIS-Deviate(i,S,$\succsim$,π)

```
if( S ∪ {i} is strictly preferable to i than her current coalition π(i) ):
    for ( every agent j ∈ S ):
        if( S is strictly preferable to j than coalition S ∪ {i} ):
            /* i cannot IS-deviate into S since j objects this deviation.   */
            return False
    /* At this point there is no objection, thus i can IS-deviate into S.   */
    return True
return False
```

Having in mind, that for HUGs with trichotomous preferences the IS($\mathcal{G}$) is non-empty, and that we can reach an IS partition in polynomial time, along with the fact that the pvIR($\mathcal{G}$) is always non-empty for any HUG, the immediate question arises: for a HUG with trichotomous hedonic preferences, is the intersection IRIS($\mathcal{G}$) = IS($\mathcal{G}$) ∩ pvIR($\mathcal{G}$) non-empty? The answer is we have no guarantees that even under these simplifying assumptions an IRIS partition exists. For instance consider example 15.

Example 15. *Consider a 4-player HUG with trichotomous preferences such that:* $N_1^+ = \big\{\{1,2\},\{1,3,4\},\{1,2,4\}\big\}$, $N_2^+ = \big\{\{2,4\},\{1,2,4\},\{1,2,3,4\}\big\}$, $N_3^+ = \big\{\{1,3,4\},\{1,3\}\big\}$, $N_4^+ = \big\{\{1,4\},\{2,4\},\{1,2,4\},\{1,3,4\}\big\}$, *and* $v(\{i\}) = 2\ \forall\ i \in N$, $v(S) = |S|\ \forall\ S \subseteq N : |S| = 2,3$, *and* $v(N) = 8$. *According to these preferences, the individually stable partitions are the following:*

$$\text{IS}(\mathcal{G}) = \Big\{\big\{\{1,2,4\},\{3\}\big\},\big\{\{1,3,4\},\{2\}\big\}\Big\}$$

while according to v the potentially individually rational partitions are the following:

$$\mathrm{p}v\mathrm{IR}(\mathcal{G}) = \Big\{\big\{\{1\},\{2\},\{4\},\{3\}\big\},\big\{\{1,2,3,4\}\big\}\Big\}.$$

Therefore, $\mathrm{IRIS}(\mathcal{G}) \equiv \mathrm{IS}(\mathcal{G}) \cap \mathrm{p}v\mathrm{IR}(\mathcal{G}) = \emptyset$, *there is no IRIS partition even if the setting is a simplified HUG with trichotomous preferences.*

Now, if the oracle verifies that for a HUG $\mathcal{G} = \langle N; \succsim; v\rangle$ with trichotomous preferences the $\mathrm{IRIS}(\mathcal{G})$ is non-empty, then we can use the aforementioned transition scheme (Subsection 5.4.1) and reach an IRIS partition where every coalition is individually stable in terms of hedonic preferences, and also has the 'capability' to provide an individually rational payoff vector. However, we said nothing so far about the agent's actual payoffs. One could use the classic transfer scheme proposed by [Shehory and Kraus, 1999], and build a kernel-stable configuration for any IRIS coalition structure. Nonetheless, as pointed out by [Bistaffa et al., 2017], the Shehory-Kraus transfer scheme will become eventually inefficient when the number of agents is increased. In [Bistaffa et al., 2017] Bistaffa *et al.* overcome this problem by not considering infeasible coalitions. In [Shehory and Kraus, 1999] the authors also discuss about predefining an acceptable range of sizes for coalitions in order to achieve polynomial complexity. Similarly, in a HUG we can reduce the coalitional search space by disregarding infeasible coalitions.

Although in general we may not be able to perform any pruning, under the assumption of trichotomous hedonic preferences we can discard infeasible coalitions.

Definition 30. *(Feasibility via trichotomous properties) Given a HUG* $\mathcal{G} = \langle N; \succsim; v\rangle$, *a coalition* $S \subseteq N$ *is infeasible if and only if for at least one* $i \in S$ *it holds that* $S \in N_i^-$, *otherwise S is feasible:*

$$\begin{aligned} S\ \textit{infeasible}^{\text{trich}} &\Leftrightarrow \exists\, i \in S : S \in N_i^-, \\ S\ \textit{feasible}^{\text{trich}} &\Leftrightarrow \forall i \in S : S \in N_i^+ \text{ or } |S| = 1. \end{aligned}$$

According to Definition 30, we consider as feasible coalitions only those that are hedonically immune to deviations. That is, any coalition $S \in N_i^-$ for some $i \in S$, is unstable since agent i would prefer to deviate into a singleton. Thus, if a coalition is non-acceptable for at least one of its members, then this coalition is unstable, and therefore there is no benefit in being included in the computations of the kernel. Note that by disregarding infeasible coalitions we lose no individually stable partitions, i.e., we lose no coalitions that can be part of any individually stable partition:

Proposition 3. *Given an IS partition π of a HUG $\mathcal{G} = \langle N; \succsim; v \rangle$, $\nexists\, S \in \pi$ s.t. $S \in N_i^-$ for some $i \in S$.*

Proof. The hedonic preferences in a HUG are defined according to trichotomous preferences model. That is for any agent $i \in N$ it holds that $S \succ_i \{i\} \succ_i T$ if and only if $S \in N_i^+$ and $T \in N_i^-$; in other words, each agent i strongly prefers to be member of any coalition in the set N_i^+ rather than being alone, but also she strongly prefers to be alone rather than being in any coalition in the set N_i^-. Now, assume there is a coalition $S \in \pi$ s.t. $S \in N_i^-$ for some $i \in S$. As indicated above, this particular agent i, strongly prefers to be on her own instead of being in S, motivating i to deviate into an empty coalition ($\{\}$). This means that π is not stable. *Therefore, in a individually stable HUG partition π, there is no coalition $S \in \pi$ s.t. $S \in N_i^-$ for some $i \in S$.* □

Since IRIS partitions are IS, and pruning is not related to their pvIR($\mathcal{G}$) component, we *lose no IRIS partitions* either.

Proposition 4. *Given an IRIS partition π of a HUG $\mathcal{G} = \langle N; \succsim; v \rangle$, $\nexists\, S \in \pi$ s.t. $S \in N_i^-$ for some $i \in S$.*

We may also consider the following simple setting: the trichotomous preferences over coalitions are actually lifted preferences over players. That is, let each $i \in N$ develops an "empathy" value e_i^j towards any other player $j \in N$, expressing i's perception as to how well it can collaborate with j; and a coalition $S \subseteq N \setminus \{i\}$ is placed in N_i^+ if a function $f(S, \mathrm{e}_i)$ meets a threshold t_i. This function f can be a summation of values e_i^j over the j in the coalition (as in Additively Separable Hedonic Games [Aziz et al., 2016b]), or average of these values (as in Fractional Hedonic Games [Aziz et al., 2014]), or the pairwise average of these values:

$$f(S, \mathrm{e}) = \frac{1}{2} \cdot \sum_{i \in S} \sum_{j \in S} \frac{e_i^j + e_j^i}{2}$$

Now, if we let each value $e_i^j \sim N(\mu_i, \sigma_i^2)$ be Gaussian independent and identically distributed (i.i.d.) random variables, the pairwise average

$$f(S, \mathrm{e}) \sim N(\frac{|S|}{2}\sum_{i\in S}\mu_i, \frac{|S|^2}{128}\sum_{i\in S}\sigma_i^2)$$

is also a Gaussian random variable. Thus, using this universal (common to all agents) pairwise average function, we can obtain a probability bound on when a coalition is pruned according to trichotomous preferences.

Proposition 5. *Given a HUG $\mathcal{G} = \langle N; \succsim; v\rangle$, with trichotomous preferences following i.i.d. $e_i^j \sim N(\mu_i, \sigma_i^2)$ and the pairwise average function $f(S, \mathrm{e})$, a coalition $S \subseteq N$ is pruned with probability:*

$$P(S \textit{ be trich-pruned}) \geq 1 - \frac{|S| \cdot \sum_{i\in S}\mu_i}{2 \cdot \max_{i\in S} t_i}$$

Proof. Consider feasibility according to Def. 30; we could reform the condition as follows: S feasible$^{\text{trich}} \Leftrightarrow f(S, \mathrm{e}) \geq \max_{i\in S}\{t_i\}$. Thus, the probability of a coalition to be pruned exploiting the properties of the trichotomous preferences model, is:

$$\begin{aligned} P(S \text{ be trich-pruned }) &= 1 - P(S \text{ feasible}^{\text{trich}}) \\ &= 1 - P\Big(f(S, \mathrm{e}) \geq \max_{i\in S}\{t_i\}\Big). \end{aligned}$$

Now, exploiting Markov's Inequality [Mitzenmacher and Upfal, 2005] we have that:

$$\begin{aligned} P\Big(f(S, \mathrm{e}) \geq \max_{i\in S}\{t_i\}\Big) &\leq \frac{E[f(S, \mathrm{e})]}{\max_{i\in S}\{t_i\}} \Leftrightarrow \\ P\Big(f(S, \mathrm{e}) \geq \max_{i\in S}\{t_i\}\Big) &\leq \frac{\frac{|S|}{2} \cdot \sum_{i\in S}\mu_i}{\max_{i\in S}\{t_i\}} \Leftrightarrow \\ 1 - P\Big(f(S, \mathrm{e}) \geq \max_{i\in S}\{t_i\}\Big) &\geq 1 - \frac{|S| \cdot \sum_{i\in S}\mu_i}{2 \cdot \max_{i\in S}\{t_i\}} \Leftrightarrow \\ P(S \text{ be trich-pruned }) &\geq 1 - \frac{|S| \cdot \sum_{i\in S}\mu_i}{2 \cdot \max_{i\in S}\{t_i\}} \end{aligned}$$

□

Chapter 6

Conclusions and Future Work

6.1 Conclusions

This thesis approached the class of Hedonic Games from both a theoretical and a practical perspective. We provided novel theoretical extensions that allow us to model real-world scenarios under a more realistic point of view. At the same time, we tackled the natural problem of uncertainty within the framework of hedonic games.

Specifically, our practical contributions include studying and exploiting well-known machine learning models such as linear regression, regression with basis functions, feed forward neural network, and probabilistic topic modeling in order to extract and discover hidden preferences and collaboration patterns. We conducted a thorough experimental evaluation of the aforementioned learning models. In the process, we developed two evaluation metrics, and devised a novel interpretation method of coalitions into text documents.

Our work also resulted to a number of theoretical contributions. We provided some natural extensions on the classic framework of hedonic games: In particular, we put forward the formal definition of Hedonic Games in Partition Function Form, a generalization of hedonic games, in which each agent owns a preference relation over partitions; we extended well-studied classes of hedonic games into their generalized model; and studied these classes of games in their partition function form under uncertainty. In addition, we introduced a novel hybrid class of cooperative games, HUGs, that couples hedonic preferences with utility ones. We extended several traditional stability concepts to the HUGs setting, via equipping them with the ability to cope with both utility and hedonic preferences; and proceeded to propose IRIS, a novel, HUGs-specific solution concept that combines the key notions of individual stability and individual rationality, and studied its computational properties. We then provided an instantiation of HUGs, along with a definition for characterizing a coalition's feasibility, and used it to prune the coalitional space in order to compute kernel-stable payoffs for IRIS partitions. Last but not least, we provided a probability bound for pruning coalitions in HUGs.

6.2 Future Work

There are several possible extensions to this thesis.

First, there is definitely more work to be done regarding the uncertainty in hedonic games. For instance, it would be interesting to investigate even more learning models, and evaluate them as more or less appropriate for learning agent preferences. In addition, it would be possible to examine even more classes of hedonic games; for instance, we could evaluate the PTMs' performance using classic ASHGs, ASHGs-PFF, BHG-PFF, etc. Further experimental work could involve 'tuning' the models we have already studied, in order to better fit our problem. Moreover, it would be interesting to examine uncertainty under a dynamic environment, that is, with new agents arriving over time, and/or preference relations altering during interactions.

Second, our proposed generalization of hedonic games into partition function form opens the way for their thorugh theoretical study. Specifically, it would be interesting to study stability concepts within hedonic games in partition function form, and examine the computational aspects of this extension.

Last but not least, our novel Hedonic Utility Games opens yet another path in the theoretical study of practical cooperative game settings. In fact, the study of the computational aspects of HUGs has just began with this thesis. One potential endeavour, interesting from both a theoretical and a practical standpoint, would be the examination of the convergence properties to an existing point in IRIS($\mathcal{G}$), and the systematic evaluation of the corresponding convergence rate in specific settings. In addition, one could devise ways to identify approximately IRIS-stable partitions. Finally, it would be interesting to examine HUGs under uncertainty.

Bibliography

[Aumann et al., 1965] Aumann, R. J., Peleg, B., and Robinowitz, P. (1965). A method for computing the kernel of n-person games. *Mathematical Computation*, 19:531–551.

[Aziz and Brandl, 2012] Aziz, H. and Brandl, F. (2012). Existence of stability in hedonic coalition formation games. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems - Volume 2*, AAMAS '12, pages 763–770. International Foundation for Autonomous Agents and Multiagent Systems.

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

[Aziz et al., 2016a] Aziz, H., Harrenstein, P., Lang, J., and Wooldridge, M. (2016a). Boolean hedonic games. In *Proceedings of the 15th International Conference on Principles of Knowledge Representation and Reasoning*, KR'16, pages 166–175. AAAI Press.

[Aziz et al., 2016b] Aziz, H., Savani, R., and Moulin, H. (2016b). *Hedonic Games*, page 356–376. Cambridge University Press.

[Ba and Kingma, 2015] Ba, J. and Kingma, D. (2015). Adam: A method for stochastic optimization. In *Proc. of the 3rd International Conference on Learning Representations (ICLR-15)*.

[Ballester, 2004] Ballester, C. (2004). Np-completeness in hedonic games. *Games and Economic Behavior*, 49(1):1 – 30.

[Bergstra et al., 2013] Bergstra, J., Yamins, D., and Cox, D. D. (2013). Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures.

[Bergstra et al., 2011] Bergstra, J. S., Bardenet, R., Bengio, Y., and Kégl, B. (2011). Algorithms for hyper-parameter optimization. In *Proc. of NIPS-2011*, pages 2546–2554.

[Bishop, 2006] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg.

[Bistaffa et al., 2017] Bistaffa, F., Farinelli, A., Chalkiadakis, G., and Ramchurn, S. D. (2017). A cooperative game-theoretic approach to the social ridesharing problem. *Artificial Intelligence*, 246:86 – 117.

[Blei, 2012] Blei, D. M. (2012). Probabilistic topic models. *Commun. ACM*, 55(4):77–84.

[Blei et al., 2003] Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, 3:993–1022.

[Bogomolnaia and Jackson, 2002] Bogomolnaia, A. and Jackson, M. O. (2002). The stability of hedonic coalition structures. *Games and Economic Behavior*, 38(2):201 – 230.

[Bogomolnaia et al., 2005] Bogomolnaia, A., Moulin, H., and Stong, R. (2005). Collective choice under dichotomous preferences. *Journal of Economic Theory*, 122(2):165 – 184.

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

[Chalkiadakis et al., 2011] Chalkiadakis, G., Elkind, E., and Wooldridge, M. (2011). *Computational Aspects of Cooperative Game Theory*. Synthesis Lectures on Artificial Intelligence & Machine Learning. Morgan & Claypool Publishers, San Rafael.

[Davis and Maschler, 1965] Davis, M. and Maschler, M. (1965). The kernel of a cooperative game. *Naval Research Logistics Quarterly*, 12(3):223–259.

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

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

[Georgara et al., 2019a] Georgara, A., Ntiniakou, T., and Chalkiadakis, G. (2019a). Learning hedonic games via probabilistic topic modeling. In Slavkovik, M., editor, *Multi-Agent Systems*, pages 62–76, Cham. Springer International Publishing.

[Georgara et al., 2019b] Georgara, A., Troullinos, D., and Chalkiadakis, G. (2019b). Extracting hidden preferences over partitions in hedonic cooperative games. In *Proc. of the 12th Knowledge Science Engineering Management (KSEM 19).*

[Hoffman et al., 2010] Hoffman, M. D., Blei, D. M., and Bach, F. (2010). Online learning for latent dirichlet allocation. In *Proceedings of NIPS*, NIPS'10, pages 856–864, USA.

[Igarashi and Elkind, 2016] Igarashi, A. and Elkind, E. (2016). Hedonic games with graph-restricted communication. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, AAMAS '16, Richland, SC.

[Jordan, 1999] Jordan, M. I., editor (1999). *Learning in Graphical Models*. MIT Press, Cambridge, MA, USA.

[Jordan et al., 1999] Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., and Saul, L. K. (1999). An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233.

[Kendall, 1948] Kendall, M. (1948). *Rank correlation methods*. Griffin, London.

[MacQueen et al., 1967] MacQueen, J. et al. (1967). Some methods for classification and analysis of multivariate observations. In *Proc. of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA.

[Mamakos and Chalkiadakis, 2017] Mamakos, M. and Chalkiadakis, G. (2017). Probability bounds for overlapping coalition formation. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI'17, pages 331–337. AAAI Press.

[Mamakos and Chalkiadakis, 2018] Mamakos, M. and Chalkiadakis, G. (2018). Overlapping coalition formation via probabilistic topic modeling. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-18)*, pages 2010–2012.

[Michalak et al., 2010] Michalak, T., Rahwan, T., Marciniak, D., Szamotulski, M., and R. Jennings, N. (2010). Computational aspects of extending the shapley value to coalitional games with externalities. *Frontiers in Artificial Intelligence and Applications*, 215.

[Michalak et al., 2008] Michalak, T. P., Dowell, A., McBurney, P., and Wooldridge, M. (2008). Optimal coalition structure generation in partition function games. pages 388–392.

[Michalak et al., 2009] Michalak, T. P., Rahwan, T., Sroka, J., Dowell, A. J., Wooldridge, M., McBurney, P., and Jennings, N. R. (2009). On representing coalitional games with externalities. In *EC*.

[Mitzenmacher and Upfal, 2005] Mitzenmacher, M. and Upfal, E. (2005). *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press.

[Oh et al., 2015] Oh, K.-K., Park, M.-C., and Ahn, H.-S. (2015). A survey of multi-agent formation control. *Automatica*, 53:424 – 440.

[Osborne, 2004] Osborne, M. J. (2004). *An Introduction to Game Theory*. Oxford University Press.

[Osborne and Rubinstein, 1994] Osborne, M. J. and Rubinstein, A. (1994). *A course in game theory*. MIT press.

[Pedregosa et al., 2011] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

[Peters, 2016] Peters, D. (2016). Complexity of hedonic games with dichotomous preferences. In *AAAI*.

[Ruder, 2016] Ruder, S. (2016). An overview of gradient descent optimization algorithms.

[Russell and Norvig, 2009] Russell, S. J. and Norvig, P. (2009). *Artificial Intelligences: a modern approach - 3rd Edition*.

[Saad et al., 2012] Saad, W., Han, Z., Zheng, R., Hjorungnes, A., Basar, T., and Poor, H. V. (2012). Coalitional games in partition form for joint spectrum sensing and access in cognitive radio networks. *IEEE Journal of Selected Topics in Signal Processing*, pages 195–209.

[Schmeidler, 1969] Schmeidler, D. (1969). The nucleolus of a characteristic function game. *SIAM Journal on Applied Mathematics*, 17(6):1163–1170.

[Shehory and Kraus, 1999] Shehory, O. and Kraus, S. (1999). Feasible formation of coalitions among autonomous agents in non-super-additive environments.

[Shoham and Leyton-Brown, 2008] Shoham, Y. and Leyton-Brown, K. (2008). *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press.

[Skibski et al., 2015] Skibski, O., Michalak, T. P., Sakurai, Y., Wooldridge, M., and Yokoo, M. (2015). A graphical representation for games in partition function form. In *AAAI*.

[Sliwinski and Zick, 2017] Sliwinski, J. and Zick, Y. (2017). Learning hedonic games. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 2730–2736.

[Stearns, 1968] Stearns, R. E. (1968). Convergent transfer schemes for n-person games. *Transactions of the American Mathematical Society*, 134(3):449–459.

[Taywade et al., 2019] Taywade, K., Goldsmith, J., and Harrison, B. (2019). *Decentralized Multiagent Approach for Hedonic Games: 16th European Conference, EUMAS 2018, Bergen, Norway, December 6–7, 2018, Revised Selected Papers*, pages 220–232.

[Teh et al., 2004] Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. (2004). Hierarchical dirichlet processes. *Journal of the American Statistical Association*, 101.

[Wooldridge, 2009] Wooldridge, M. (2009). *An introduction to multiagent systems- Second Edition*. John Wiley & Sons.

[Zhao, 1991] Zhao, J. (1991). The equilibria of a multiple objective game. *International Journal of Game Theory*, 20(2):171–182.