EMYS: a social robot that plays “Sueca”

Filipa Isabel Nogueira Correia

Thesis to obtain the Master of Science Degree in

Information Systems and Computer Engineering

Supervisors: Prof. Dr. Ana Maria Severino de Almeida e Paiva
Prof. Dr. Francisco António Chaves Saraiva de Melo

Examination Committee

Chairperson: Prof. Dr. João Emílio Segurado Pavão Martins
Supervisor: Prof. Dr. Ana Maria Severino de Almeida e Paiva
Member of the committee: Dr. Manuel Fernando Cabido Peres Lopes

November 2015

Acknowledgments

I would like to thank my advisors, Professor Ana Paiva and Professor Francisco Melo, for their invaluable experience, expertise and guidance provided throughout the development of my thesis. Their patience, optimism and wise advices were always of the utmost importance and kept me going and striving to accomplish the best possible outcome.

Additionally, I would like to thank my colleagues in the research project PArCEIRO, particularly Tiago Ribeiro and Patrícia Alves-Oliveira, for all their valuable insights and prompt guidance.

Lastly, I am grateful to my family, in particular to my parents, who have always believed in me, and to my closest friends, who have always supported me.

Abstract

The computational complexity of some card games attract the interest of Artificial Intelligence (AI) researchers. Their main challenge is to deal with hidden information, nonetheless recent approaches start to overcome this problem, such as Monte-Carlo Methods. On the other hand, the strong social component every multi-player game presents can also be included in an artificial player through an embodied agent that interacts with other players. Therefore, this thesis proposes the development of a social *Sueca* player that is able of both playing the game and communicate with human players, enhancing their game experience. This agent includes an AI module able of deciding which card to play, based on Perfect Information Monte-Carlo (PIMC) algorithm. Furthermore, in order to be socially present during the game, this agent also contains a decision maker module able of evaluating the game state and producing adequate verbal or non-verbal behaviours. Finally, user studies revealed significant comparisons to human players that encourage future development of this work.

Keywords

Artificial Intelligence, Trick-taking Card Game, Hidden Information, Interactive Companions, Socially Intelligent Behaviour

Resumo

A complexidade computacional de alguns jogos de cartas atrai o interesse de investigadores na área da Inteligência Artificial. Apesar do maior desafio ser a informação escondida, já existem algumas abordagens capazes the ultrapassar este problema, tais como metodologias baseadas em Monte-Carlo. Por outro lado, a componente social que os jogos com multi-jogadores apresentam é bastante forte e pode, no entanto, ser incluída num jogador artificial através de robôs que interagem com os outros jogadores. Deste modo, esta tese propõe o desenvolvimento de um jogador social de Sueca que é capaz de jogar o jogo enquanto comunica com jogadores humanos, melhorando a experiência do jogo. Este agente incluiu um módulo de IA capaz de decidir que carta jogar, com base no algoritmo *Perfect Information Monte-Carlo*. Para além disso, de maneira a estar socialmente presente durante o jogo, este agente também contém um módulo de decisão capaz de avaliar o estado do jogo e de produzir comportamentos verbais e não verbais adequados. Por fim, estudos com utilizadores revelaram comparações significativas com jogadores humanos, incentivando o futuro desenvolvimento deste trabalho.

Palavras Chave

Inteligência Artificial, Jogos de cartas, Informação escondida, Companheiro Interactivos, Comportamentos socialmente inteligentes

Contents

List of Figures

List of Tables

Glossary

EMYS EMotive headY System

HRI Human-Robot Interaction

AI Artificial Intelligence

QA Question Answering

MCTS Monte-Carlo Tree Search

UCT Upper Confidence Bounds for Trees

PIMC Perfect Information Monte-Carlo

ISMCTS Information Set Monte-Carlo Tree Search

IIMC Imperfect Information Monte-Carlo

GIB Ginsberg's Intelligent Bridgeplayer

PIPMA Perfect Information Post-Mortem Analysis

CSP Constraint Satisfaction Problem

FGR Favourable Games Rate

1

Introduction

Games have been a subject of particular interest to the AI field over the years, and the reason for that is the complexity of computationally solving them. From board games to card games, or even role-playing games, the goal of these computer programs is to create rational agents capable of evaluating the game and achieving the best possible outcome. Three remarkable artificial players that raised the bar for developing this kind of agents were: Deep Blue, the first artificial chess player that defeated a human world champion in 1997 [5]; Chinook, a checkers program that proved the game leads to a draw with two optimal players [34]; and Watson, the Question Answering (QA) system that beat the two highest ranked Jeopardy players in 2011 [10].

However, different games introduce different challenges due to their properties and some of them varying complexity. For instance, most card games add to board games two properties: unknown information (hidden cards) and the element of chance. As a result, AI researchers have been dealing with card games in recent years, and some card games remain unsolved even today. The game of Poker illustrates this idea since most AIs still have to deal with limited versions of the game[42]

Another point is the social component present in most games, specially multi-players. The dynamics of these games are strongly attached to players' interactions and can, therefore, enhance the game experience. Hence, the artificial players previously mentioned can evolve to another level of interaction during the game. In other words, a certain artificial player for a specified game can be adapted and integrated into an embodied agent to play while interacting with other players. This kind of embodied agents can either be virtual entities or physical robots, and the study of their interactions with humans belongs to the Human-Robot Interaction (HRI) field. Some agents of this nature illustrate this idea, such as the iCat chess tutor [19] and the EMotive headY System (EMYS) Risk player [27]. These last two examples explore different challenges from an HRI point of view: the iCat has the role of tutoring while targeting young population; EMYS plays as an opponent.

These examples have inspired the idea of creating a card game scenario where an embodied agent plays with human players. Considering some card games are still unsolved challenges for AI, and also trying to bring relevant achievements for HRI, the game of *Sueca* seems to meet all these requirements. It is a Portuguese card game, known in Portugal and Brazil across many age groups, especially the elderly. Since the four players are divided into two teams, each one has two opponents and one team companion. These two roles together have not yet been studied in an artificial embodied game player.

Another advantage of exploring this scenario is that it reaches a diverse audience that includes the elderly, which is an increasingly important point considering the world population is ageing dramatically. The elderly have specific needs, physical and cognitive, that are often not considered in the way we, as a society, conduct our lives. Some of these concerns are recently being solved with the help of technology which may range from computer programs to intelligent robots. However, existing technology with elderly purposes is commonly focused on health care, and when dealing with aged people with no serious health problems, that are still capable of doing their regular daily tasks, occupying their free time with leisure activities is also a necessity for their well-being. Therefore, the social robot this project aims to create might be used for further studies with elder care purposes.

Thesis problem

The challenge this thesis proposes is the development of a social agent, which, embodied in an expressive robotic entity, can efficiently play *Sueca* with and against human players whilst interacting with them throughout the game.

Contributions

The first contribution this thesis presents is an artificial intelligent *Sueca* player using the Perfect Information Monte-Carlo algorithm. Secondly, it also introduces a social *Sueca* agent, capable of interacting with human players according to the game state. Furthermore, this social robotic partner proved to be comparable to human partners in many different measures, and also to positively change other players' affect after the game.

*

The next chapter presents some background research that helps the reader understand the problems further mentioned, Chapter 2. The report proceeds with the state-of-art of playing card-games and human-robot-interaction, Chapter 3. Additionally, Chapters 4 addresses the artificial intelligent *Sueca* player. Moreover, a user-centred study is revealed in Chapter 5, preceding the implementation of the social agent in Chapter 6 and its corresponding results in Chapter 7. Finally, it presents the conclusions and future work, Chapter 8.

2

Background

Contents

The current chapter introduces the discussion of relevant research in order to understand some concepts and terminology further mentioned.

2.1 Game theory concepts

Game theory studies decision making problems involving multiple decision makers. A problem of this nature is usually called a game and defines a set of constraints to the players' actions. It also studies the strategies these players might take and the properties of each game.

Each decision maker tries to maximise the payoff/reward of his possible actions and one possible approach to do that is to consider the opponents' actions. The Nash-equilibrium [24] of a game is a stable strategy for every player and occurs when each player chooses the best strategy for himself, considering their opponents have the same behaviour. Moreover, each player cannot have a better benefit by changing his strategy unilaterally.

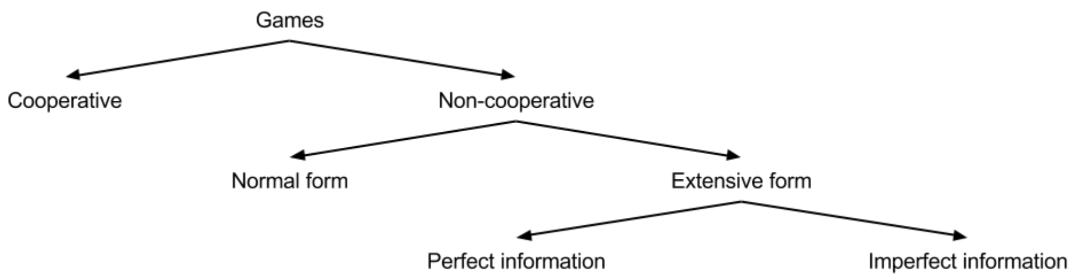


Figure 2.1: Hierarchy of games

Figure 2.1 shows how games can be hierarchically categorised [25]. In a cooperative game, players cooperate with one another in order to achieve a common goal. Alternatively, in a non-cooperative game, each player works independently for its own purposes. Non-cooperative games can also be branched into two forms: normal and extensive form [35]. A normal form game can be defined as the tuple $(N, (A_k)_{k=1}^N, (u_k)_{k=1}^N)$, where:

- N is the number of players;
- A_k is the finite set of available actions for the k-th player;
- u_k is the payoff for the k-th player.

Additionally, considering the players' payoffs, another relevant concept is the zero-sum game, where the sum of all players' payoffs is zero. For instance, in a zero-sum 2 players game, $u_1 = -u_2$. Although the normal form games assume that players' actions are made simultaneously, in the extensive form games, the players' actions are sequential. This evidence leads to another branching in the hierarchy of games and, consequently, an extensive game can be considered as a perfect information and an imperfect information game. In perfect information games, each player knows exactly the real state of his opponents, (e.g. Chess). In imperfect information games, the game state is not fully observable to the players. For instance, in a Poker game, a player only knows its own cards, the cards in the table, and the bets of all players.

In imperfect information games, an *information state* or *information set* for a player k corresponds to a set of all games states that yield the same "observation" to player k. For example, in a Poker game, an information set consists in all game-states that lead to the same observed cards in the table and in the player's hand.

All the concepts previously defined will be further mentioned in order to describe the *Sueca* game characteristics, and also in context of some presented algorithms.

2.2 The game of *Sueca*

Sueca is a card game categorised as trick-taking, which means the game has a finite number of rounds, called tricks. In this case, there are ten tricks, since the deck has forty cards equally distributed among the four players. This game uses the standard French card deck, excluding the rank 8 through 10. Although most trick-taking card games count the number of winning tricks to determine the winner, *Sueca* assigns points to the cards, according to Table 2.1. The most significant difference, compared to other games, is the card with rank 7 being higher than the King (K) and lower Ace (A).

All valued cards sum 120 points, which means a team with more than 60 points wins the game. Moreover, each player is paired with the player in front of him, and the two adjacent players form the opposing team. Hence, the game involves both cooperation and competition.

Table 2.1: Rank of cards per suit and respective reward values

Cards	2-6	Q	J	K	7	A
Points	0	2	3	4	10	11

After the deck has been shuffled and divided, the dealer chooses the top or bottom card to be the trump suit, leaves it on the table, and distributes the remaining cards among all players. The remaining rules are quite similar to any other trick-taking games:

- Follow the suit of the first card played in the turn (lead suit), if possible;
- A player wins the trick if his card has the highest value belonging to the lead suit or the trump suit.

Sueca is a nondeterministic game, since it includes what is called the element of chance by the cards being dealt randomly at the beginning. Additionally, since the cards of each player are hidden from the other players, this is considered as an imperfect information game. There are almost 1.9×10^{22} possible card distributions[1].

[1] $4 \times {}^{40}C_{10} \times {}^{30}C_{10} \times {}^{20}C_{10}$

3

Related work

Contents

This chapter presents the state of the art related to this work. Since no relevant studies on *Sueca* have been found, the research is focused on algorithms used in similar card games. It also presents existing companion and game-player robots that will allow the analysis on human-robot interaction relevant for this work.

3.1 AI in games

AI has been solving many games over the years, however, the definition of "games" usually refers to zero-sum and perfect information games. These kind of games are commonly solved by creating a tree representing all possible states and searching for the optimal, or a nearly optimal, solution. The greatest achievements related to perfect information games are generally based on finding good heuristics to refine the search and also good prunings to reduce the search space. Deep Blue can exemplify this idea [5], it uses an iterative-deepening alpha-beta search and the key of its success is mostly the null move heuristic and the futility pruning. Another example is Chinook, also a perfect information game that was solved using alpha-beta search [34].

Nevertheless, *Sueca* is considered an imperfect information game, as described in Section 2.2, and this class of games is usually solved by one of three different approaches [7]. The first one, and the most popular, is based on Monte-Carlo Methods. Then, another possible approach is trying to compute a Nash equilibrium strategy or an approximation thereof. Lastly, belief distributions involving game state inferences and opponent models can also be used. The first two mentioned approaches are mutually exclusive, while the last one can be used as a supplement. The following two subsections detail how Monte-Carlo Methods and Belief Distributions can be applied in hidden information games. The second pointed approach will not be addressed due to the imposed limitations of our domain, considering, for instance, that the maximum known number of states for computing a Nash equilibrium is 10^{12} [42], which is much lower than the number of possible states in a *Sueca* game.

3.1.1 Monte-Carlo Methods

The popularity and acceptance of Monte-Carlo based methods have increased since its success on Bridge. Ginsberg's Intelligent Bridgeplayer (GIB)[1] was the first computer bridge champion using Monte-Carlo Methods, and subsequently, another two successful domains were Skat[2] and Computer Go [14]. Since some of these domains remain a challenge for traditional AI techniques, this method seems to be very promising.

In order to solve a hidden information game, the first challenge is to deal with information sets. The most used approach to solve it is determinization, which samples choice nodes instead of

[1] http://www.gibware.com/
[2] https://skatgame.net/

considering all of them in an unique set. The combination of this approach to Monte-Carlo Methods is known as PIMC. For instance, in a card game scenario, each iteration of PIMC samples the cards distributions for all players and the simulation process of the game behaves as a perfect information game. In other words, during the simulation each player makes decisions as if his opponents' cards are visible. The first successful implementation of this technique was GIB [15].

In 1998, Frank and Basin [11] produced an analysis on PIMC's limitations. They identified two distinct problems: *strategy fusion* and *non-locality*. Due to the repeated minimaxing architecture that PIMC has and its evaluation of possible distributions with the best strategy, applying this knowledge, when information is missing, might produce suboptimal decisions. This is called the *strategy fusion*. For instance, when having a move with a guaranteed reward and another move with a possible reward of the same value although depending on the current world, PIMC equally considers both moves.

The second problem, *non-locality*, results from the propagation of values. The value of a game tree node only considers its children' values, however, in an imperfect information game, some guesses might be done using values of the non-local subtree. For instance, considering 2 different worlds, the player 1 can guarantee a winning trick in the world 1 by making a certain move, and if in that state, he makes another move instead, player 2 might assume they are in world 2. PIMC cannot make such an inference.

Despite the satisfying outcomes of PIMC, there were still difficulties in understanding the strong results of this algorithm. As such, Long et al. have analysed the previously mentioned problems of PIMC search, and they have shown how three different properties of a game can influence the success of PIMC [23]. The first property is *leaf correlation*, which refers to how likely it is to affect a player's payoff in the neighbourhood of a leaf. When the probability of all siblings having the same payoff values is higher, the correlation value increases. Secondly, *bias* indicates the chance of a player being preferred over another. Finally, the last game characteristic that has been pointed is *disambiguation factor*, that denotes how rapidly the hidden information is revealed.

These properties have been tested in a set of experiments in both PIMC and a random player against an optimal Nash equilibrium player. Results shown the performance of PIMC increases as the correlation value is higher, bias does not considerably affect its success, and, finally, disambiguation has the greatest impact on the results of the algorithm. When this last value is higher, it means the game turns more quickly into a perfect information game. Additionally, the authors demonstrate these properties on real game examples, such as Skat and Kuhn poker. Skat indicates a considerably good performance of PIMC, due to its values of *leaf correlation*, *bias*, and *disambiguation factor*. Since Skat presents strong similarities to *Sueca*, it is expected that PIMC also has a good performance when applied to *Sueca*.

Cowling et al. have also investigated the application of Monte-Carlo Tree Search (MCTS)[3]

[3]MCTS algorithm builds a search tree according to the results of previous iterations by sampling unknown informations.

to hidden information games [7]. Their research supports a new descendant family of algorithms, Information Set Monte-Carlo Tree Search (ISMCTS). ISMCTS works with information sets, instead of game states and uses determinization to sample the game, however producing a single tree. The main advantages are the computational budget efficiency and the fact of suffering less from *strategy fusion* than PIMC. The authors also presented some experiments in three different games, including a card game. Their results on the card game Dou Di Zhu were very similar to Upper Confidence Bounds for Trees (UCT) and did not introduce any improvement to the playing strength. The authors explained these results with the high branching factor this domain produces, which has discouraged the usage of this technique on the domain of *Sueca*, since in the information set tree, the initial branching factor would also be high (10^8, ${}^{40}C_{10}$).

Recently, Furtak & Buro [13] presented a new search algorithm called Imperfect Information Monte-Carlo (IIMC) that can be suitably applied to hidden information games and reduces the *strategy fusion* problem. During the simulation phase, each player's move is chosen inside a player's module and the game behaves as an imperfect information due to this encapsulation. Additionally, the players' modules allow the differentiation of players using different strategies. The authors revealed the great potential of this approach when applied to trick-based card games, considering it has been successfully tested in the Skat scenario.

Table 3.1: Advantages and disadvantages of the mentioned Monte-Carlo algorithms

Algorithm	**Advantages**	**Disadvantages**
PIMC	Offline Computation Easy to parallel	Strategy fusion Non-locality
ISMCTS	Offline Computation Easy to parallel Computational budget	Strategy fusion (less than PIMC) Non-locality Complexity
IIMC	Offline Computation Easy to parallel Allow a different player model per player	Strategy fusion (less than PIMC) Non-locality Complexity

The advantages and disadvantages of the mentioned MCTS variations for imperfect informations games are clearly summarised in Table 3.1. Both of three techniques are easy to parallel and allow an offline computation. However, ISMCTS uses the computational budget more efficiently than the other two techniques, and IIMC allows different player models per player. Disadvantages show all the three techniques have the *non-locality* and *strategy fusion* problem, although *strategy fusion* is lower in ISMCTS and IIMC.

Overall, the chosen approach for the *Sueca* domain is PIMC, since it will be used at runtime and this algorithm provides the lightest computational burden to do that.

3.1.2 Game State Inference & Opponent Modelling

While discussing imperfect information games, belief distributions, game state inference and opponent modelling are other relevant subjects to consider. Predicting some of the opponents' cards or other clues would be beneficial to select better actions at each state of the game. Additionally, inferring hidden information, while using a Monte-Carlo based method, can also decrease the *non-locality* problem [7].

Buro in 2009 [4] presented his work on state evaluation and inference that has been included in his Skat player. His approach combines two techniques, one for evaluating the bidding and another for selecting hypothetical worlds during the game play. The former technique uses logistic regression to evaluate the winning probability of each hand and it has 22 million Skat games as data base. This winning probability determines the strength of a hand and can, therefore, be used on the bidding.

The second technique is mainly based on two heuristics. Fastest-cut-first search heuristic evaluates each move according to its beta-cutoff value and minimises the expected number of visited nodes. Additionally, in order to reduce the tree exploration, another heuristic groups cards by their strength value and considers, for example, 7♣ and 8♣ the same move, when holding both cards in a player's hand. The author compares his work to other similar ones and concludes the strength of his techniques lies in two central points. First, determining the $P(\text{world}|\text{move})$ on offline data, instead of doing it in runtime. Sencond, his formulation is generalised in a way that it is possible to perform it on high-level features. Since the main difference between *Sueca* and Skat is that the first one does not have the bidding phase, Buro's first technique would not be appropriate for the *Sueca* game. However, the search enhancements could be suitably applied, considering the game trees are identical.

Usually, opponent modelling uses optimal strategies to predict the other players' actions and these models tend to be overly defensive. Consequently, Long & Buro in 2011 [22] suggested a post-processing analysis that is able to infer opponent's qualities based on their decisions in a certain environment. The main idea is to classify each opponent with a mistake rate and use that value to be more or less defensive. This approach, called Perfect Information Post-Mortem Analysis (PIPMA), computes a procedure after each game episode (in a trick-taking card game, it would be after each trick) to incrementally update the mistake rate of each opponent. The authors made some experiments in a Skat player with very good results, where they used the mistake rate to adjust the bidding behaviour during the game. Despite the fact that *Sueca* does not have the bidding phase, classifying opponents with a mistake rate can useful to other purposes. As a result, it would be interesting to model the opponents in a similar way in the domain of *Sueca*, in order to make better decisions or even for the embodied agent to produce adequate behaviours.

Another highly suitable card game to make opponent models is Poker, since predicting the players' moves can naturally affect the outcome of this game. In order to predict players' cards and their future actions, Posen et al. in 2010 [29] have investigated this subject. They proposed an opponent model that starts with a prior distribution and changes over time with a differentiating function. The prior distribution allows it to make reasonable inferences while having insufficient information. In addition, the relational probability tree algorithm TILDE builds a decision tree with the stored samples of a player. This decision tree represents the differentiating function that will adapt the initial prior distribution. Besides this opponent model, the authors explain how to integrate this function with MCTS. Instead of sampling the cards randomly, MCTS uses card predictions and, therefore, the algorithm does not need a numerous amount of iterations to reach a uniform card distribution. Furthermore, the probabilities of action predictions are used in the selection phase of the MCTS, according to the state of the game and the sampled cards. Since MCTS can be used in the *Sueca* domain, a similar opponent model can also improve the capabilities of this algorithm, as shown in Poker.

Table 3.2: Techniques signed with ×, ✓ and ∼ symbols are, respectively, not suitable, suitable and conditionally suitable to the *Sueca* domain.

Tested domain	**Technique**	**Goal**	**Suitable to *Sueca***
Skat	Determine the winning probability of a hand	Improve the bidding	×
	Fastes-cut-first heuristic	Order moves	✓
	Considering similar states equally	Reduce tree exploration	✓
	Calculate the mistake rate of each player	Improve the bidding	∼
Poker	Opponent model	Improve MCTS policies	✓

Table 3.2 summarises what techniques have been reviewed, their purposes, and, finally, if they can be applied to the *Sueca* domain. The technique of determining the winning probability cannot be used for the exact same purpose, since our domain does not include a bidding phase. The next two search enhancements can naturally be used due to the similarities between Skat and our domain game trees. The mistake rate was signed as conditionally suitable because it can also be used, although with a different purpose. Thinking in the embodied agent of our work, it can assign a mistake rate variable to each player and produce appropriate behaviours according to their values. A similar approach might be thought to use the winning probability, however, opponents' hands are not visible and the agent should not reveal its own information. The last technique also can be an addition to the MCTS base policies.

3.2 Human-robot interaction

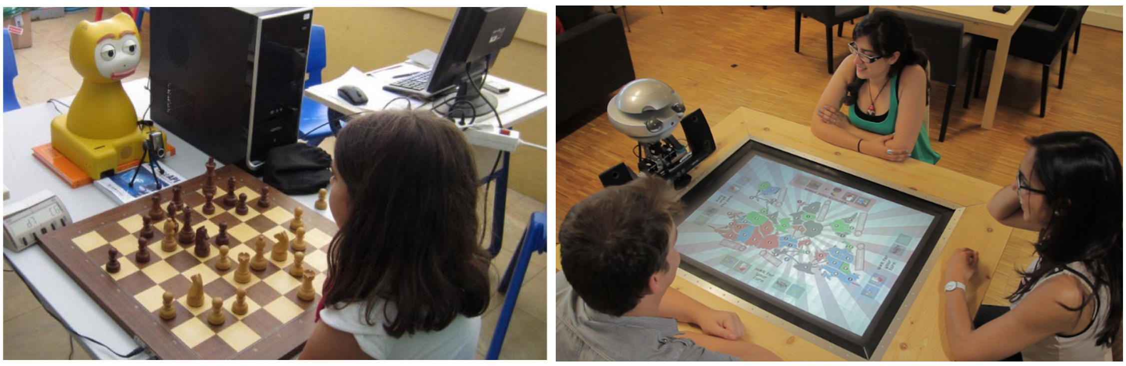
Regarding the goals of this project, it is crucial to investigate and evaluate the state of the art of HRI, in particular in the context of robot companions or players. On one hand, the idea is to understand how social agents have been integrated into games. On the other hand, to investigate the gaps in existing robots with an elderly care purpose. The next subsections will address these points.

3.2.1 Social robots in games

The idea of entertainment robots is expanding and becoming more frequent. Its general goal is to create a social robot to interact with humans through a specific entertainment activity. These activities should be lifelike experiences providing pleasure and enjoyment feelings. Depending on the target audience, they can also be included in more challenging or even pedagogic activities.

Leite et al. uses the *iCat* robot in a chess game scenario with children [6, 20, 21]. This chess companion also has the role of a tutor due to the help it provides during the game, for instance, it expresses opinions about children' moves so that they can improve their chess skills. After their first pilot studies, the authors revealed the need of including social and cognitive abilities, commonly referred as empathy. Their further studies introduced into the *iCat* affect recognition in order to improve the robot's social cues. The way they address this point includes recognising users' expressions and considering others' affective states. For instance, when a child is losing, the *iCat* comments about his moves should not cause embarrassment. In addition, and considering their goals were also focused on long-term interactions, this chess player recognises faces and greet people mentioning past events.

This agent has some similarities and differences with the proposed agent of this work. On one hand, including empathic behaviour to robots usually leads to more engaging, natural and likeable experiences to users. On the other hand, the *iCat* in this scenario needs access to more details of users' emotional state because of its tutoring advices. Our *Sueca* player will not advise other players about their actions, instead it will comment the game state. Additionally, the target audience is clearly different and may lead to different concerns, and their work was also focused on long-term interaction.



(a) iCat - Chess tutor

(b) EMYS - Risk player

Figure 3.1: Companion robots in game playing scenarios.

Another example of a robot integrated into a game scenario is the Risk player by Pereira et al. [26]. The goal of their work was to create a robot that interacts with humans and is perceived as socially present in long-term interactions. Firstly, the authors presented how physical embodiments can provide interactivity and, therefore, cause the belief of social presence and improve face-to-face interactions. They also presented some guidelines in order to improve social presence and how they implemented them in the *EMYS* robot for the mentioned scenario [27]. In the Risk scenario, the agent produces non-verbal interactions through a gazing system and a speech direction detector, and it is capable of giving verbal feedback using a topology of speeches according to the game state. Moreover, the authors included an emotion or appraisal system that considers the values of some variables to improve the agent's behaviours, for instance, every event is rated with a relevance value and the robot only comments important moves. Another example is measuring the power of each player and, since Risk is about conquering and controlling, this power measure is used to shape the robot's mood and defining its strategy to play. Equally important are the simulation of social roles and the luck perception when rolling the dice. All the described behaviours were fully inspired by user studies.

Pereira's work is by far the most similar to the purposes of our goals. It demonstrates how to enrich the Risk game experience with a robot capable of social behaviours at a human level. The main difference from the proposed *Sueca* player is the game. Since no relevant user studies have been done with *Sueca*, applying the Risk' constraints to the *Sueca*'s scenario would lead to inconsistencies. However, an analogous approach might be taken, considering the domain data collection and the following development of the game player architecture.

3.2.2 Robots in elderly care

The greying of population is an undeniable demographic fact and, consequently, assisting the elderly in their daily living is a worrying subject. In order to address this concern, robots can be a valuable aid, however, considering the limitation of current robotic technology, their purposes are present in more specific tasks.

In 2009, Broekens et al. analysed and reviewed the most relevant literature about social robots in elderly care [3]. The authors categorised assistive robots for elderly as shown in Figure 3.2. The first division distinguishes social robots from nonsocial robots. The nonsocial ones are used for rehabilitation purposes and physical assistance, such as a smart wheelchair or an artificial limb, however, regarding the main purposes of this work, nonsocial robots will not be discussed. Social robots should be perceived as social entities due to their interaction with humans and can also be divided into two different sets, service type and companion type. The intersection of these two sets represents some of the robots that are used for both purposes and cannot be strictly categorised.

A well known social service robot is *Pearl* (Figure 3.3a), developed in the Carnegie Mellon University within the Nursebot Project [28]. This autonomous robot's duties are to guide the elderly through their environment, and to remind them about their daily activities, such as eating or taking their medicine. In other words, this functional assistant is capable of giving advice and

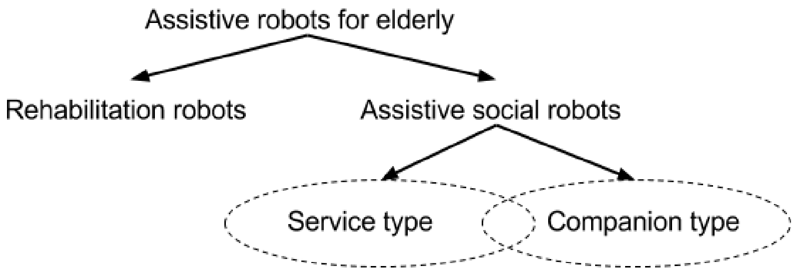


Figure 3.2: Categorization of assistive robots for elderly

providing cognitive support. When analysing *Pearl* through a more general AI point of view, this robot is equipped with many different technologies. Firstly, it has a speech recognition module and also has speech synthesis. Secondly, it has stereo camera systems and performs a fast image processing including face recognition. Lastly, *Pearl* also provides a navigation system and its body is touch sensitive.

Another two similar service robots are *RoboCare* [2] and *Care-O-bot II* [16]. They both are autonomous and provide indoor guidance to the elderly and, due to their advanced domotic components, strong planning, and scheduling frameworks, they can improve the independence of their owners. Since the aid these service type robots may grant to the elderly covers most of their daily basic activities, the involved concerns are amplified when compared to the proposed robot that plays a card game. These worries are reflected, for instance, in the extensive amount of sensors these robots should include.

(a) Pearl (b) Paro (c) Aibo

Figure 3.3: Service and companion robots for the elderly.

Paro is a seal shaped companion robot used as medical therapy for the elderly (Figure 3.3b). Since 2003, the work by Wada et al. provides a very good psychological and physiological evaluation of *Paro*'s effects on the residents of a care house [39–41]. This robot contains a behaviour generation system that provides proactive, reactive and physiological reactions, such as, poses or motions, looking at the direction of a sound, and sleeping. Their studies of both three weeks and one year have shown improvements in residents' moods, depression, stress levels, and social interactions with other residents. The goal of such a robot is fully inspired in animal-assisted treatments, which have studied benefits in humans' health. However, hospitals and health centres do not allow animals due to hygienic and safety reasons. Hence, researchers found a great opportunity to build similar robotic animals.

Another example of a purely companion robot is the *Huggable* [37], a teddy bear shaped covered of extremely sensitive touch sensors. The *Huggable* not only detects hard and soft touches, but also distinguishes between an object and a human touch. Considering experiments in an hospital, this robot was connected to a computer in the nurses' station and allowed the staff to access the sensory input data. Nurses could detect fear or insecurity by the way people hold the robot and provide appropriate assistance.

Purely companion robots in elderly care have only been applied to people with some kind of psychological or physiological disorder. As a result, these studies have distinct target audiences and also different concerns when compared to the purposes of our proposed embodied agent.

Aibo illustrates a robot that can be assigned to both the service type and the companion type (Figure 3.3c). It is considered by its creators as an entertainment type due to its puppy shaped body [12], and its appearance tries to maintain a lifelike experience to its owners. Tamura et al. started to study the acceptance and effects of this robot on elders with severe dementia [38]. Their study revealed a relevant increase of social actions, emotions and feelings of comfort about past memories.

Table 3.3: Robots for the aged population, their type and purposes

	Pearl	RoboCare	Care-O-Bot-II	Paro	Huegable	Aibo
Service type	✓	✓	✓			✓
Companion type				✓	✓	✓
Guidance	✓	✓	✓			
Advice	✓	✓	✓			
Therapy				✓	✓	✓

Table 3.3 groups all the previously mentioned robots and their purposes. This information strengthens the pertinence of our work for a senior audience, since existing robots for the elderly are focused on their physical and mental disabilities. Providing pleasuring activities for the aged population, that are still capable of reasoning, should also be a concern.

4

AI for *Sueca*

Contents

This chapter will describe the most relevant implementation details of the developed AI for *Sueca*, and also the effects of different parametrizations that have been tested.

4.1 Implementing PIMC

After thoroughly analysing state-of-the-art techniques to solve imperfect information games, and considering *Sueca* is, at this moment, computationally unsolved, the chosen approach was PIMC. Other presented techniques require computations that would be impractical to do at runtime, and therefore PIMC provide the best trade-off between computational resources and results for similar domains.

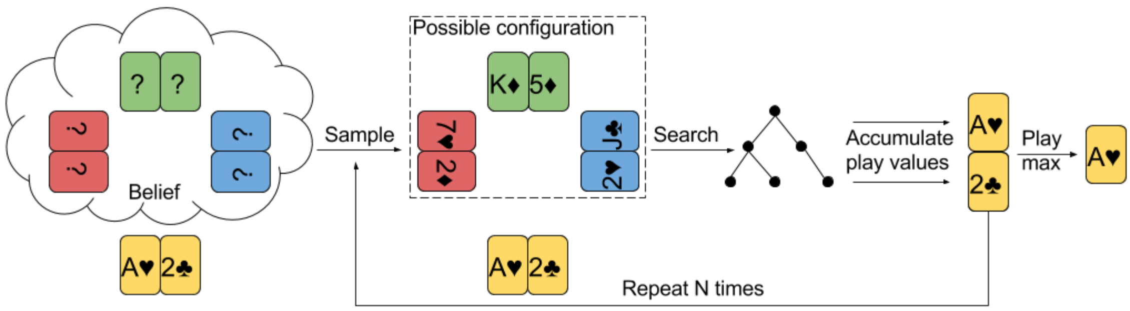


Figure 4.1: PIMC algorithm illustrated to exemplify the choosing procedure in the 8$^{\text{th}}$ trick

This algorithm samples cards distributions or configurations for other players' unknown hands. Then, it calculates the reward of playing each card in its own hand for every sampled distribution. The chosen card to play is, therefore, the one with the maximum accumulated reward for all the sampled distributions.

To implement this search technique, there are three key concepts or algorithms that require a full understanding: the Information Set, the PIMC and the MinMax Algorithm. Moreover, the encountered drawbacks are also further described, as well as implemented enhancements to overcome those limitations.

4.1.1 Information Set

An information set represents all the visible information during a game, and also inferred information based on certain events. The player must keep an instance of the information set per game and update it when necessary. It stores the known hand of the player and a deck with all the cards whose owner is unknown. As a result, each time another player plays a card, it should be removed from that deck.

The purpose of managing unplayed cards is to sample possible card distributions for the other three players with their real conditions. These sampled distributions will be used during the PIMC search and the closer they are to the real world, the better the search returning value will be. Additionally, the information set keeps track of suits per player and, when a player does not follow the leadsuit of a trick, it removes that suit from the player possible suits. By possessing this information, sampling possible distributions gets even closer to the real world, however it increases the complexity of the sampling process. The sampling method builds a Constraint Satisfaction Problem (CSP) where:

- variables are the unplayed cards;
- each domain is the set of players that still have that suit;
- and the constraints are the number of times a player can be assigned to a card.

4.1.2 PIMC Search

The following pseudo-code of the PIMC search algorithm guided the implementation.

Algorithm 1 PIMC search algorithm

```
1: procedure PIMC(InfoSet I, int N)
2:     for all m ∈ Moves(I) do
3:         val[m] = 0
4:     for all i ∈ {1..N} do
5:         x = Sample(I)
6:         for all m ∈ Moves(I) do
7:             val[m] += PerfInfoValue(x, m)
8:     return argmax_m {val[m]}
```

To recapitulate the main points of this algorithm, considering it can choose up to #Moves(I), it samples N possible card distributions for the other three players and calculates the reward of playing each possible move for the N sampled worlds. The returned move is the one that gave more accumulated reward.

The number of iterations this algorithm perform is imposed by the N parameter. Another version of the algorithm, instead of limiting the number of iterations, specifies the execution time of the main loop.

4.1.3 MinMax Algorithm

As mentioned above, PIMC has to calculate the reward of playing a card, for each sampled world (line 7 of Algorithm 8). Since a sampled distribution assigns the remaining cards to players, every game can be handled as a perfect information game. Therefore, to compute a perfect information game, considering each player or team intends to win, the MinMax algorithm was used.

MinMax is a popular algorithm for calculating optimal decisions in multiplayer games. Each node corresponds to a possible move by a player and their successors correspond to the possible moves of the next player. The player representing the agent and his team mate are both *max* players, likewise, the other two opponents are *min* players.

The complete game tree has 40 levels, from l_0 to l_{39}, and each group of l_{4n} to l_{4n+3} represents a trick. Additionally, since the utility value can only be determined in terminal nodes, these back-propagate their best or worst child utilities, if they are *max* or *min* nodes, respectively. The utility function to evaluate a sequence of moves deserved a serious consideration and is further detailed in Section 4.2.

4.1.4 Drawbacks and enhancements

When applying PIMC to decide which move m to make, the number of computed game trees is $\#\text{Moves}(I)$ times N (the number of different distributions), and the size of each game tree depends on the moves that are left to finish the game. Additionally, the algorithm tends to choose a near optimal decision as long as the N is reasonably large. As a result, this algorithm has to process a large number of nodes to make a proper decision, specially in the beginning of the game, and, without the enhancements further described, this task was impractical.

Russell and Norvig [32] suggest that MinMax performance can be improved using alpha-beta prunning, a move ordering heuristic, and a transposition table. Alpha-beta prunning, by simply storing the best choices so far for the max and min nodes, does not explore nodes that will not influence the final decision.

This technique can also be improved with a favourable ordering heuristic that produces earlier alpha-beta cuts. Therefore, the implemented ordering heuristic is dynamic and uses an auxiliary computation to decide how to order moves. Similarly to a human player reasoning, it analyses the current trick and tries to anticipate its winner. If this auxiliary procedure expects the winner to be one of its opponents, it orders the cards from the less valuable to the most valuable, otherwise, it does the opposite. This concept might bring a trade-off between the produced speed and the time spent on this extra computation plus the sort. However, alpha-beta cuts have reasonably increased and, therefore, significantly reducing the exploration time of the whole tree.

Another improvement with positive results on the MinMax exploration performance was a transposition table. Considering each card configuration or distribution will produce $\#\text{Moves}(I)$ game trees, they will contain sufficient similar subtrees to store their first computed values. Therefore, instead of recomputing them, they can be reused. This enhancement allowed the computation of a game tree with 7 depth levels in a reasonable time, which was not possible before. However, space inefficiency prevent the usage of this enhancement.

Furthermore, another heuristic was used, aiming to reduce redundancy in the state generation, suggested by Buro et al. [4]. When computing $\text{Moves}(I)$, two or more cards of the same suit, with consecutive ranks and with the same value can be considered as the same move, since they produce the same value. For instance, holding 3♣, 4♣ and 5♣ on the same hand will produce three equivalent states and therefore, this heuristic produce only one.

Limiting the maximum depth achieved by the MinMax algorithm was another available option, especially for earlier decisions that produce larger trees with impractical compute time. However, treating a non-terminal node as terminal may imply a new utility function or even the inclusion of a prediction. Different parametrizations related to the depth cut are further detailed in Section 4.2

4.2 Measuring parametrization effects

After implementing the PIMC search previously described, some tests were executed in order to observe the effects of different parametrizations. These tests had to be comparative and a baseline or benchmark was required to establish a standard measure.

4.2.1 Creating benchmarks

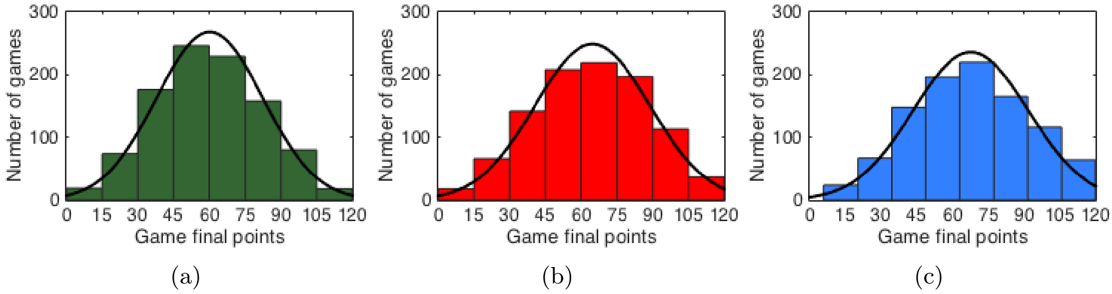
The baseline agent was called Rule-based and its main idea was to choose a move considering predefined rules, instead of using hard computational algorithms. It tries to roughly reproduce the reasoning of a non-professional human player.

Its procedure starts by collecting the highest cards of each allowed suit for the current play. The possibility of playing such a highest card is granted by two requirements: being the highest unplayed card of that suit; and not holding at least other 5 cards from that suit, except for the trump suit. Otherwise, this rule-based player return the lowest possible card.

The first experiments to test this baseline player compare three different scenarios:

- A - 1000 games with 4 Rule-based players [dark green];
- B - 1000 games with 1 Rule-based player and 3 Random players [red];
- C - 1000 games with 2 Rule-based players against 2 Random players [dark blue].

Each scenario has a corresponding colour that will be used in every chart for the same scenario.



(a) (b) (c)

Figure 4.2: Histograms of the final points obtained in 1000 games by: (a) one of teams in Scenario A; (b) the team with 1 Rule-based and 1 Random in Scenario B; (c) the team with 2 Rule-based in Scenario C

The histograms presented on Figure 4.2 exhibit the distribution of final points in 1000 games by one of the teams in each scenario. However, comparing the three histograms gets easier when merging the three fitting curves in one graph, Figure 4.3.

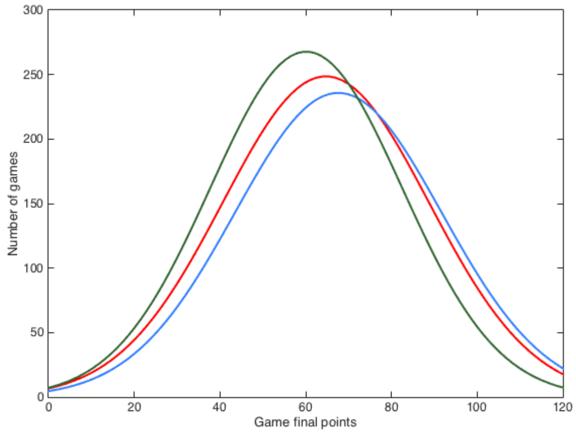


Figure 4.3: Fitting curves of the histograms presented on Figure 4.2 with the same colour scheme

In Scenario A, results were very balanced, as expected, because all players had the same deliberation process. In 1000 games, one of the teams obtained a winning percentage of 48.5%, a drawing percentage of 1.9% and a losing percentage of 49.6%. The Scenario B showed that a team with 1 Rule-based player and 1 Random player can beat a 2 Random players team with a winning percentage of 51.6%, a drawing percentage of 1.8% and a losing percentage of 41.6%. Finally, in Scenario C, the team with 2 Rule-based players beat the the 2 Random players with the highest winning percentage of 61.1%, a drawing percentage of 1.8% and losing percentage of 37.1%.

A player's performance can only be measured when playing with different players; otherwise, playing a considerable amount of games will balance the winning and losing rates, as seen in Scenario A. Additionally, theses results also demonstrated the impact of the team player on the team score, since having 2 Rule-based players in the same team increased the winning rate of the team with only 1 Rule-based player.

In addition to these conclusions, considering the opponents of both teams in Scenarios B and C have completely random procedures for playing the game, their winning rates were expected to be lower. A possible reason might be *Sueca*'s element of chance, which means certain hands can limit the result even though the opponents are Random players.

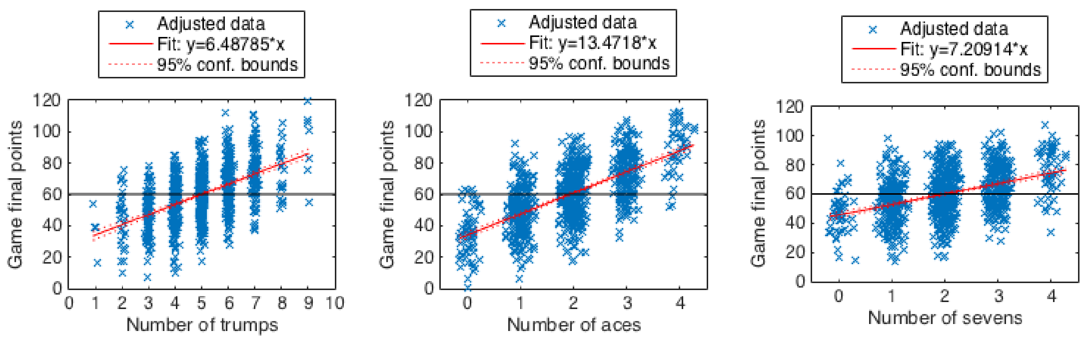
This idea incited some research on the influence of the players' initial conditions on the game result. On the one hand, the power of a hand is completely dependent on the playing style of each player, and therefore, using Random players to measure this property is inappropriate. On the other hand, this measure will not be used to carefully predict a hand's effect on the final result. Instead, its goal is to generally classify a hand in one out of three distinct categories (*hard*, *medium* and *easy*) and to filter the hands that are hardly or easily capable of winning. As a result, the chosen scenario to extract these categories' features was A.

In order to derive such characteristics, the first step is to speculate and collect possible features of the initial game conditions that may influence the final result. The next step is computing a linear regression on that data to decide the relevant features. In the first iterations of this process, many variables were tested for one player of the team and also for both. For instance, the total points, the trump points, the number of aces, the number of sevens, the number of trumps, being the first to play, having the trump ace, the number of suits. However, many of them were rejected by the null hypothesis with a significance level of 0.05, and the remaining features were only three: team aces number, team sevens number and team trumps number.

```
Linear regression model:
    TeamFinalPoints ~ 1 + teamNumTrumps + teamNumAces + teamNumSevens

Estimated Coefficients:
                     Estimate      SE        tStat        pValue
                     ________    _______    _______    __________

    (Intercept)       -12.839     2.2919    -5.6018     2.7444e-08
    teamNumTrumps      6.4878     0.3307     19.619     1.0318e-72
    teamNumAces        13.472    0.48949     27.522    1.8088e-124
    teamNumSevens      7.2091     0.4911     14.679     2.6071e-44


Number of observations: 1000, Error degrees of freedom: 996
Root Mean Squared Error: 15
R-squared: 0.553,  Adjusted R-Squared 0.552
F-statistic vs. constant model: 411, p-value = 8.44e-174
```

Figure 4.4: Linear regression with the *team aces number*, the *team sevens number* and the *team trumps number* as predictor variables and *team final points* as the response variable

Figure 4.4 shows the detailed statistic relationship of the mentioned variables on the team final result. Although the model has a low r-squared value, the p-values of the predictors can reject their null hypothesis and prove their importance in the final result.

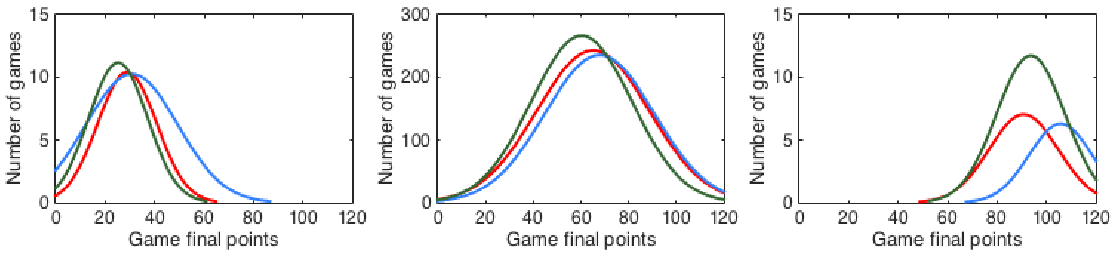


(a) Variable *team trumps number* (b) Variable *team aces number* (c) Variable *team sevens number*

Figure 4.5: Fitting function for each individual predictor variable to estimate the final points of the team

Finally, Figure 4.5 can help quantifying the importance of each feature and to build the domain values for each class (*hard*, *medium* and *low*). Regarding the Figure 4.5a, the numbers of initial trumps held by a team, where at least 60% of the samples were lost games, are 1, 2, 3 and 4. In the same way, the numbers of initial trumps held by a team, where ate least 60% of the samples were won games, are 6, 7, 8 and 9. Applying the same procedure to the three predictor variables, the decided *hard* initial conditions to a team (with low probability of winning) are having at most 4 trumps, at most 1 ace and at most 1 seven. Conversely, the decided *easy* initial conditions to a team (with high probability of winning) are to have at least 6 trumps, at least 3 aces and at least 3 sevens. Other cases are considered *medium* hands.

The developed classifications will be used in two different measures: (1) the FGR, which means the percentage of won or drawn games; (2) the histogram of points obtained by a certain team. The first measure is more general and is used to compare the effective performance of players. On the other hand, the second measure details the points distribution, since the same FGRs can lead to a different histogram of the final points.



(a) 31, 25 and 29 *hard* games out of executed 1000 games for each scenario A, B and C, respectively

(b) 941, 954 and 948 *medium* games out of executed 1000 games for each scenario A, B and C

(c) 29, 948 and 23 *easy* games out of executed 1000 games for each scenario A, B and C, respectively

Figure 4.6: Fitting curves of the final points histograms from 1000 games in each Scenario A (dark green), B (red) and C (dark blue), divided into initial hands classification

Figure 4.6 shows the distribution of the final points obtained in Scenarios A (dark green), B (red) and C (dark blue) divided into the three aforementioned classifications. Every fitting curve presented has its corresponding histogram in Appendix A. The goal of dividing Figure 4.3 is to see if the detected differences play a prominent role in each individual initial hand type. However, out of each 1000 collected games, a few percentage refers to games with *hard* or *easy* initial conditions (between 2% and 3% each one), which means the apparent results from *hard* and *easy* initial conditions have low confidence values. The final points of a team are higher as its Gaussian fitting curve moves to the right in the x axe. Hence, the distribution charts for *hard* and *medium* evidence the same results: the team with 1 Rule-based player (Scenario B) achieved higher final scores and the team with 2 Ruled-based players (Scenario C) obtained even higher when compared to 4 Random players (Scenario A). Nevertheless, regarding *easy* initial conditions, the team with 1 Rule-based player underperforms slightly the other two scenarios.

Additionally, the FGRs, presented in Figure 4.7, measure the effectiveness of players when competing with each other. This rate also evidenced that the Rule-based player outperforms the Random player, mainly based on the FGR of the *medium* hands, in which the confidence is higher due to number of samples.

Favourable Games Rate (%)	Hard Games	Medium Games	Easy Games
Scenario (a)	0	50,6	100
Scenario (b)	0	59,0	100
Scenario (c)	13,8	63,5	100

Figure 4.7: FGR in each Scenario A, B and C, divided into initial hands classification

In sum, the Rule-based player guided the development of new measures and will be the benchmark for evaluating the PIMC algorithm.

4.2.2 The Trick Player

The PIMC algorithm, implemented as described in Section 4.1, cannot explore complete trees until the middle of the game, and therefore, the depth had to be limited. So, the first two possible parametrizations of the algorithm were the depth limit and the N that defines the number of different distributions to be sampled while choosing a card to play. As a result, two distinct branches were clear, creating a version with a low depth limit and a high N value, and another one that has a higher depth limit with lower N values. Additionally, the third possible parametrization is the utility function used by the player.

The Trick player, as the name suggests, evaluates only one trick of every game tree (depth limit of 1) and samples 1000 different distributions (N value). The mean time of its deliberation process for each move is 0.13 seconds. Its utility function is modelled by:

$$u_1 = \begin{cases} teamPoints, & teamPoints \geq opponentTeamPoints \\ -opponentTeamPoints, & teamPoints < opponentTeamPoints \end{cases} \tag{4.1}$$

In order to observe the Trick player performance, the following scenarios will be considered:

- D - 1000 games with 1 Trick player and 3 Rule-based players [yellow];
- E - 1000 games with 2 Trick players against 2 Rule-based players [orange].

The FGR of the team with 1 Trick player and 1 Rule-based (Scenario D) was 52.8% (50.5% won games and 2.3% drawn games), and at the same time, the FGR of the team with 2 Trick players (Scenario E) was 55.6% (53.4% won games and 2.2% drawn games). In the same way there was a difference between Scenarios B and C, having two Trick players on the same team significantly improves the results when compared to only one. This evidence was expected, since *Sueca* is a team game.

Additionally, Figure 4.8 presents the distributions of the 1000 obtained final scores from each scenario. Scenario A (green) was also included to provide a baseline of equilibrium. The Gaussian fitting curve of Scenario D (yellow) is nearly coincident with the one of Scenario A, suggesting one Trick player in a team does not influence the results. On the other hand, a team with 2 Trick players, Scenario E (orange), evidences a higher distribution between 60 and 80 points.

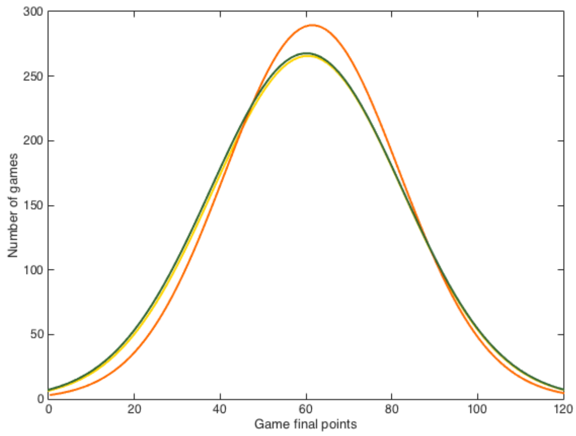
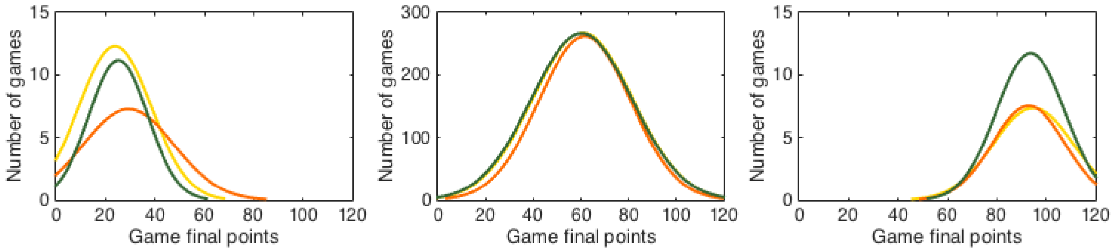


Figure 4.8: Fitting curves of the final points histograms from 1000 games in each Scenario A (dark green), D (yellow) and E (orange)

These conclusions are also supported by the FGRs of 50.4%, 50.5% and 53.4% from the teams of Scenarios A, D and E, respectively.



(a) 31, 27 and 24 *hard* games out of executed 1000 games for each scenario A, D and E, respectively

(b) 941, 953 and 954 *medium* games out of executed 1000 games for each scenario A, D and E

(c) 28, 20 and 22 *easy* games out of executed 1000 games for each scenario A, D and E, respectively

Figure 4.9: Fitting curves of the final points histograms from 1000 games in each Scenario A (dark green), D (yellow) and E (orange), divided into initial hands classification

When analysing the Gaussian fitting curves divided into the classes of initial hands, results agree with previous conclusions, except for the *hard* classification of initial conditions. So, in Scenario E (orange) the team of 2 Trick players achieved higher scores with *hard* and *medium* hands, while with *easy* hands the modal values of the three scenarios seem to be concurrent. However, as previously mentioned, out of each 1000 collected games, a few percentage refers to games with *hard* or *easy* initial conditions (between 2% and 3% each one), which means the apparent results from *hard* and *easy* initial conditions have low confidence values.

Favourable Games Rate (%)	Hard Games	Medium Games	Easy Games
Scenario (a)	0	50,6	100
Scenario (d)	3,7	53,3	95
Scenario (e)	0	56	100

Figure 4.10: FGR in each Scenario A, D and E, divided into initial hands classification

Moreover, the FGRs in Figure 4.10 evidenced that the Trick player slightly outperforms the

Rule-based player, mainly based on the FGR of the *medium* hands, in which the confidence is higher due to number of samples. The unexpected differences on Scenario D with both *hard* and *easy* initial conditions refer to only one game, in both cases, and may reflect some flaws on the classification inferred from the linear regression.

4.2.3 The Deep-1 Player

In contrast to the last player, which has a low depth limit and high N value, the Deep-1 player has the highest reasonable depth limit and lower N values. In other words, each time this player has to choose a move, it sets the maximum depth limit considering it has to sample at least 30 different distributions and its deliberation time must be less than 2 seconds. Figure 4.11 shows the chosen N values and depth limits for each tree size, and also the explored depth percentage.

Full tree depth size	N	Depth limit	Explored depth (%)
10	50	3	30
9	50	3	33.33333333
8	50	3	37.5
7	100	3	42.85714286
6	50	4	66.66666667
5	50	-	100
4	200	-	100
3	1000	-	100
2	1000	-	100

Figure 4.11: N values and depth limits for each tree size, and also the explored depth percentage

Additionally, the mean time of its deliberation process for each move is 0.6 seconds and its utility function is the same of the Trick player, presented in Equation 4.1.

In order to observe the Deep-1 player performance, the following scenarios will be considered:

- F - 1000 games with 1 Deep-1 player and 3 Rule-based players [pink];
- G - 1000 games with 2 Deep-1 players against 2 Rule-based players [purple].

The overall FGR of the team with 1 Deep-1 player and 1 Rule-based (Scenario D) was 58.3% (57.6% won games and 0.7% drawn games), and at the same time, the FGR of the team with 2 Deep-1 players (Scenario E) was 64.2% (62.7% won games and 1.5% drawn games). In the same way there was a difference between Scenarios D and E, having two Deep-1 players on the same team significantly improves the results when compared to only one.

Figure 4.12 presents the Gaussian fitting curves of the Scenarios D and E as reference points, and Scenarios F and G. The final points obtained by the team with 1 Deep-1 player and 1 Rule-based (f, pink) were higher than the ones obtained by the team with 1 Trick and 1 Rule-based (d, yellow), which is visible in the density between 80 and 120 points. Similarly, the team with 2

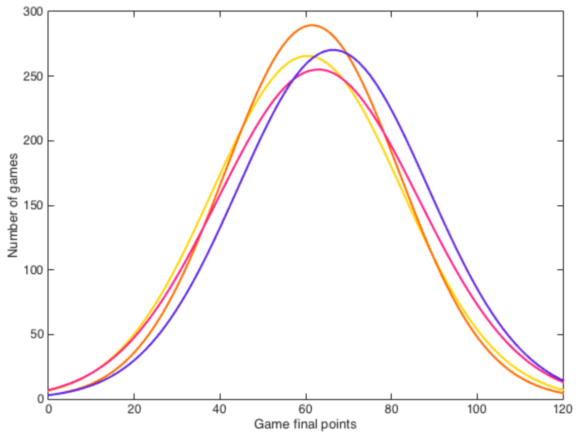
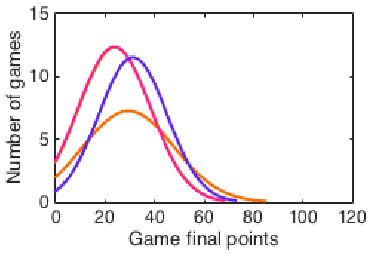
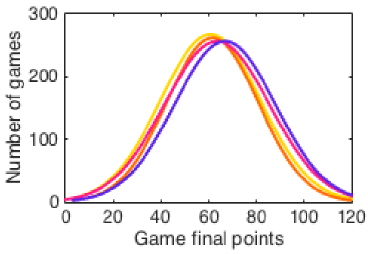


Figure 4.12: Fitting curves of the final points histograms from 1000 games in each Scenario D (yellow), E (orange), F (pink)and G (purple)

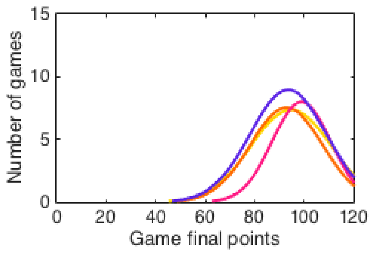
Deep-1 players also has its Gaussian fitting curve on the right of the team with 2 Trick players, suggesting its achieved final points were higher.



(a) 27, 24, 33 and 29 *hard* games out of executed 1000 games for each scenario D, E, F and G, respectively



(b) 953, 954, 945 and 948 *medium* games out of executed 1000 games for each scenario D, E, F and G, respectively



(c) 20, 22, 22 and 23 *easy* games out of executed 1000 games for each scenario D, E, F and G respectively

Figure 4.13: Fitting curves of the final points histograms from 1000 games in each Scenario D (yellow), E (orange), F (pink)and G (purple), divided into initial hands classification

Figure 4.13 presents the Gaussian fitting curves of Scenarios D, E, F and G divided by the three initial hands conditions. By comparing results of *medium* initial conditions, from Scenarios D to G, each evaluated team incrementally improves the previous one. On the other hand, both teams from Scenarios F and G underperform in the achieved final scores for *hard* initial hands and outperform the achieved finals cores for *easy* initial hands when compared to Scenarios D and E.

Favourable Games Rate (%)	Hard Games	Medium Games	Easy Games
Scenario (d)	3,7	53,3	95
Scenario (e)	0	56	100
Scenario (f)	0	59,4	100
Scenario (g)	0	65,3	100

Figure 4.14: FGR in each Scenario D, E, F and G, divided into initial hands classification

Finally, the last Figure 4.14 comparing the Deep-1 player and the Trick player agrees with previous conclusions and emphasises the effective competing results of Deep-1 player.

4.2.4 The Deep-2 Player

The last configuration of the PIMC algorithm is the Deep-2 player. Its difference from the Deep-1 player is the utility function, modelled by Equation 4.2.

$$u_2 = \begin{cases} 2, & teamPoints > 90 \\ 1, & teamPoints > 60 \\ 0.1, & teamPoints > 30 \\ -2, & opponentTeamPoints > 90 \\ -1, & opponentTeamPoints > 60 \\ -0.1, & opponentTeamPoints > 30 \end{cases} \tag{4.2}$$

Instead of maximizing the final points, this utility function groups the final points into 6 possible rewards for the agent and tries to maximize the number of won games. The main advantage of this utility function is the time spent on the game search, since there are more nodes with the same rewards, and therefore some $\alpha\beta$-cuts occur earlier. On the other hand, when limiting the depth of the search, without any heuristic, PIMC algorithm may be misled to worse nodes.

In order to observe the Deep-2 player performance, the following scenarios will be considered:

- H - 1000 games with 1 Deep-2 player and 3 Rule-based players [light blue];
- I - 1000 games with 2 Deep-2 players against 2 Rule-based players [light green].

The overall FGR of the team with 1 Deep-2 player and 1 Rule-based (Scenario D) was 58.6% (57.3% won games and 1.3% drawn games), and at the same time, the FGR of the team with 2 Deep-1 players (Scenario E) was 62.6% (61.79% won games and 0.7% drawn games). In the same way there was a difference between Scenarios F and G, having two Deep-2 players on the same team significantly improves the results when compared to only one, although FGRs have decreased from the last scenarios.

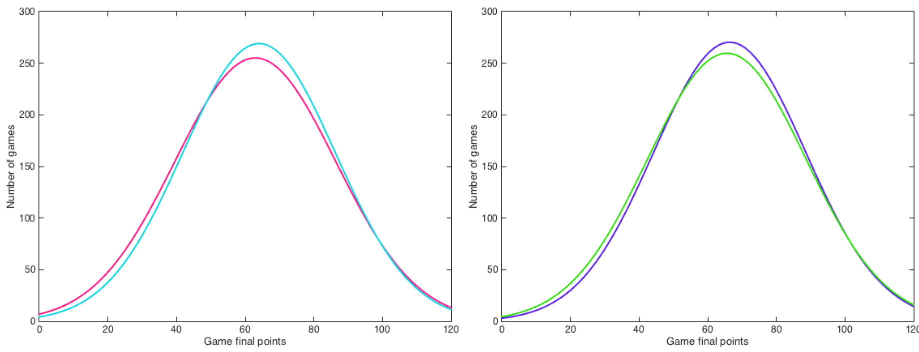
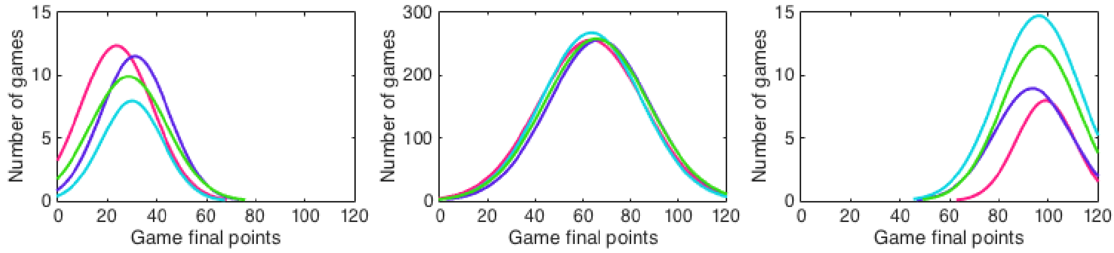


Figure 4.15: Fitting curves of the final points histograms from 1000 games in each Scenario F (pink) and H (light blue) on the left, G (purple) and I (light green) on the right

Figure 4.15 presents the Gaussian fitting curves of Scenario H compared to Scenario F on the left, and the Gaussian fitting curves of Scenario I compared to Scenario G on the right. The two main considerations about these charts are the modal values of Scenario G (light blue) in the frontier of winning results; also, the team with 2 Deep-2 players underperforms in the obtained points of the team with 2 Deep-1 players.



(a) 33, 29, 24 and 28 *hard* games out of executed 1000 games for each scenario F, G, H and I, respectively

(b) 945, 948, 938 and 942 *medium* games out of executed 1000 games for each scenario F, G, H and I, respectively

(c) 22, 23, 38 and 30 *easy* games out of executed 1000 games for each scenario F, G, H and I, respectively

Figure 4.16: Fitting curves of the final points histograms from 1000 games in each Scenario F (pink), G (purple), H (light blue) and I (light green), divided into initial hands classification

Figure 4.16, as shown for other scenarios, divides the histogram fitting curves of final points into the classifications of the initial conditions. Slight deviations with the *hard* and *easy* initial conditions are negligible, due to the discrepancy of samples in each scenario. The *medium* initial conditions chart evidences the similarities in the four approaches.

Favourable Games Rate (%)	Hard Games	Medium Games	Easy Games
Scenario (f)	0	59,4	100
Scenario (g)	0	65,3	100
Scenario (h)	0	58,9	100
Scenario (i)	0	63,4	100

Figure 4.17: FGR in each Scenario F, G, H and I, divided into initial hands classification

Additionally, Figure 4.17 presents the FGRs of Scenarios F, G, H and I, suggesting that the Deep-2 player underperforms the Deep-1 player.

4.2.5 Conclusion

In order to clearly compare both developed parametrizations of PIMC algorithm, Figure 4.18 summarizes the FGR achieved by each player when playing with and against Rule-based players. Taking the results into account, Deep-1 was the chosen player to participate in the user studies described in Chapter 7.

Favourable Games Rate (%)	Hard Games	Medium Games	Easy Games
Trick Player	3,7	53,3	95
Deep-1 Player	0	59,4	100
Deep-2 Player	0	58,9	100

Figure 4.18: FGR in each Scenario D, F and H, divided into initial hands classification

5

User-centred Studies

Contents

The chosen *Sueca* scenario gives this work an opportunity to study HRI with a senior audience in an entertainment activity. However, developing a robot for aged people brings some delicate questions. The potential users sometimes have few, or nonexistent, experience with technology, which makes it difficult for them to understand how robots work and what they can actually do. As a result, understanding their needs, expectations, and fears is another concern [1].

The current chapter explains the methodology, procedures and current results of a developed user-centred study in a care home. It involved two different activities, a focus group and a pilot card game study, as a result of two distinct motivations: to understand the elderly' concerns about robots, and to analyse both the game flow and the interactions between players during a *Sueca* session of games.

5.1 Focus Group

A focus group is a good approach for a first meeting due to the informal and conversational way of interacting with participants. The goal of this activity was to introduce to the elderly the robots' theme, and to understand their opinions and expectations. To accomplish this purpose, used techniques were a Brainstorming and a Storytelling.

5.1.1 Methodology

The elderly participants were divided into groups of 5. There were 2 researchers per group commanding and guiding all the process. The list of materials used, per group:

- An illustrative video of existing robots;
- 6 photographs of different robots, including 3 of service type and 3 of companion type (Paro, EMYS, Pleo, Pearl, PR2, and Care-O-Bot);
- Two white boards and three pens (black, red and green);
- Three hypothetical stories of robots;
- An audio recorder;
- Four lavalier microphones;
- A video camera.

The last three items will only be used for a further analysis of this focus group. The video tries to answer the questions: what is a robot, what can robots do, how do they work, do they fail and how do science fiction movies present robots to us. In order not to bias their thoughts, we tried to gather positive and negative aspects of existing robots. The three hypothetical stories aim to bring ethical discussions to the focus group [18, 36]. For instance, an elderly that owns a robot in his home tells him a secret. If that robot is questioned about the secret, should it or should it not tell other people the truth?

5.1.2 Procedures

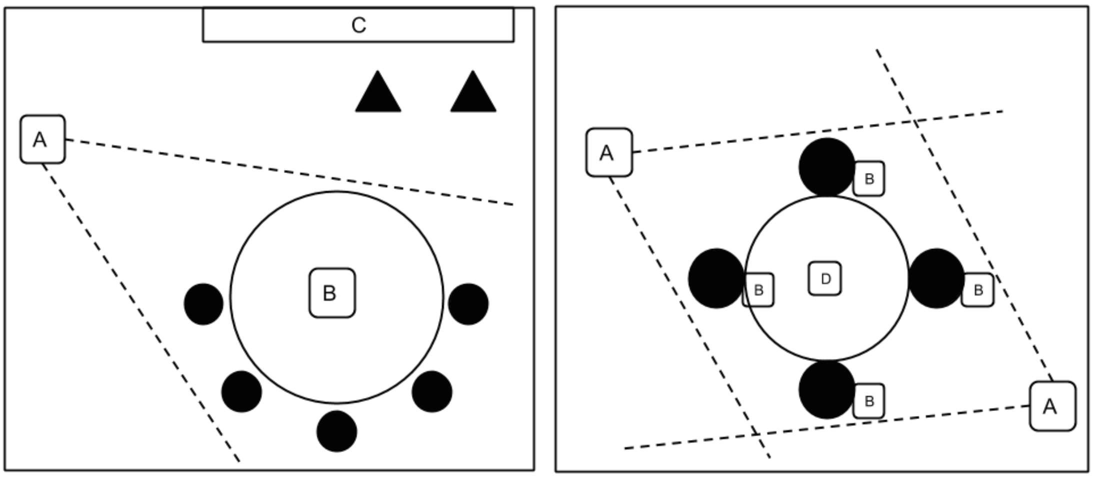
All the materials enumerated in the previous list were arranged as in Figure 5.1a. Firstly, each person in the room briefly introduces himself in order to make everyone feeling more comfortable. Secondly, the video is shown. Then, everyone starts discussing about robots' purposes and they are registered in one of the white boards with the black coloured pen. People also express a positive or negative impression of each robot's purpose and their opinions decide the colour of the surrounding line (Appendix). For instance, the sentence "Call an ambulance" written on the board is surrounded by a green line if they think it is a good purpose for a robot. After finishing this task, one of the group leaders writes all the sentences previously collected in the second board but without the surrounding green or red lines. The other group leader starts reading the hypothetical stories and opens a new discussion about what the robots of each story should do. He also presents the photographs and tries to understand which robot is more suitable for each purpose in their opinion. When bringing the new board to the room, the idea is to understand if their positive and negative opinions about each purpose have changed.

5.1.3 Results

Three focus group sessions were performed and analysed, with 16 participants from a day-home care institution in Lisbon (12 females, 4 males; M age = 78.69 ± 12.20). Most subjects lived alone in their home (81.3%), or with their friends (12.5%), and relatives (6.3%).

The analyses focused on the mentioned activities in which independent-living older adults require a robot. Out of 75 mentioned activities, 65 were non-repeated activities that were further classified according to their primary goal and context. The used classification was: Basic Activities of Daily Living (BADL), Instrumental Activities of Daily Living (IADL), Enhanced Activities of Daily Living (EADL) and Social Activities (SA). Consequently, the distribution of activities went as follows: 24 IADL, 17 BADL, 12 EADL and 12 SA. These results evidenced how independent-living older adults expect robots might help to improve their quality of life, and also which type of activities they require the most.

Moreover, the analysis on the robot types demonstrated their preferences on service robots for each activity type. The main reason for this choice is their physicality, which is perceived as less limiting when compared to other robot types.



(a) Focus Group (b) Card game

Figure 5.1: Setting of the user study activities. **A** - Video recorder **B** - Audio recorder / Microphone **C** - White board **D** - cards ● - Aged person ▲ - Group leader

5.2 Card Games

The further reproduction of human behaviours, in the *Sueca* domain, will be required for the social robotic agent. Consequently, this user-centred study plays a crucial role in the following stages of this work. Besides collecting examples of verbal and non-verbal interactions from human players, the analysis of these card games aimed to capture the precise game situations that trigger those behaviours.

5.2.1 Methodology

Considering each *Sueca* card game includes four players, the required materials were: a card deck, a table and chairs for the four players, two video cameras and an audio recorder with four microphones.

5.2.2 Procedures

All the previously enumerated material was arranged as in the Figure 5.1b. Each video camera was positioned to capture the hands of two adjacent players. Players were recorded during a tournament of several games. They were told to play as long as they wanted with a maximum duration of one hour.

5.2.3 Results

The session took only 40 minutes, since the 4 players were feeling weary. Ten games, with an average duration of 3,75' each, were recorded for a further analysis. From the average duration, 1' belongs to the initial setting of shuffling, distributing, and rearranging the cards in each hand.

Relevant Game Situation
Shuffling
Cutting
Dealing
Receiving cards
Choosing the next play
Playing a card
Playing a trump card
Winning the trick
Winning the game
Losing the trick
Losing the game

Figure 5.2: Relevant game situations extracted from the user-centred studies

Figure 5.2 lists all the captured game situations that trigger verbal and non-verbal behaviours. Additionally, a player interacts when executing himself each action of the list or, also, when other player does it.

Table 5.1: Examples of expressions collected during the card game activity and its respective classification.

Expression	**Game Stage**	**Intention**
Joga [player-name]*!*	Before a play	Speed up a play.
Anda [player-name]*!*	Before a play	Speed up a play.
Podes jogar, [player-name]*!*	Before a play	Speed up a play.
Quase que livrámos.	After collecting the first points	Hopeful or ironic comment.
E eu puxo trunfo.	Initialising a turn with a trump card	State an action.
Outro trunfo!	Play a trump card after an already played trump card	State an action.
Outro(a) [suit]*!*	Play a [suit] card after an already played [suit] card	State an action.
O trunfo é [suit].	Anytime	Give game information. Answer a question.

Table 5.1 illustrates the some of the collected expressions, their corresponding game situations and their intentions. Considering players said specific domain words, expressions were not translated in order not to lose their meaning and regarding the future usage of these sentences in a Portuguese environment.

Furthermore, there were other relevant considerations from these games analysis. After a game, paired players frequently discuss extremely good or bad moves from each other. Also, their main gazing points were the table zone with the card being played and their hands.

6

EMYS: the *Sueca* player

Contents

Revising the purposes of this work, the robotic agent that plays *Sueca* has two main tasks: to choose an adequate card to play and to interact socially according to the game state. The AI module, previously described in Chapter 4, answers clearly to the first goal. In the same way the current chapter explains how the second goal has been achieved. It starts with an overview of the whole system and proceeds with the development of the social agent and the considerations that were taken into account.

6.1 Architecture Overview

The architecture presented in Figure 6.1 organises all the components involved in this system and their communications. It considers a scenario where an embodied agent plays a physical card game against human players over a touch table.

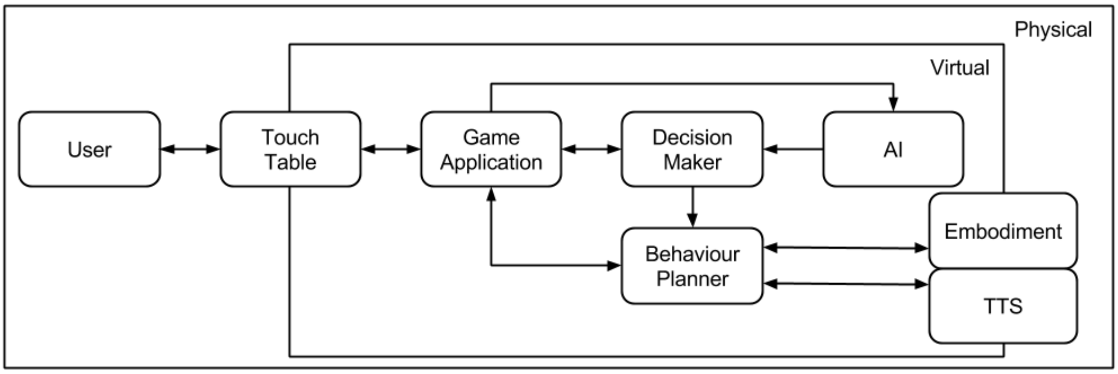


Figure 6.1: System architecture using components

First of all, this model distinguishes physical components from virtual ones. However, some entities are presented as both physical and virtual components and will not be detailed since their usage in this system did not demand any extensions for the scope of our domain (*Touch Table*, *Embodiment* and *Text To Speech (TTS)*).

The basic work-flow that illustrates the main functionalities of each component is as follows. The human players, *Users*, play with physical cards on top of a *Touch Table*, and their game actions are managed by the *Game Application* and communicated to both the *AI* and the *Decision Maker*. The *AI* includes all the reasoning about the game and decides the next move of the artificial player. However, the *Embodiment* will not only play a certain card, but will also include social behaviours. As a result, the *Decision Maker* balances the *AI* decisions and game information to produce an appropriate sequence of behaviours and inform them to the *Behaviour Planner*. Lastly, the *Behaviour Planner*, after receiving high-level intention-directed instructions, builds a suitable plan to execute the chosen instructions, considering the state of the *Embodiment*, *TTS*, and additional game information from the *Game Application*.

The previously described architecture is instantiated as shown in Figure 6.2 and the blue

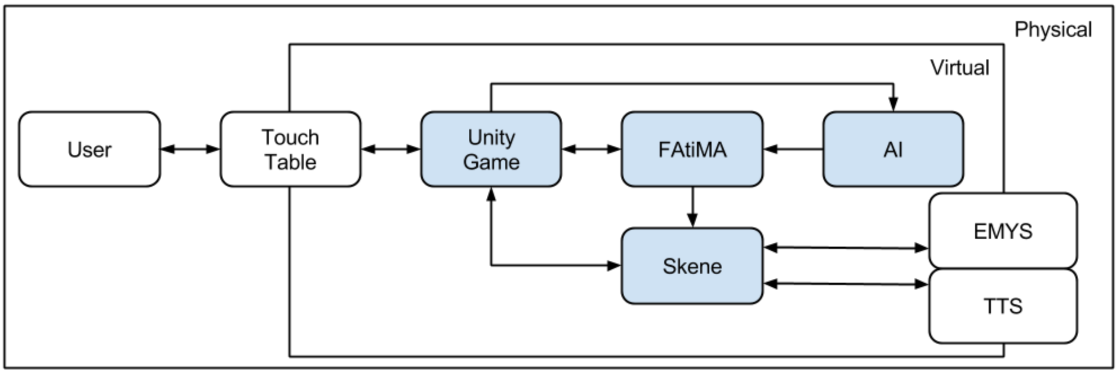


Figure 6.2: System architecture using modules

modules are thalamus communicating entities. This concept arises from the Thalamus Framework [30], which enables the usage of entities that can be registered at runtime in a server so it becomes possible to send and receive specific messages. These entities are publishers and subscribers of the channels they want to write on and listen to, respectively. The implementation provided by this framework works by simply inheriting from the *ThalamusClient* class and implementing the interfaces of the messages that the entity wants to exchange.

The *Unity Game* module is responsible for displaying the interface of the game, reading the physical cards, publishing all the relevant game events and subscribing to play events of the artificial players.

The chosen *Behaviour Planner* is *Skene* [31], which tightens the communication between the world and an embodied agent with a high-level behaviour description language, also known as utterances. These utterances might include instructions for gazing, pointing, animating or sound, among other things. Additionally, considering some instructions require target positions or other game information, *Skene* subscribes to *Unity game* messages to keep that information updated.

The *AI* module contains an instance of the Deep-1 Player presented in Chapter 4. Moreover, the implementation of *FAtiMA* module [8], as decision maker of our *Sueca* player, is carefully described in the following section.

6.2 A social player

First of all, a social player in a card game scenario is basically a player that can interact with other human players in a proper way according to the game situation. Since its behaviours must be as similar as possible to the interactions of human players, the most expressive robot was chosen to embody this player, EMYS. Nevertheless, when creating behaviours for an embodied agent, it is important to consider that our perception of a social robot, as a unique entity that interacts, is indeed composed of distinct modules that allow the robot to talk, move, animate, gaze at some point or glance at another. For this reason, the architecture, presented in Figure6.2, uses

Skene as its *Behaviour Planner*. *Skene* has its own language, called utterances, that allow the communication with the robot as a single entity. These utterances are classified with a category and subcategory and may specify verbal or non-verbal behaviours, as well as both interleaved. Most of EMYS behaviours in this scenario were conducted by *Skene* due to the provided abstraction while producing complete behaviours (verbal and non-verbal), and also due to the utterances classification that can associate behaviours to game states. The following subsections will present the main aspects of the utterances list that characterizes EMYS behaviours and the way it will be perceived.

6.2.1 *Sueca* behaviours

The analysis of the card game players on user-centred studies, Chapter 5, revealed key aspects of the interaction during a *Sueca* game. First of all, there are specific game situations that may cause verbal or non-verbal behaviours. As a result, these game situations guided the categories and subcategories of the utterances list, presented on the following figure.

Session Start	Session End	GameEnd		Trick End	Receive Cards
Greeting	Win	Single Win	Single Lost	Self	Self
	Lost	Double Win	Double Lost	Team	
	Draw	Quad Win	Quad Lost	Opponent	
		Team Cheat	Draw	Opponent Zero	
		Other Cheat			
Play	**Playing**	**Shuffle**	**Cut**	**Deal**	**Next Player**
Self Happy	New Trick	Self	Self	Self	Team
Happy For	Following	Other	Other	Other	Opponent
Gloating	Not Following				
Resentment	Cut				
Self Pitty					
Pitty					

Figure 6.3: Categories and subcategories of the utterances list

The final list of 205 distinct utterances was inspired by the collected behaviours and replicated to similar ones in order to enrich interactions and to avoid speech redundancies. The annotated non-verbal behaviours were also applied on EMYS during the same game situations, for instance, looking at a played card and analysing its own hand after that, simulating a re-evaluation of the game.

6.2.2 Human-like behaviours

Besides simply replicating behaviours from human players, there are other things to consider in order to make the robot act as a human, for instance, its speech frequency or its emotional state. Consequently, this social player applies a probability to decide whether or not to perform an utterance for each game situation. Additionally, the *FAtiMA* module was used as decision maker of our *Sueca* player, as shown in Figure 6.2, to enrich EMYS presence and allow it to share its emotional state.

FAtiMA is a modular architecture for an emotional agent capable of producing 22 different emotions based on its goals and its perceptions of new events for a determined scenario. Perceptions can be updated by changing the values of 6 appraisal variables (desirability, desirability for other, success probability, failure probability, praiseworthiness and like) and their combination can generate one or more emotions. However, the current emotional agent of this *Sueca* player is only using 4 appraisal variables, which means it only produces 12 emotions, as presented in Figure 6.4.

Used emotions	Happy for	Satisfaction	Joy
	Resentment	Relief	Distress
	Gloating	Disappointment	Hope
	Pitty	Fears confirmed	Fear
Unused emotions	Gratification	Pride	Love
	Gratitude	Admiration	Hate
	Remorse	Shame	
	Anger	Reproach	

Figure 6.4: Distinction of used and non used FAtiMA emotions

The first approach was to subcategorize utterances with emotional states, however, most of the annotated behaviours by human players evidenced they revealed their emotions in scarce situations during a the game. A possible reason may lay on the fact that *Sueca* has the element of chance and unknown information should remain hidden.

As a result, emotional states were used on this social player to subcategorize only utterances of the *Play* category. These utterances are triggered by a play from any player and the idea is to produce an adequate behaviour considering the immediately rewarded benefit. In other words, each time a player plays a card, the current winner of the trick is computed to analyse how much the agent benefits with that move and also the player itself. With this strategy, when the agent or its team player make a move, the possible emotions are *Happy For* and *Pitty*, otherwise, when an opponent plays, the possible emotions are *Resentment* and *Gloating*.

Besides the previous usage of the emotional agent, this *Sueca* player is permanently exhibiting its emotional state through its posture. Since the game success probability is always being updated, together with the mentioned perception of reward, this agent also produces *joy*, *distress*, *hope* and *fear* to set its posture during the game.

Finally, another consideration was the opponent and partner component of the *Sueca* game. From the analysis based on user-centred studies (Chapter 5), annotated verbal behaviours presented some differences between partners and opponent. For instance, players tend to be more supportive and encouraging to partners and more competitive to opponents. Theses differences were also included in our *Sueca* player to subcategorize some utterances, as shown in Figure 6.3.

6.2.3 Enhancing the game interface with behaviours

Beyond the idea of creating a player that acts humanly in this scenario, other considerations must influence its behaviours. The final game interface was quite similar to what traditional *Sueca* players are used to, specially due to the usage of physical cards. However, there are two main concerns to consider when playing over the touch table instead of a traditional *Sueca* game: players must respect their time to play in order for the card to be assumed in the correct order; when a trick has finished, cards must be removed in order to proceed the game.

Consequently, the two utterances' categories differing from analysed human behaviours were *Next Player* and *Trick End*. The first one is different mainly due to the frequency the agent talks to the next player. This frequency is higher than the observed by human players in order to enhance this new game experience and encourage players to play on their own times. The second pointed difference, in *Trick End* utterances, was not taken from user-centred studies. The pilot experiences evidenced the urge of introducing some cues to remove cards after the trick, and this *Sueca* player warned other players about this.

7

User studies

Contents

In order to evaluate the created social robot on the game playing scenario of *Sueca*, a user study was conducted. The main idea was to set up the environment in which this robot is supposed to interact with human players, and collect, in an adequate way, their feelings and perceptions.

The first measure this study aims to evaluate is trust, since *Sueca* contains companionship between team players. At the same time, this game includes two teams competing with each other and therefore, the influence of these conditions can also be calculated for every defined measure. Additionally, measuring the social presence of every *Sueca* partner will also provide a comparison between the two conditions. Finally, the last chosen measure is affect in order to evaluate the evolution of participants' feelings.

Participants answered two questionnaires, one before playing with EMYS and one after. The current chapter starts with the samples description and proceeds with analyses of the three mentioned measures.

7.1 Methodology and Procedures

Each session lasted an hour and involves 3 participants playing with EMYS. Firstly, each subject selected his team player in a draw. Secondly, according to each condition, having a human or robot partner, participants answered a questionnaire before playing *Sueca*. This questionnaire, available in Appendix B the robot partner version, contained two parts: the PANAS Questionnaire [9] and Human-Robot Trust Questionnaire [33]. Then, one researcher explained the game rules with a standard deck, and played some tricks until everyone felt comfortable. After reviewing the *Sueca* game, participants moved to the touch table and started a session of 5 games with or against EMYS, considering the results in the initial draw.

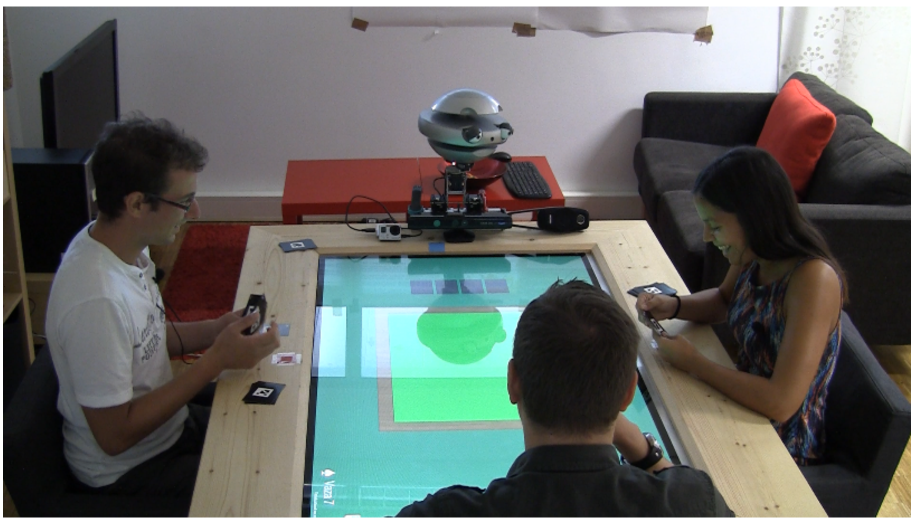


Figure 7.1: Example of a game session

Lastly, participants answered another questionnaire, available in Appendix B the human partner version, divided into four parts: the PANAS Questionnaire, Human-Robot Trust Questionnaire, the Networked Minds Questionnaire [17] and some demographic questions. All statistical analyses further mentioned used a significance level of 5%.

7.2 Samples Description

A group of 60 participants were included in this study with a mean age of 24,31 $\pm$ 3,852. Out of the 60 subjects, 40 played the game with a human partner and 20 played with EMYS. These distributions aimed to collect a valid number of answers from EMYS' partners. Additionally, out of the 59 subjects that revealed their gender, 20 were females and 39 were males. Furthermore, most participants affirmed to know their partners in spite of not having played with them before, and their *Sueca* knowledge was nearly medium.

7.3 Trust

The trust in the partner was measured by the answers of each individual to the Human-Robot Trust Questionnaire, before and after the game session. Consequently, the following three study hypothesis arose:

- Are there changes in trust after the experience of interacting with the *Sueca* partner?
- Are the trust levels influenced by the partner (robot or human)?
- Are the trust levels influenced by the game results?

Are there changes in trust after the experience of interacting with the *Sueca* partner?

The statistical test Mixed ANOVA was used to infer a conclusion about this question, with *time* as a factor of 2 levels and *condition* (partner) as the between-subjects factor. Additionally, assumptions were tested to guarantee the results validity. The dependent variable (*time*) showed a significant effect with $p = 0.03$. However, by adding the independent variable (*condition*), the effect was not significant with $p = 0.65$.

Figure 7.2 presents the evolution of the trust percentage between the two time levels, before and after the game. The trust values correspond to estimated means separated by time, since it was the only significant variable.

Answer: There were significant differences in Trust before and after playing *Sueca*. However, there was no significant differences in Trust before and after playing *Sueca* for different partners. Additionally, the trust levels of participants increased after playing *Sueca* with EMYS.

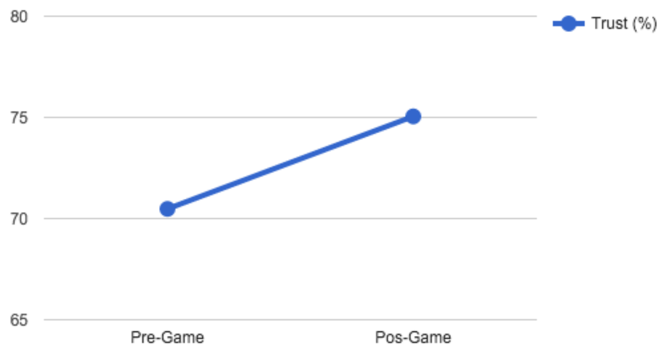


Figure 7.2: Evolution of the trust percentage between the two time levels

Are the trust levels influenced by the partner (robot or human)?

The statistical test Welch Test was used to infer a conclusion about this question, with *condition* as factor and *final trust* as dependent variable. As a result, the *condition* effect was proved with p = 0, suggesting the means of trust were significantly different between having a robot partner or a human partner.

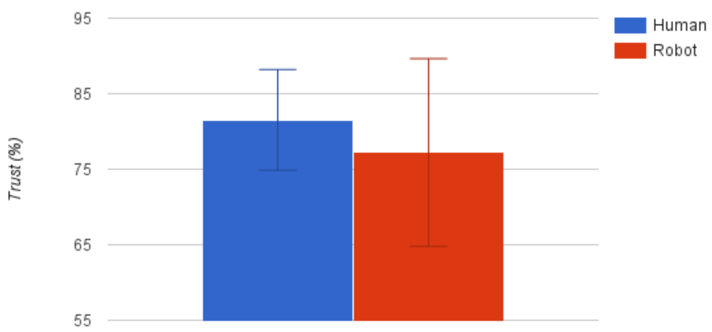


Figure 7.3: Differences of trust levels between conditions

Figure 7.3 evidences the trust level was higher for the *condition* human partner, with a trust mean value of 81.538%, when compared with the *condition* robot partner, with a trust mean value of 77.215%.

Answer: There were significant differences in Trust between different *Sueca* partners. Additionally, subjects' trust levels for human partners was higher than robot partners.

Are the trust levels influenced by the game results?

A two-way ANOVA was run on data to analyse if the game results influenced the trust levels, with *condition* and *game result* as factors, and *final trust* as dependent variable. The effect of *condition* was significant, with p = 0.01 (already proved in the previous question). On the other hand, the *game result* cannot reject the null hypothesis with p = 0.065, and therefore indicates a non significant effect on the trust measure. Moreover, the effect of both *condition* and *game result* also proved to be non significant with p = 0.507.

Answer: There were no significant differences in Trust between different game results, which seems to suggest that independently of losing or winning the game, the perception of Trust in the game partner remains stable.

7.4 Social Presence

After playing the game, each subject answered the Networked Minds Questionnaire in order to measure the social presence of his partner. This measure of social presence includes six different subdimensions: co-presence, attentional allocation, perceived message understanding, perceived affective understanding, perceived emotional interdependence, and perceived behavioural interdependence.

By considering this measure in a *Sueca* scenario, one study hypothesis arose:

- Is the social presence influenced by the partner (robot or human)?

Is the social presence influenced by the partner (robot or human)?

The statistical test One-Way ANOVA was used to infer a conclusion about this question, with *condition* as factor and each social presence subcategories' values as dependent variables. *Condition* presented the following statistical effects on each subdimension results:

- There was not a statistically significant difference between the *co-presence* as determined by one-way ANOVA (F = 1.559, p = 0.217);
- There was not a statistically significant difference between the *attentional allocation* as determined by one-way ANOVA (F = 0.002, p = 0.965);
- There was not a statistically significant difference between the *perceived message understanding* as determined by one-way ANOVA (F = 0.081, p = 0.777);
- There was a statistically significant difference between the *perceived affective understanding* as determined by one-way ANOVA (F = 7.850, p = 0.007);
- There was a statistically significant difference between the *perceived emotional interdependence* as determined by one-way ANOVA (F = 4.148, p = 0.046);

- There was not a statistically significant difference between the *perceived behavioural interdependence* as determined by one-way ANOVA (F = 0.699, p = 0.406).

The social presence of partner evidenced discrepancies for the two conditions in two subdimensions: *perceived affective understanding* and *perceived emotional interdependence.* As a result, Figure 7.4 shows theses discrepancies, demonstrating the *perceived affective understanding* and *perceived emotional interdependence* were higher in human partners.

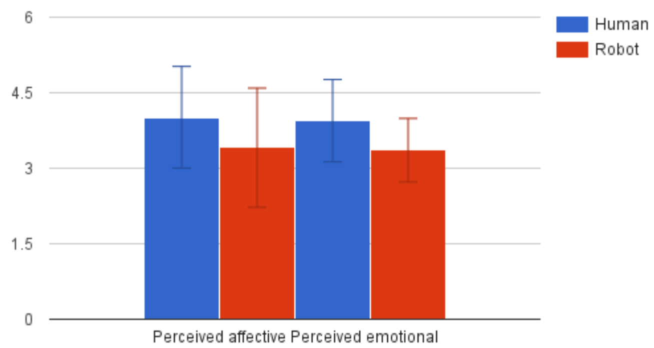


Figure 7.4: Perceived affective understanding and perceived emotional interdependence means for each condition

Answer: There were significant differences in Social Presence between *Sueca* partners for two dimensions: perceived affective understanding and perceived emotional interdependence. The mean values of both subdimensions were higher for human partners. Additionally, there were no significant differences in the remaining subdimensions of Social Presence between *Sueca* partners.

7.5 Affect

The affect was measured by the answers of each individual, before and after the game session, to the PANAS Questionnaire. It is divided into positive and negative affects and, therefore, there are two study hypothesis:

- Are there changes in the positive affect after the experience of interacting with the *Sueca* partner?

- Are there changes in the negative affect after the experience of interacting with the *Sueca* partner?

Are there changes in the positive affect after the experience of interacting with the *Sueca* partner?

In order to answer this questions, a Mixed ANOVA was run on the collected data, with *time* as a factor of 2 levels and *condition* as the between-subjects factor. So, *time* proved to have a statistical significant effect on the positive affect, $p = 0.008$. On the other hand, *time* levels for each *condition* did not present a significant effect on the positive affect, $p = 0.488$.

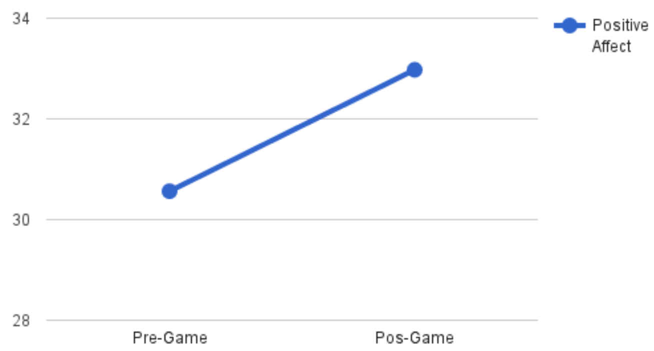


Figure 7.5: Evolution of the positive affect between the two time levels

Figure 7.5 evidences the evolution of the positive affect before and after playing *Sueca* with EMYS.

Answer: There were significant differences in Positive Affect before and after playing *Sueca*. However, there were no significant differences in Positive Affect before and after playing *Sueca* between different partners.

Are there changes in the negative affect after the experience of interacting with the *Sueca* partner?

In order to answer this questions, a Mixed ANOVA was run on the collected data, with *time* as a factor of 2 levels and *condition* as the between-subjects factor. The dependent variable (*time*) did not present a significant effect with $p = 0.267$. Furthermore, by adding the independent variable (*condition*) to *time*, the effect was also not significant, with $p = 0.184$.

Answer: There were no significant differences in Negative Affect before and after playing *Sueca*. Also, there were no significant differences in Negative Affect before and after playing *Sueca* influenced by different partners.

*

Overall, the difference on the trust levels between conditions suggested that humans cannot yet trust in robots, when playing *Sueca*. In addition, the trust levels were not influenced by the game result, which reinforces the importance of condition on this measure. However, trust levels have increased after playing the game, without the influence of the condition. On the other hand, the social presence of the partner, was not influenced by the condition in most subdimensions, suggesting this robotic *Sueca* player was socially perceived as a human in those subdimensions. The first difference influenced by the condition, on the *perceived affective understanding*, suggests that people who had EMYS as partner were either less able to perceive its affective state, or they found it difficult for EMYS to understand their affective state. The second difference influenced by the condition, on the *perceived emotional interdependence*, suggests that people who had EMYS as partner were either less affected by its affective state, or they found EMYS was less affected by their affective state. Interestingly, the second difference may be caused the first one. Finally, even though the negative affect did not change after the game with EMYS, the positive affect increased after the game, suggesting it was a pleasing experience for participants.

8

Conclusions

Contents

This thesis addressed three main contributions aligned with the problems presented in Section 1. First of all, the implementation of the PIMC algorithm on an artificial *Sueca* player and later analysis on different parametrizations of this algorithm. Additionally, this intelligent player was included as a module of an architecture for a social *Sueca* player. This social entity was able of playing the card game with human players while interacting with them according to game state. Finally, we conducted user studies to compare trust and social presence between human partners and EMYS, and also a affect evolution after the game.

8.1 Future work

The future work for enhancing the artificial *Sueca* player starts by testing the results of other reviewed algorithms. In addition, modelling opponents would also be a great improvement through machine learning techniques. This idea combines with Monte-Carlo Methods, since it would decrease the numerous sampling requirements. Furthermore, considering the gap of social robots on elderly population, as reviewed in this thesis, it would be interesting to target this *Sueca* player for older adults.

Bibliography

[1] Patrícia Alves-Oliveira, Sofia Petisca, Srinivasan Janarthanam, Helen Hastie, and Ana Paiva. "How do you imagine robots?" Childrens ' expectations about robots. 2014.

[2] S Bahadori, A Cesta, G Grisetti, L Iocchi, R Leone, and D Nardi. RoboCare: an Integrated Robotic System for the Domestic Care of the Elderly. (December 2002):1–16.

[3] Joost Broekens, Marcel Heerink, and Henk Rosendal. Assistive social robots in elderly care: a review Assistive social robots. 8(2), 2009. doi: 10.4017/gt.2009.08.02.002.00.

[4] Michael Buro, Jeffrey R Long, Timothy Furtak, and Nathan Sturtevant. Improving State Evaluation, Inference, and Search in Trick-Based Card Games. pages 1407–1413.

[5] Murray Campbell, A Joseph Hoane, and Feng-hsiung Hsu. Deep Blue. 134:57–83, 2002.

[6] Ginevra Castellano, Iolanda Leite, André Pereira, Carlos Martinho, Ana Paiva, and Peter W Mcowan. Affect recognition for interactive companions: challenges and design in real world scenarios. pages 89–98, 2010. doi: 10.1007/s12193-009-0033-5.

[7] Peter I Cowling, Edward J Powley, Daniel Whitehouse, and Student Member. Information Set Monte Carlo Tree Search. 4(2):120–143, 2012.

[8] J Dias, Samuel Mascarenhas, and Ana Paiva. Fatima modular: Towards an agent architecture with a generic appraisal framework. *International Workshop on Standards for Emotion Modeling*, pages 1–8, 2011. ISSN 16113349. doi: 10.1007/978-3-319-12973-0{_}3. URL `http://www.lorentzcenter.nl/lc/web/2011/464/presentations/Dias.pdf`.

[9] Boris Egloff. The independence of positive and negative affect depends on the affect measure. *Personality and Individual Differences*, 25(6):1101–1109, 1998. ISSN 01918869. doi: 10.1016/S0191-8869(98)00105-6. URL `http://linkinghub.elsevier.com/retrieve/pii/S0191886998001056`.

[10] David Ferrucci, Eric Brown, Jennifer Chu-carroll, James Fan, David Gondek, Aditya A Kalyanpur, Adam Lally, J William Murdock, Eric Nyberg, and John Prager. Building Watson: An Overview of the DeepQA Project. pages 59–79, 2010.

[11] Ian Frank and David Basin. Search in games with incomplete information: a case study using Bridge card play. *Artificial Intelligence*, 3702(97), 1998.

[12] Masahiro Fujita. AIBO: Toward the Era of Digital. (1), 1983.

[13] Timothy Furtak and Michael Buro. Recursive Monte Carlo Search for Imperfect Information Games.

[14] Sylvain Gelly and David Silver. Monte-Carlo tree search and rapid action value estimation in computer Go. *Artificial Intelligence*, 175(11):1856–1875, July 2011. ISSN 00043702. doi: 10.1016/j.artint.2011.03.007. URL `http://linkinghub.elsevier.com/retrieve/pii/S000437021100052X`.

[15] Matthew L Ginsberg. GIB: Imperfect Information in a Computationally Challenging Game. 14:303–358, 2001.

[16] Birgit Graf, Matthias Hans, and Rolf D Schraft. Care-O-bot II — Development of a Next Generation Robotic Home Assistant. pages 193–205, 2004.

[17] Chad Harms and Frank Biocca. Internal Consistency and Reliability of the Networked Minds Measure of Social Presence. *Seventh Annual International Workshop: Presence*, 2004. URL `http://cogprints.org/7026/`.

[18] Peter H. Kahn, Hiroshi Ishiguro, Batya Friedman, and Takayuki Kanda. What is a human? - Toward psychological benchmarks in the field of human-robot interaction. In *Proceedings - IEEE International Workshop on Robot and Human Interactive Communication*, pages 364–371, 2006. ISBN 1424405653. doi: 10.1109/ROMAN.2006.314461.

[19] Iolanda Leite. *iCat, the Chess Tutor An Affective Game Buddy Based on Anticipatory Mechanisms*. PhD thesis, Instituto Superior Técnico, 2007.

[20] Iolanda Leite, Samuel Mascarenhas, and Carlos Martinho. "Why can't we be friends?" An empathic game companion for long-term interaction. .

[21] Iolanda Leite, André Pereira, Carlos Martinho, and Ginevra Castellano. Towards an Empathic Chess Companion. .

[22] Jeffrey Long and Michael Buro. Real-Time Opponent Modelling in Trick-Taking Card Games. pages 617–622, 2009.

[23] Jeffrey Long, Nathan R Sturtevant, Michael Buro, and Timothy Furtak. Understanding the Success of Perfect Information Monte Carlo Sampling in Game Tree Search. pages 134–140, 2010.

[24] John Nash. *Non-Cooperative Games*. PhD thesis, 1950.

[25] Martin J Osborne. *A Course in Game Theory*. 2011. ISBN 0262650401.

[26] André Pereira. *Socially Present Agents for Tabletop Games*. PhD thesis, 2014.

[27] André Pereira, Rui Prada, and Ana Paiva. Socially Present Board Game Opponents.

[28] Martha E Pollack, Sandra Engberg, Judith T Matthews, Jacqueline Dunbar-jacob, Colleen E Mccarthy, and Sebastian Thrun. Pearl: A Mobile Robotic Assistant for the Elderly. 2002.

[29] Marc Ponsen, Geert Gerritsen, and Guillaume Chaslot. Integrating Opponent Models with Monte-Carlo Tree Search in Poker Monte-Carlo Tree Search. (Coulom 2006):37–42, 2008.

[30] Tiago Ribeiro, Eugenio Tullio, Lee J Corrigan, Aidan Jones, Ruth Aylett, Ginevra Castellano, and Ana Paiva. Developing Interactive Embodied Characters using the Thalamus Framework: A Collaborative Approach. pages 1–10, .

[31] Tiago Ribeiro, Eugenio Tullio, and Ana Paiva. From Thalamus to Skene: High-level behaviour planning and managing for mixed-reality characters. .

[32] Stuart Russell and Peter Norvig. Artificial Intelligence: A Modern Approach. chapter 5 - Advers, pages 161 – 190. 3rd edition, 2009.

[33] Kristin E Schaefer. *The perception and measurment of human-robot trust*. PhD thesis, 2013.

[34] Jonathan Schaeffer, Robert Lake, Paul Lu, and Martin Bryant. CHINOOK The World Man-Machine Checkers Champion. 17(1):21–29, 1996.

[35] Yoav Shoham and Kevin Leyton-brown. Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations. 2010.

[36] Introduction Should. Granny and the robots: Ethical issues in robot care for the elderly. pages 1–43, 2010.

[37] Walter Dan Stiehl, Cynthia Breazeal, Kuk-hyun Han, Jeff Lieberman, Levi Lalla, Allan Maymin, Jonathan Salinas, Robert Toscano, Cheng Hau Tong, Aseem Kishore, Matt Berlin, and Jesse Gray. The Huggable: A Therapeutic Robotic Companion for Relational, Affective Touch. page 2005, 2005.

[38] Toshiyo Tamura, Satomi Yonemitsu, Akiko Itoh, Daisuke Oikawa, Akiko Kawakami, Yuji Higashi, Toshiro Fujimooto, and Kazuki Nakajima. Is an Entertainment Robot Useful in the Care of Elderly People With Severe Dementia ? 59(1):83–85, 2004.

[39] Kazuyoshi Wada and Takanori Shibata. Living With Seal Robots — Its Sociopsychological and Physiological Influences on the Elderly at a Care House. 23(5):972–980, 2007.

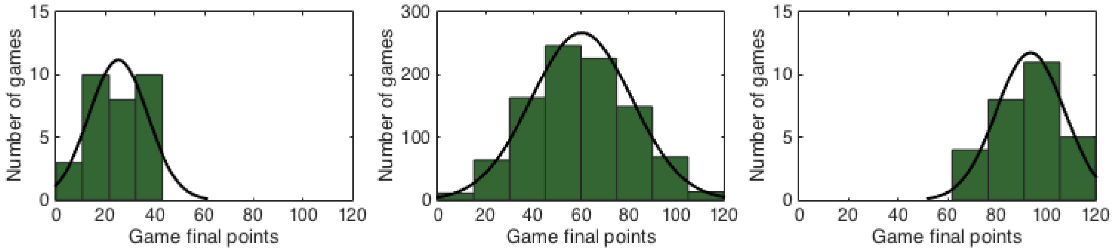
[40] Kazuyoshi Wada, Takanori Shibata, Tomoko Saito, and Kazuo Tanie. Effects of Robot Assisted Activity to Elderly People who Stay at a Health Service Facility for the Aged. *Conference on Intelligent Robots and Systems*, 2003.

[41] Kazuyoshi Wada, Takanori Shibata, Tomoko Saito, Kayoko Sakamoto, and Kazuo Tanie. Psychological and Social Effects of One Year Robot Assisted Activity on Elderly People at a Health Service Facility for the Aged. (April), 2005.

[42] Martin Zinkevich, Michael Bowling, and Michael Johanson. Regret Minimization in Games with Incomplete Information. pages 1–14.

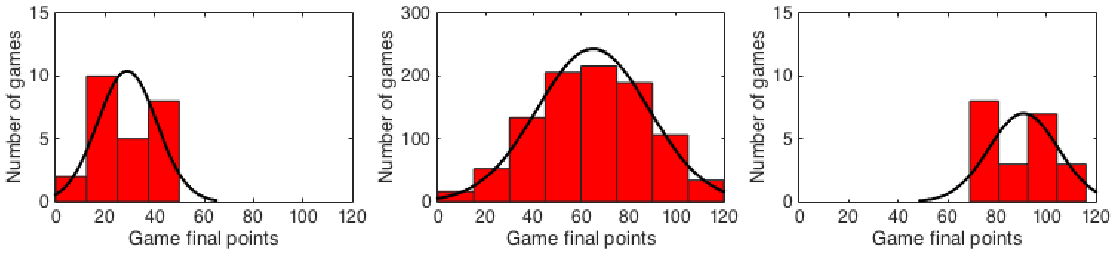
A

Final points histograms for each scenario



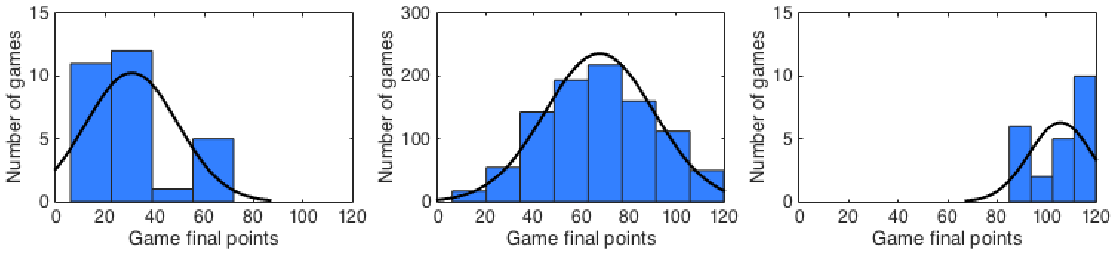
(a) 31 *hard* games out of 1000 (b) 941 *medium* games out of 1000 (c) 28 *easy* games out of 1000

Figure A.1: Histograms of the final points obtained by a team of 2 Rule-based players against 2 Ruled-based players in 1000 games



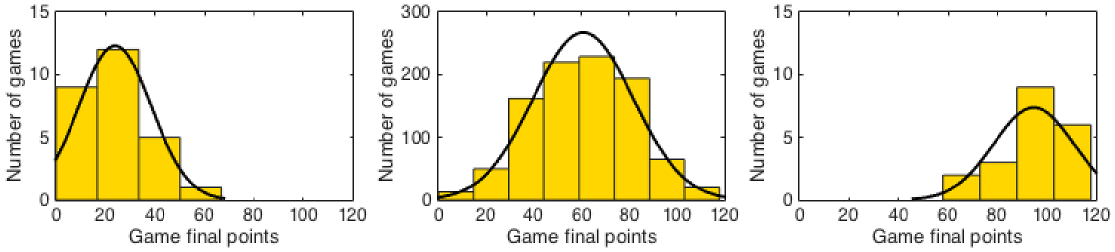
(a) 25 *hard* games out of 1000 (b) 954 *medium* games out of 1000 (c) 21 *easy* games out of 1000

Figure A.2: Histograms of the final points obtained by a team of 1 Rule-based player and 1 Random player against 2 Random players in 1000 games



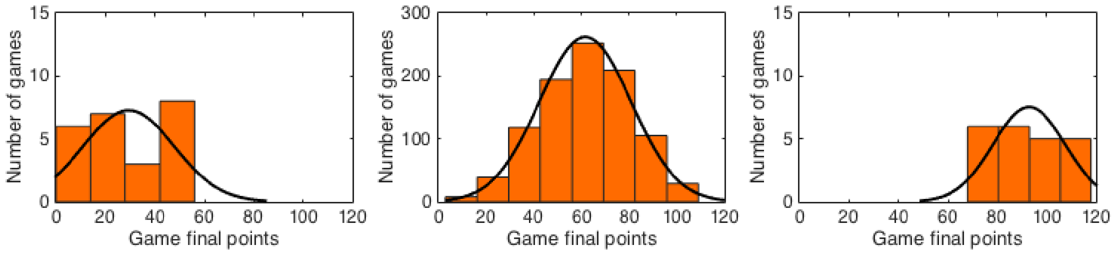
(a) 29 *hard* games out of 1000 (b) 948 *medium* games out of 1000 (c) 23 *easy* games out of 1000

Figure A.3: Histograms of the final points obtained by a team of 2 Rule-based players against 2 Random players in 1000 games



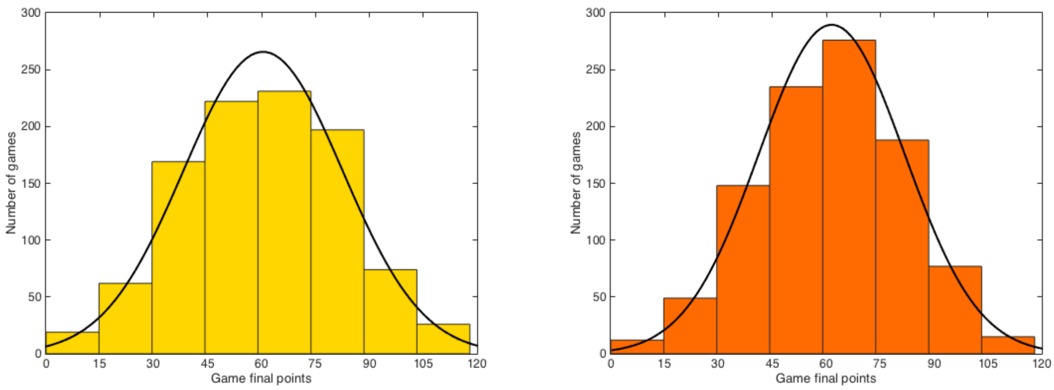
(a) 27 *hard* games out of 1000 (b) 953 *medium* games out of 1000 (c) 20 *easy* games out of 1000

Figure A.4: Histograms of the final points obtained by a team of 1 Trick player and 1 Rule-based player against 2 Rule-based players in 1000 games



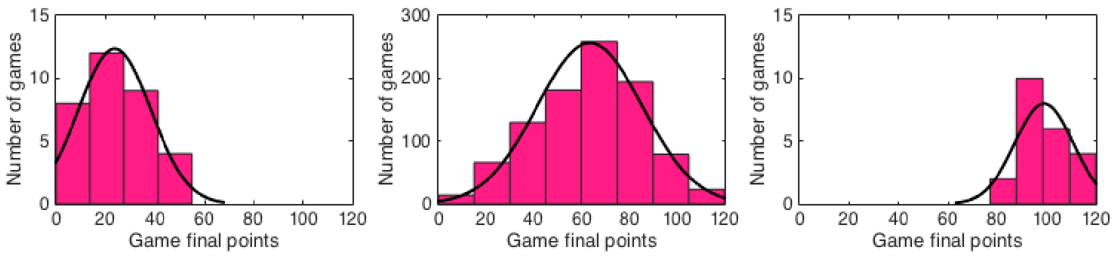
(a) 24 *hard* games out of 1000 (b) 954 *medium* games out of 1000 (c) 22 *easy* games out of 1000

Figure A.5: Histograms of the final points obtained by a team of 2 Trick players against 2 Rule-based players in 1000 games



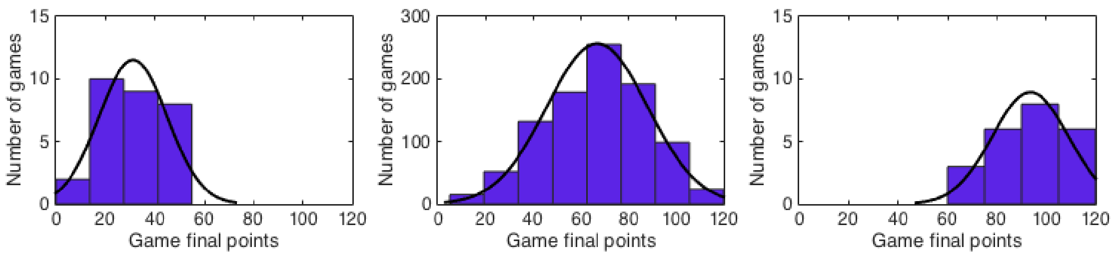
(a) Final points histogram of the team with 1 Trick player and 1 Rule-based against 2 Ruled-based players in 1000 games, scenario (d)

(b) Final points histogram of the team with 2 Trick players against 2 Ruled-based players in 1000 games, scenario (e)



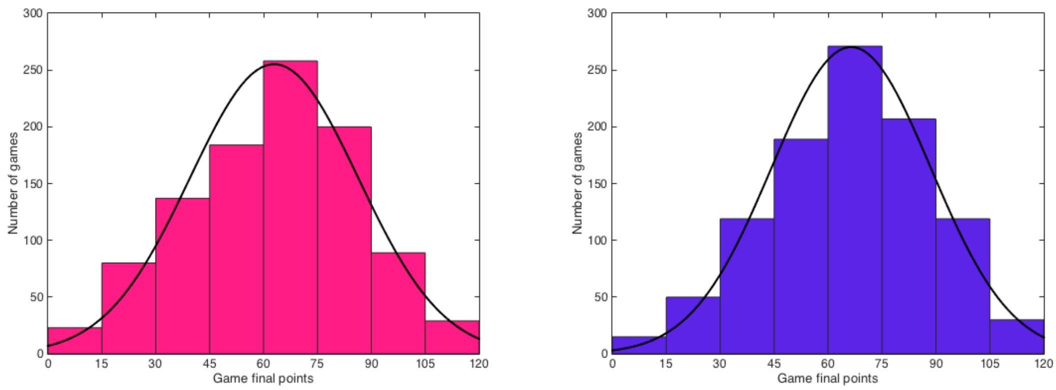
(a) 33 *hard* games out of 1000 (b) 945 *medium* games out of 1000 (c) 22 *easy* games out of 1000

Figure A.7: Histograms of the final points obtained by a team of 1 Deep-1 player and 1 Rule-based player against 2 Rule-based players in 1000 games



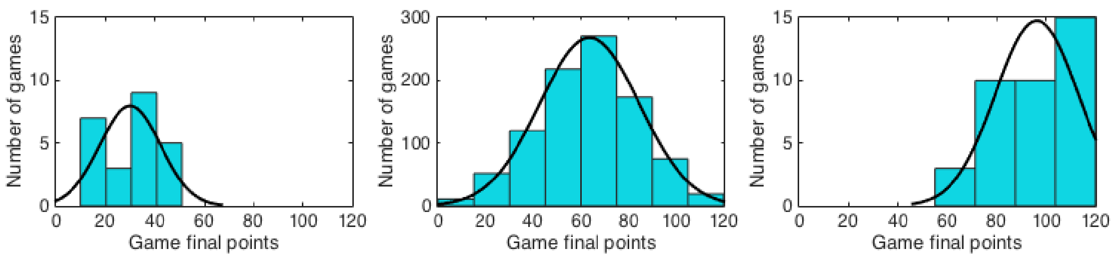
(a) 29 *hard* games out of 1000 (b) 948 *medium* games out of 1000 (c) 23 *easy* games out of 1000

Figure A.8: Histograms of the final points obtained by a team of 2 Deep-1 players against 2 Rule-based players in 1000 games



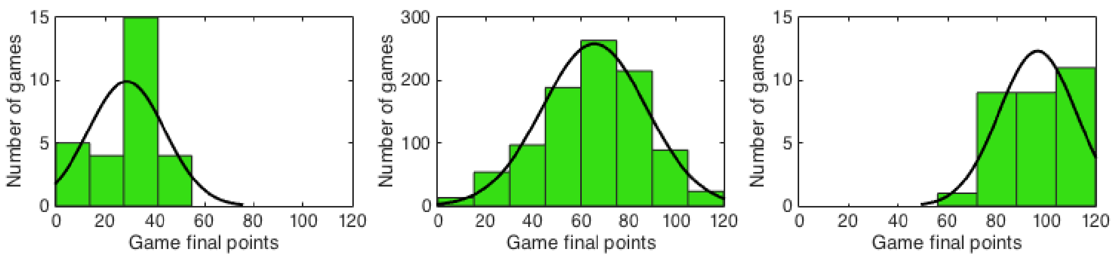
(a) Final points histogram of the team with 1 Deep-1 player and 1 Rule-based against 2 Ruled-based players in 1000 games, scenario (f)

(b) Final points histogram of the team with 2 Deep-1 players against 2 Ruled-based players in 1000 games, scenario (g)



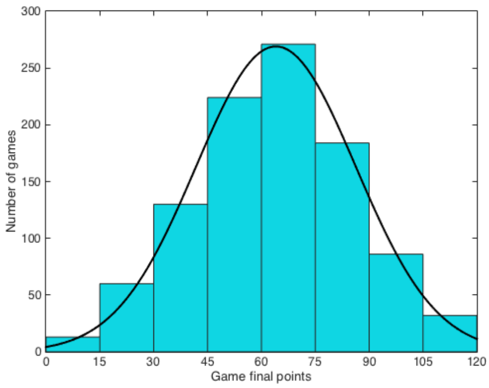
(a) 24 *hard* games out of 1000 (b) 938 *medium* games out of 1000 (c) 38 *easy* games out of 1000

Figure A.10: Histograms of the final points obtained by a team of 1 Deep-2 player and 1 Rule-based player against 2 Rule-based players in 1000 games

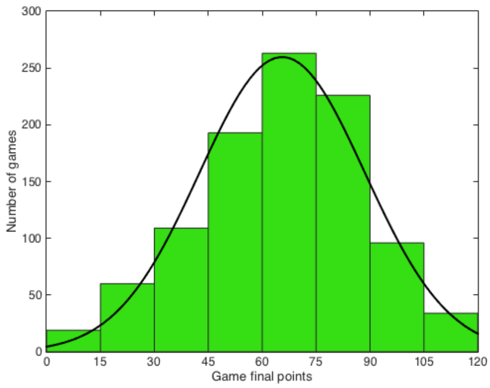


(a) 28 *hard* games out of 1000 (b) 942 *medium* games out of 1000 (c) 30 *easy* games out of 1000

Figure A.11: Histograms of the final points obtained by a team of 2 Deep-2 players against 2 Rule-based players in 1000 games



(a) Final points histogram of the team with 1 Deep-2 player and 1 Rule-based against 2 Ruled-based players in 1000 games, scenario (h)



(b) Final points histogram of the team with 2 Deep-2 players against 2 Ruled-based players in 1000 games, scenario (i)

B

Questionnaires used in User Studies

Session:
ID:
Date:

Obrigada pelo interesse em participar!

Antes de iniciar o jogo de Sueca, pedimos que responda a este questionário cujas instruções se encontram na página seguinte.

Este questionário é anónimo e confidencial, isto significa que as respostas não serão ligadas a si, sendo que ninguém terá acesso ao conteúdo das mesmas, com exceção dos investigadores deste projeto.
Assim, pedimos que seja sincero(a) em todas as suas respostas.

Se tiver alguma dúvida durante o questionário, chame a investigadora presente na sala.

Esta escala consiste num conjunto de palavras que descrevem diferentes emoções. Para cada uma das emoções, **escreva um número de 1 a 5** de acordo com *o que sente neste momento.*

1 Muito ligeiramente ou nada
2 Um Pouco
3 Moderadamente
4 Bastante
5 Extremamente

Culpado_____

Determinado_____

Excitado_____

Irritado_____

Trémulo _____

Amedrontado _____

Atormentado _____

Ativo _____

Caloroso _____

Encantado _____

Perturbado _____

Orgulhoso _____

Inspirado _____

Remorsos _____

Assustado _____

Repulsa _____

Entusiasmado _____

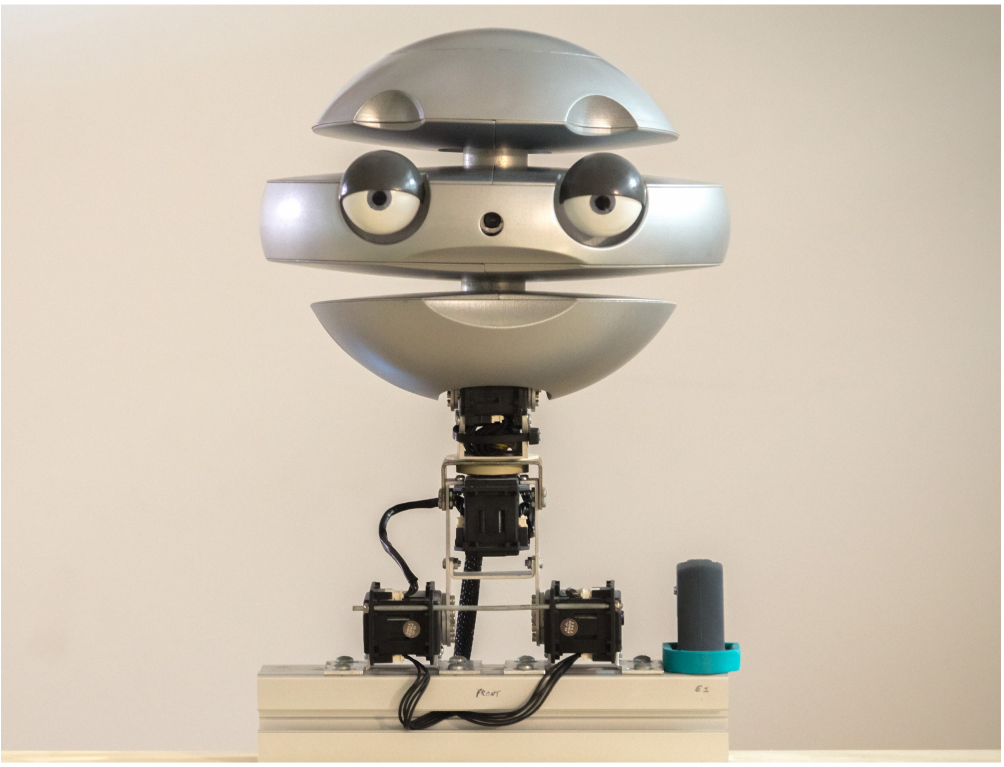
Nervoso _____

Agradavelmente surpreendido _____

Interessado _____

2 /6

Robot Emys



3/6

De acordo com as suas expectativas, avalie os seguintes items sobre o Robot Emys, colocando um X no círculo que melhor representa a sua opinião:

A percentagem de tempo que este Robot…	**0%**	**10%**	**20%**	**30%**	**40%**	**50%**	**60%**	**70%**	**80%**	**90%**	**100%**
Tem erros	O	O	O	O	O	O	O	O	O	O	O
Um bom companheiro de equipa	O	O	O	O	O	O	O	O	O	O	O
Previsível	O	O	O	O	O	O	O	O	O	O	O
Fiel	O	O	O	O	O	O	O	O	O	O	O
Avaria	O	O	O	O	O	O	O	O	O	O	O
Responsável	O	O	O	O	O	O	O	O	O	O	O
Considerado parte da equipa	O	O	O	O	O	O	O	O	O	O	O
Toma decisões sensatas	O	O	O	O	O	O	O	O	O	O	O
Agradável	O	O	O	O	O	O	O	O	O	O	O
Desencaminha-se por mudanças inesperadas no ambiente envolvente	O	O	O	O	O	O	O	O	O	O	O
Funciona com sucesso	O	O	O	O	O	O	O	O	O	O	O
Autónomo	O	O	O	O	O	O	O	O	O	O	O
Comunica claramente	O	O	O	O	O	O	O	O	O	O	O
Consegue desempenhar várias funções ao mesmo tempo	O	O	O	O	O	O	O	O	O	O	O
Sabe a diferença entre amigo e inimigo	O	O	O	O	O	O	O	O	O	O	O
Corresponde ao que é esperado na tarefa	O	O	O	O	O	O	O	O	O	O	O
Executa uma tarefa melhor do que um usuário humano principiante	O	O	O	O	O	O	O	O	O	O	O

4/6

A percentagem de tempo que este robot…	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Comunica abertamente	O	O	O	O	O	O	O	O	O	O	O
Consciente	O	O	O	O	O	O	O	O	O	O	O
Comunica com as pessoas	O	O	O	O	O	O	O	O	O	O	O
Dependente	O	O	O	O	O	O	O	O	O	O	O
Amigável	O	O	O	O	O	O	O	O	O	O	O
Tem capacidades adequadas de tomada de decisão	O	O	O	O	O	O	O	O	O	O	O
Protege pessoas	O	O	O	O	O	O	O	O	O	O	O
Consegue trabalhar com pessoas	O	O	O	O	O	O	O	O	O	O	O
Dá informação apropriada	O	O	O	O	O	O	O	O	O	O	O
Vivo	O	O	O	O	O	O	O	O	O	O	O
Um bom companheiro de equipa	O	O	O	O	O	O	O	O	O	O	O
Desempenha as suas funções na tarefa	O	O	O	O	O	O	O	O	O	O	O
Age como pertencente à equipa	O	O	O	O	O	O	O	O	O	O	O
Dá feedback	O	O	O	O	O	O	O	O	O	O	O
Guarda informações privadas	O	O	O	O	O	O	O	O	O	O	O
Requer manutenção frequente	O	O	O	O	O	O	O	O	O	O	O
Não responsivo	O	O	O	O	O	O	O	O	O	O	O
Apoiante	O	O	O	O	O	O	O	O	O	O	O
Avisa as pessoas de potenciais riscos	O	O	O	O	O	O	O	O	O	O	O
Age de forma coerente	O	O	O	O	O	O	O	O	O	O	O

5/6

A percentagem de tempo que este robot…	**0%**	**10%**	**20%**	**30%**	**40%**	**50%**	**60%**	**70%**	**80%**	**90%**	**100%**
Segue instruções	O	O	O	O	O	O	O	O	O	O	O
Funciona num ambiente de equipa integrado	O	O	O	O	O	O	O	O	O	O	O
Trabalha melhor em equipa	O	O	O	O	O	O	O	O	O	O	O

Chegou ao fim deste questionário. Pedimos que o reveja uma última vez para se assegurar que respondeu a todas as perguntas.
Depois, basta aguardar em silêncio que os outros participantes terminem.

Session:
ID:
Date:

Espero que se tenha divertido a jogar Sueca!

Este é o último questionário que terá que preencher para a atividade terminar.

Relembro que este questionário é anónimo e confidencial, isto significa que as respostas não serão ligadas a si, sendo que ninguém terá acesso ao conteúdo das mesmas, com exceção dos investigadores deste projeto.
Assim, pedimos que seja sincero(a) em todas as suas respostas.

Se tiver alguma dúvida durante o questionário, chame a investigadora presente na sala.

1 /9

Esta escala consiste num conjunto de palavras que descrevem diferentes emoções. Para cada uma das emoções, **escreva um número de 1 a 5** de acordo com *o que sente neste momento.*

1 Muito ligeiramente ou nada
2 Um Pouco
3 Moderadamente
4 Bastante
5 Extremamente

Culpado_____

Determinado_____

Excitado_____

Irritado_____

Trémulo _____

Amedrontado _____

Atormentado _____

Ativo _____

Caloroso _____

Encantado _____

Perturbado _____

Orgulhoso _____

Inspirado _____

Remorsos _____

Assustado _____

Repulsa _____

Entusiasmado _____

Nervoso _____

Agradavelmente surpreendido _____

Interessado _____

Avalie os seguintes items sobre o seu parceiro de sueca, colocando um X no círculo que melhor representa a sua opinião:

A percentagem de tempo que o meu parceiro de sueca...	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Um bom companheiro de equipa	O	O	O	O	O	O	O	O	O	O	O
Previsível	O	O	O	O	O	O	O	O	O	O	O
Fiel	O	O	O	O	O	O	O	O	O	O	O
Responsável	O	O	O	O	O	O	O	O	O	O	O
Considerado parte da equipa	O	O	O	O	O	O	O	O	O	O	O
Toma decisões sensatas	O	O	O	O	O	O	O	O	O	O	O
Agradável	O	O	O	O	O	O	O	O	O	O	O
Desencaminha-se por mudanças inesperadas no ambiente envolvente	O	O	O	O	O	O	O	O	O	O	O
Autónomo	O	O	O	O	O	O	O	O	O	O	O
Comunica claramente	O	O	O	O	O	O	O	O	O	O	O
Consegue desempenhar várias funções ao mesmo tempo	O	O	O	O	O	O	O	O	O	O	O
Sabe a diferença entre amigo e inimigo	O	O	O	O	O	O	O	O	O	O	O
Corresponde ao que é esperado na tarefa	O	O	O	O	O	O	O	O	O	O	O
Comunica abertamente	O	O	O	O	O	O	O	O	O	O	O
Consciente	O	O	O	O	O	O	O	O	O	O	O
Comunica com as pessoas	O	O	O	O	O	O	O	O	O	O	O
Dependente	O	O	O	O	O	O	O	O	O	O	O
Amigável	O	O	O	O	O	O	O	O	O	O	O

3/9

A percentagem de tempo que o meu parceiro de sueca...	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Tem capacidades adequadas de tomada de decisão	O	O	O	O	O	O	O	O	O	O	O
Protege pessoas	O	O	O	O	O	O	O	O	O	O	O
Consegue trabalhar com pessoas	O	O	O	O	O	O	O	O	O	O	O
Dá informação apropriada	O	O	O	O	O	O	O	O	O	O	O
Vivo	O	O	O	O	O	O	O	O	O	O	O
Um bom companheiro de equipa	O	O	O	O	O	O	O	O	O	O	O
Desempenha as suas funções na tarefa	O	O	O	O	O	O	O	O	O	O	O
Age como pertencente à equipa	O	O	O	O	O	O	O	O	O	O	O
Dá feedback	O	O	O	O	O	O	O	O	O	O	O
Guarda informações privadas	O	O	O	O	O	O	O	O	O	O	O
Não responsivo	O	O	O	O	O	O	O	O	O	O	O
Apoiante	O	O	O	O	O	O	O	O	O	O	O
Avisa as pessoas de potenciais riscos	O	O	O	O	O	O	O	O	O	O	O
Age de forma coerente	O	O	O	O	O	O	O	O	O	O	O
Segue instruções	O	O	O	O	O	O	O	O	O	O	O
Funciona num ambiente de equipa integrado	O	O	O	O	O	O	O	O	O	O	O
Trabalha melhor em equipa	O	O	O	O	O	O	O	O	O	O	O

Tendo em conta a interação que teve com o seu parceiro na Sueca, circule o número que melhor se adequa à sua opinião. *A sua resposta pode ir desde* ***1 (discordo completamente) a 6 (concordo completamente).***

1. **O comportamento do meu parceiro foi muitas vezes em resposta direta ao meu comportamento.**

   1 2 3 4 5 6

2. **O meu parceiro reparou em mim.**

   1 2 3 4 5 6

3. **Eu distrai-me com facilidade do meu parceiro quando aconteciam outras coisas.**

   1 2 3 4 5 6

4. **O meu parceiro teve dificuldade em perceber-me.**

   1 2 3 4 5 6

5. **Os sentimentos do meu parceiro influenciaram o humor da nossa interação.**

   1 2 3 4 5 6

6. **Perceber o meu parceiro foi difícil.**

   1 2 3 4 5 6

7. **O meu parceiro retribuía as minhas ações.**

   1 2 3 4 5 6

8. **As atitudes do meu parceiro influenciaram como eu me senti.**

   1 2 3 4 5 6

9. **Conseguiria descrever os sentimentos meu parceiro com exatidão.**

   1 2 3 4 5 6

10. **Os meus pensamentos foram claros para o meu parceiro.**

    1 2 3 4 5 6

11. **Os pensamentos do meu parceiro foram claros para mim.**

    1 2 3 4 5 6

5/9

A sua resposta pode ir desde 1 (discordo completamente) a 6 (concordo completamente).

12. **O meu parceiro captou a minha atenção.**

1 2 3 4 5 6

13. **As emoções do meu parceiro não foram claras para mim.**

1 2 3 4 5 6

14. **O meu parceiro foi por vezes influenciado pelo meu humor.**

1 2 3 4 5 6

15. **As minhas atitudes influenciaram como o meu parceiro se sentiu.**

1 2 3 4 5 6

16. **O meu parceiro achou que era fácil perceber-me.**

1 2 3 4 5 6

17. **O meu comportamento estava alinhado ao comportamento do meu parceiro.**

1 2 3 4 5 6

18. **Os meus sentimentos influenciaram o humor da nossa interação.**

1 2 3 4 5 6

19. **A minha presença foi óbvia para o meu parceiro.**

1 2 3 4 5 6

20. **O meu parceiro distraia-se facilmente de mim quando aconteciam outras coisas.**

1 2 3 4 5 6

21. **A presença do meu parceiro foi óbvia para mim.**

1 2 3 4 5 6

22. **Eu captei a atenção do meu parceiro.**

1 2 3 4 5 6

A sua resposta pode ir desde 1 (discordo completamente) a 6 (concordo completamente).

23. O comportamento do meu parceiro estava alinhado ao meu comportamento.

1 2 3 4 5 6

24. O meu comportamento foi muitas vezes em resposta direta ao comportamento do meu parceiro.

1 2 3 4 5 6

25. O meu parceiro conseguiria descrever os meus sentimentos com exatidão.

1 2 3 4 5 6

26. Eu reparei no meu parceiro.

1 2 3 4 5 6

27. O meu parceiro conseguiu perceber como é que eu me senti.

1 2 3 4 5 6

28. Mantive-me concentrado no meu parceiro durante a interação.

1 2 3 4 5 6

29. Eu não recebi toda a atenção do meu parceiro.

1 2 3 4 5 6

30. O meu parceiro não recebeu toda a minha atenção.

1 2 3 4 5 6

31. Eu retribuía as ações do meu parceiro.

1 2 3 4 5 6

32. O meu parceiro achou que era fácil perceber-me.

1 2 3 4 5 6

33. Eu consegui perceber como é que o meu parceiro se sentiu.

1 2 3 4 5 6

A sua resposta pode ir desde 1 (discordo completamente) a 6 (concordo completamente).

34. O meu parceiro manteve-se concentrado em mim durante a interação.

1 2 3 4 5 6

35. As minhas emoções não foram claras para o meu parceiro.

1 2 3 4 5 6

36. Às vezes senti-me influenciado pelo humor do meu parceiro.

1 2 3 4 5 6

Assinale com um X a opção que melhor se adequa a si:

1. Já tinha interagido com o seu parceiro de sueca anteriormente?

___ Nunca tinha visto o meu parceiro no passado.

___ Já tinha visto o meu parceiro no passado, mas nunca interagi com ele(a) diretamente.

___ Já conhecia o meu parceiro.

2. Qual o seu domínio de jogo na Sueca?

___ Inexistente: nunca tinha jogado, aprendi hoje.

___ Médio: já joguei no passado mas não dominava ou não me lembrava de todas as regras.

___ Alto: Sei jogar e jogo às vezes com os meus amigos.

___ Profissional: Sei tudo sobre a Sueca e participo competições.

Complete:

Idade: _____

Sexo:

___ Feminino

___ Masculino

Chegou ao fim deste questionário. Pedimos que o reveja uma última vez para se assegurar que respondeu a todas as perguntas.
Depois, basta aguardar em silêncio que os outros participantes terminem.
Obrigada pela participação, foi um contributo para o desenvolvimento da ciência.

9/9