EMYS: a social robot that plays “Sueca”

Filipa Correia
Instituto Superior Técnico
filipacorreia@tecnico.ulisboa.com

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, Social Robotics, Socially Intelligent Behaviour

1. INTRODUCTION

Games have been a subject of particular interest to the Artificial Intelligence (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. 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[9]


In addition, there is a 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. Some agents of this nature illustrate this idea, such as the iCat chess tutor [4] and the EMotive headY System (EMYS) Risk player [5]. Additionally, these last two examples explore different challenges from an Human-Robot Interaction (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 trick-taking 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 partner. These two roles together have not yet been studied in an artificial embodied game player.

2. AI FOR 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. Although most trick-taking card games count the number of winning tricks to determine the winner, Sueca assigns points to the cards and by winning each trick, the team collects the trick points. The rules are quite similar to any other trick-taking games:

- To follow the suit of the first played card in the trick (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.
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 Perfect Information Monte-Carlo (PIMC). Other techniques (for instance Imperfect Information Monte-Carlo (IIMC) or Information Set Monte-Carlo Tree Search (ISMCTS)) 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.

PIMC algorithm is illustrated in Figure 1 for the 8th trick of a Sueca game, where players still hold two cards in their hands. The idea is to sample 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.

2.1 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 lead suit 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.

2.2 MinMax Algorithm

As mentioned above, PIMC has to calculate the reward of playing a card, for each sampled world. 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.

2.3 Measuring parametrization effects

The PIMC algorithm, implemented as previously described, 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 number of different sampled distributions (\( N \)) 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.

Furthermore, a Rule-based player was created in order to provide a comparison between the different parametrizations of PIMC. This baseline agent tries to roughly reproduce the reasoning of a non-professional human player with predefined rules in its deliberation procedure, instead of using hard computational algorithms.

The Trick 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:

\[
\begin{align*}
    u_1 = \begin{cases} 
    \text{teamPoints}, & \text{teamPoints} > \text{opponentTeamPoints}; \\
    -\text{opponentTeamPoints}, & \text{opponentTeamPoints} > \text{teamPoints}; \\
    \text{teamPoints} < \text{opponentTeamPoints}. & 
    \end{cases}
\end{align*}
\]

In order to observe the Trick player performance, 1000 games were executed between a team with 1 Trick player and 1 Rule-based player against 2 Rule-based players. The Favourable Games Rate (FGR) of the team with 1 Trick and 1 Rule-based was 52.8% (50.5% won games and 2.3% drawn games).

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. 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 1.

To analyse the Deep-1 player performance, 1000 games were executed between a team with 1 Deep-1 player and 1 Rule-based player against 2 Rule-based players. The measured FGR of the team with 1 Deep-1 and 1 Rule-based was 58.3% (57.6% won games and 0.7% drawn games). These results evidenced that the Deep-1 player outperforms the Trick player.

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 2.
Figure 1: PIMC algorithm illustrated to exemplify the choosing procedure in the 8th trick

\[
u_2 = \begin{cases} 
2, & \text{teamPoints} > 90 \\
1, & \text{teamPoints} > 60 \\
0.1, & \text{teamPoints} > 30 \\
-2, & \text{opponentTeamPoints} > 90 \\
-1, & \text{opponentTeamPoints} > 60 \\
-0.1, & \text{opponentTeamPoints} > 30 
\end{cases}
\] (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, 1000 games were executed with a team of 1 Deep-2 player and 1 Rule-based player against 2 Rule-based players. The overall FGR of the team with 1 Deep-2 player and 1 Rule-based was 58.6% (57.3% won games and 1.3% drawn games). These results evidenced that 1 Deep-2 player slightly outperforms the Deep-1 player.

Besides analysing the results of adding only one player, from each type, to a team, we repeated the same tests for a team of two equal players against 2 Rule-based players. The overall FGR of the team with 1 Deep-2 player and 1 Rule-based was 58.6% (57.3% won games and 1.3% drawn games). Theses results evidenced that 1 Deep-2 player slightly outperforms the Deep-1 player.

3. EMYS: THE SUECA PLAYER

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 answers clearly to the first goal, in the same way the current session explains how the second goal has been achieved.

The architecture presented in Figure 2 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 2: 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 workflow 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 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 and TTS and additional game information from the Game Application.

The previously described architecture is instantiated as shown in Figure 3 and the blue modules are thalamus communicating entities. This concept arises from the Thalamus Framework [6], which enables the usage of entities that can be registered at runtime in a server in order 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 inherit from the ThalamusClient class and implement the interfaces of the messages that the
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 the plays of artificial players.

The chosen Behaviour Planner is Skene [7], 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 previous section. Moreover, the implementation of FAtiMA module, as decision maker of our Sueca player, is carefully described in the following subsection.

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 interact, is indeed composed of distinct modules that make the robot talk, move, animate, gaze at some point or glance at another. For this reason, the architecture, presented in Figure 3, 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.

3.1 Sueca behaviours

The analysis of the card game players on user-centred studies, 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.

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.

3.2 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 3, 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 [1]. Perceptions can be updated by changing the values of 6 appraisal variables (desirability, desirability for other, success probability, failure probability, praiseworthy 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 5.

Figure 3: System architecture using modules

Figure 4: Categories and subcategories of the utterances list

Figure 5: Distinction of used and non used FAtiMA emotions

One first approach was to subcategorize utterances with emotional states, however, most of the annotated behaviours by human players evidenced they revealed their emotions in few 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 Pity, 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, 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 4.

3.3 Enhancing the game interface

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 at all captured by human players. 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.

4. USER STUDIES

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.

4.1 Methodology and Procedures

Each session last an hour and included 3 participants to play 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 contained two parts: the PANAS Questionnaire [2] and Human-Robot Trust Questionnaire [8]. 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. Lastly, participants answered another questionnaire divided into four parts: the PANAS Questionnaire, Human-Robot Trust Questionnaire, the Networked Minds Questionnaire [3] and some demographic questions.

4.2 Samples Description

A group of 60 participants were included in this study with a mean age of 24.31 ± 3.852. Out of the 60 subjects, 40 played the game with a human partner and 20 played with EMYS as their team player. 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.

4.3 Trust

The trust in the partner was measured by the answers of each individual, before and after the game session, to the Human-Robot Trust Questionnaire. 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 6 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.
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.

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 proves an non significant effect on the trust measure. Moreover, the effect of both condition and game result also proved to be non significant with $p = 0.507$.

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. The condition presented the following statistical effects of 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 not 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 8 shows these discrepancies, demonstrating the perceived affective understanding and perceived emotional interdependence were higher in human partners.
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.

4.5 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 affect 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 question, a Mixed ANOVA has been 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, the time levels for each condition did not present a significant effect on the positive affect, \( p = 0.488 \).

Figure 9 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 question, a Mixed ANOVA has been 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 \( p = 0.184 \).

Overall, the difference on the trust levels between conditions suggested that humans cannot yet trust in robots in the game playing scenario of 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, were 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, the negative affect did not change after the game with EMYS, however, the positive affect increased after the game, suggesting it was a pleasing experience for participants.

---

**Figure 8:** 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.

**Figure 9:** Evolution of the positive affect between the two time levels

**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 between different partners.
5. CONCLUSIONS

This thesis addressed three main contributions aligned with the presented problems. 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.

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.

REFERENCES