

Active Learning and Case-Based Reasoning for the Deceptive Play in the Card Game of Truco

Daniel P. Vargas, Gustavo B. Paulus, and Luis A. L. Silva

Graduate Program in Computer Science
Federal University of Santa Maria – UFSM
Santa Maria – RS, Brazil

{pinheiro.vargas, gustavobpaulus, silva.luisalvaro}@gmail.com

Abstract. Deception is an essential behavior in many card games. Despite this fact, it is not trivial to capture the intent of a human strategist when making deceptive decisions. That is even harder when dealing with deception in card games, where components of uncertainty, hidden information, luck and randomness introduce the need of case-based decision making. Approaching this problem along with the investigation of the game of Truco, a quite popular game in Southern regions of South America, this work presents an approach that combines active learning and Case-Based Reasoning (CBR) in which agents request a human specialist to review a reused game action retrieved from a case base containing played Truco hands. That happens when the agents are confronted with game situations that are identified as opportunities for deception. The goal is to actively capture problem-solving experiences in which deception can be used, and later employ such case knowledge in the enhancement of the deceptive capabilities of the Truco agents. Experimental results show that the use of the learned cases enabled different kinds of Truco agents to play more aggressively, being more deceptive and performing a larger number of successful bluffs.

Keywords: Deception, Case-Based Reasoning, Active Learning.

1 Introduction

Deception involves a deliberate attempt to introduce in another person a false belief or belief in which the deceiver considers false [1, 2]. Such deceptive behavior can be modeled as a) concealment, aiming to hide/omit the truth and b) simulation, whose purpose is to show the untruth [3]. To conceal the deceiver acts by withholding information and omitting the truth. To simulate, in addition to the retention of genuine information, unreal information is presented as being legitimate. Among the various forms of deception is the bluff, where deception and bluffing is interchangeably used in this work. In the context of a card game, a bluff is an action where players, to deceive their opponents, seek to make an illusory impression of strength when they hold weak hands. Alternatively, players may try to show that their strong hands have little value in the game.

To card games with hidden, stochastic and imperfect information, acting deceptively is an essential strategy for players to succeed. The use of deceptive moves can also be closely related to the nature of some popular games, and the entertainment that it introduces in the game disputes. To do so, players should be able to identify what the best opportunities for the use of deception are, considering the strength of their hand and their betting history in order to make themselves as unpredictable as possible [4]. In general, real-world situations present complex characteristics for the modeling of deceptive agents, such as the need for learning and decision-making with a small number of training examples, for instance.

For the development of agents capable of acting deceptively in card games, this work explores Case-based Reasoning (CBR) [5]. With relevant explanatory capabilities, CBR combines learning and problem-solving with the use of specific knowledge captured in the form of cases. In particular, this technique has supported the development of agents which competitively play Poker [6, 7]. In this line of research, this paper extends past work [8-10] in the CBR modeling of a popular game in the Southern regions of South America, a game that is under-investigation in Artificial Intelligence (AI): the game of Truco [11].

CBR allows continuous learning by retaining concrete problem-solving experiences in a reusable case base. Despite this fact, it is not simple to capture and label the intention of human players when making deceptive moves in card game. To approach this problem, active learning [12] is investigated in the analysis of Truco opportunities for being deceptive, and the consequent collection of such problem-solving experiences in a case base. Then, the acquired case knowledge is used to equip different kinds of CBR agents to make deceptive actions. In the proposed approach, the case learning is focused on the review of decisions and retention of cases in the case base. As a result of the implemented solution reuse policy, whenever a game action is reused by the agent, if a certain pre-established learning criterion is met, the agent requests for a human expert to review the reused game action and the current game state. If the solution presented by the reuse policy is not considered to be the most effective according to the judgment of the domain specialist, the expert suggests a game action to be played (deceptive or not). With attention to the capture and reuse of deceptive game actions from human players, the contributions of this paper are: i) the exploration of active learning to support the retention of case problem situations in which deceptive moves can be used, ii) the performance evaluation of deceptive Truco agents configured according to alternative solution reuse policies, and iii) the analysis of the resulting game playing behavior of the implemented agents when using case bases storing the collected problem-solving experiences.

2 Background to This Work

CBR [5] combines learning and problem-solving with the use of knowledge obtained from concrete problem-solving experiences. Learning in CBR aims to acquire, modify or improve different knowledge repositories [13], where the enhancement of the case base is often sought in different applications. In doing so, it is possible to explore the

automatic case elicitation (ACE) technique [14]. This technique focuses on the system's ability to explore its domain in real time and automatically collect new cases. Another technique is the learning by observation, also referred as demonstration learning or imitation learning. In such learning modality, the system learns to perform a certain behavior by observing an expert to act [15]. The first learning stage is the acquisition of cases from the expert demonstrations. The second stage is the resolution of a problem using the case base collected from the observations [16]. An alternative to learning by observation is the active learning, where the goal is to obtain greater quality in the learning process considering the smallest possible number of labeled instances. Active learning tries to overcome the labeling and data distribution bottlenecks by allowing the learner to intelligently choose the most representative instances. This model allows requesting that a human expert present a solution to the problem. Later it allows adding the resolved instances to the training set.

2.1 CBR, Active Learning and Games

Active learning and CBR have been explored in a number of digital game applications. In the SMILe - Stochastic Mixing Iterative Learning [17] game, SMILe controls the agent while observing the specialist behavior. When the game iteration ends, SMILe uses the collected observations to train a new policy that can be used in subsequent game iterations. The DAgger – Dataset Aggregation [18] algorithm enhances SMILe by preventing the agent from selecting actions using outdated policies. In doing so, the agent updates a single policy learned each iteration. In both SMILe and DAgger, the control to determine whether the player is the agent or the specialist is defined probabilistically.

The SALT algorithm – Selective Active Learning from Traces [19, 20] allows the learner agent to perform a task, and when it is determined that the agent has left the space for which it has training data, the control is assigned to an expert. As in SMILe and DAgger, the focus is on the collection of training data for the set of states that are expected to be found during testing. Unlike SMILe and DAgger, control in SALT is assigned to the specialist only when the agent leaves the state-space of the training set. The training data is generated only when the specialist is in control, reducing the specialist's cognitive load.

With regard to expert consultation strategies, in [21], the retrieval of most similar cases is used to determine a game action to be taken according to a vote. Considering the average similarity value of the cases retrieved from the last five decisions made in the game and a coefficient obtained from a linear regression, which determines whether the similarities are increasing or decreasing during the last performed movements, the CBR agent gives the game control to the human expert. This happens whenever the mean similarity is increasingly moving away from the space of known situations. The expert plays until the states of the game are familiar again. To avoid continuous changes between the CBR agent and the human specialist, each one has to perform certain minimum play before giving control to the other.

In [22] and [15], a similarity threshold value is used to determine when the human specialist is consulted. Then the specialist automatically gives control to the CBR

agent after performing a move in the game. Moreover, the retention of cases in the case base only happens when the human specialist is in control. Unlike passively acquired cases, which can result in the retention of redundant cases in the case base, the use of active learning in these games allowed the learning of certain situations that would not be observed in a purely passive manner. To achieve a reasonable expert imitation, active learning required a considerably lower number of cases than when a fully passive approach was used.

In contrast to these past works, this work actively learns only in the resolution of the required problems, which are identified as deception game opportunities. In addition to use a similarity threshold, the condition to query the human specialist is combined with a strategy that employs hand strength and probability to determine whether a situation is opportune for the use of deceptive moves. Instead of using active learning to collect any kind of expert experience of game playing, this work directs such learning to the improvement of the deceptive capabilities of card playing agents.

The AI research has also investigated the effectiveness of CBR in the modeling of card games, mainly with respect to the game of Poker [7, 23]. Considering deceptive-related Poker strategies, however, only [7] explicitly addresses this issue. There, the developed agent, whose case base starts empty, performs the random play strategy to populate the case base. With respect to the game of Truco, [10] addresses the case retention problem, especially considering the lack of large numbers of cases. It investigates alternative learning techniques such as ACE, Imitation Learning, and Active Learning to enable an agent to learn how to act in situations in which past case knowledge is limited. Through the assistance of a human player, the purpose of the active learning technique is to guide the agent in its use of any kind of game action whenever the agent had not encountered similar game situations stored in the case base. Despite this research, Truco matches disputed amongst the agents implemented according to the analyzed learning techniques showed that, unlike the automatic retention and the retention of new cases strategies, which demonstrated an improvement in the agents' performance, the active learning technique did not show an improvement in the agents' performance. Unlike [10], which performed a broad collection of case situations in Truco, this paper investigates the use of active learning in the analysis of deceptive game opportunities and, for those in which the expert decided that it was worth acting deceptively, the collection of new problem-solving experiences.

[8, 9] address the indexing of the Truco case base through the organization of cases into different clusters. Using such clusters, the goal was to identify game actions along with game states in which such actions are performed. In addition, it is proposed a two-step solution reuse model, which is further explored in our work. The model involves a step that retrieves the most similar cases for a given query, where a reuse criterion is used in the choice of the group of cases that is more similar to the current query situation (extra cluster reuse criterion). After selecting this group of cases, a filtering is performed in order to select only the retrieved cases that belong to the chosen group. Based on these filtered cases, a second reuse step can use another reuse criterion to choose the game action that is used to solve the current problem (intra cluster reuse criterion). The reuse policies that showed to be the most effective according to their experiments are described in Table 1.

Table 1. Reuse policies used by the implemented Truco agents.

Reuse policy	Criterion for the choice of clusters	Criterion for the choice of game actions	Reuse model
Number Points Solution (NPS)	-	Number Points	Standard reuse model
Probability Victory with Cluster and Number Points Solution (PVCNPS)	Probability Victory	Number Points	Two-step reuse model
Probability Victory Solution (PVS)	-	Probability Victory	Standard reuse model
Probability Victory with Cluster and Solution (PVCS)	Probability Victory	Probability Victory	Two-step reuse model

Considering the cases retrieved from a given query, the number of points solution criterion (NPS) involves the reuse of game actions, where the game action choice is supported on the amount of earned points due to the use of that action in the game. The probability victory (PV) criterion involves the choice of either clusters (PVC) or game actions (PVS) to be reused (or both in the PVCS), where the reuse is based on the calculation of the chances of victory for each of the different game actions recorded in the retrieved cases. These policies were thoughtfully explored in the development of the Truco playing agents investigated in this paper.

2.2 The Card Game of Truco

Truco is a widely practiced card game in Southern regions of South America [11]. The AI techniques covered in this work were investigated with the use of matches disputed between two opposing players. Such blind Truco version (Truco “Cego”) uses 40 of the 48 cards in the Spanish deck, as the four eights and four nines are removed. The deck is divided into “Black” cards, which are the cards with figures (King – 12, Horse – 11 and Sota – 10), and “White” cards that are from ace to seven.

In Truco, the dispute takes place through successive hands that are initially worth a point. Each player receives three cards to play one hand. A hand can be divided into two phases of dispute: ENVIDO and TRUCO. In each stage, players have different ways to increase the number of points that are played. Each hand can be played in a best of three rounds, in which the player who plays the highest card in each round wins. Finally, the match comes to an end when a player reaches twenty-four points.

ENVIDO is a dispute that takes place during the first round of a hand. Such a dispute is based on the sum of the value of each one of the player’s cards. For ENVIDO, each card is worth the value presented in it, with the exception of “black” cards that are not computed in the sum of points. ENVIDO has the following betting modalities: ENVIDO, REAL_ENVIDO, and FALTA_ENVIDO, which the player can bet before playing the first card on the table. If a player advances any one of these bets, the opponent can accept or deny the ENVIDO dispute. There is a special case of ENVIDO, which is called FLOR. The FLOR occurs when a player has three cards with the same suit. The FLOR bet cancels any ENVIDO modality previously advanced since it in-

creases the value of the ENVIDO dispute. As in ENVIDO, FLOR allows one to fight back (e.g. CONTRA_FLOR) if the opponent also has three cards of the same suit.

When the ENVIDO dispute ends, the TRUCO phase begins. At this stage, one to four points are disputed during the three rounds of the hand (one for each card in the hand). In one round, each player drops a card at the table starting with the hand player's or the winner from the previous round. These cards are confronted according to a Truco ranking involving each card. The player who wins two of the three rounds wins the hand. Unlike bet actions for ENVIDO, which can only be placed during the first round of each hand, TRUCO bets can be placed at any time during a hand dispute. In addition, if a player decides to go to the deck, the opponent receives a score of points equivalent to the points in dispute in that TRUCO stage.

Similar to other card games, such as in the different variations of Poker, for example, the game of Truco involves different degrees of deception/bluffing. These strategies allow players to win hands and even matches in situations where they do not own strong cards for the ENVIDO and TRUCO disputes. Most importantly, human players in real-life Truco matches employ deceptive actions with certain frequency. Among other reasons, this behavior makes the game more fun, even if such bluffs don't necessarily result in better results in the game.

3 Active Learning and CBR in the Card Game of Truco

Agents can employ CBR to learn game strategies for playing Truco. In our work, whenever such agents take the game turn, they evaluate the current state of the game. To do it, a query containing the game state information is formed. Then, the K-NN algorithm is executed along with a similarity function that averages case attribute similarities to perform the retrieval of past cases from the case base. After retrieval, the selected cases are used to generate a game move which is played in the current game situation. The reuse is supported by a reuse policy which defines, among other criteria, the number of similar cases considered in the solution choice and the minimum similarity value (threshold, set to 98% in this work) so that the solutions represented in the retrieved cases are reused in the resolution of the current problem. At the end of such problem-solving procedure, the system can decide whether the derived problem-solving experience is worth retaining as a new case in the case base.

3.1 The Case Base Formation

A web-based system was developed to permit the collection of Truco cases, where these cases were the result of Truco matches played between two human opponents who had various levels of Truco experience. At the end of each disputed Truco hand, a new case (i.e. a hand of Truco) was stored in the case base. In our project, 147 matches were played among different players using this system. In total, 3,195 cases were collected and stored into a case base called BASELINE. To represent the cases, a set of attributes captured the main information and game actions employed in the Truco disputes. Table 2 summarizes these attributes.

Table 2. Attributes for representing a Truco case.

Attributes	Description
Case identification	Identify the hand and match to which the hand belongs.
Player who starts actions on each hand	Player 1 or 2. The player that starts the hand dispute is known as "hand player".
Received and played cards	Cards received and played during the 3 hand rounds by each player organized as: "high card, medium card and low card".
Scoreboard	Points from each player when starting and finishing the hand.
Points regarding the ENVIDO dispute	Point score for the ENVIDO dispute.
Played bets/actions	Bets and game actions that each player performed during the hand dispute.
Winners for ENVIDO, TRUCO and the hand rounds	Player 1 or 2 who won every hand dispute.
Amount of points earned/lost in each dispute	Amounts of points awarded to the winners of each dispute that occurred in the hand.
...	...

The played cards were recorded according to a numerical codification. The encoding uses a nonlinear numerical scale ranging from 1 to 52. Code 1 is assigned the cards with the lowest value (all 4's). Code 52 is assigned the highest value card, which is the ace of spades. Then it was explored both in the representation of cases and in the similarity evaluations. In effect, the codification is based on both the categories identified in [24] and the Truco knowledge from our research group participants. Each value in this encoding represents the relative strength of the Truco cards.

To collect deceptive game information to support the development of the active learning task (only used during the active case learning) through the course of each played Truco match, other set of attributes were added into the case representation model. These attributes are described in Table 3.

Table 3. Attributes regarding the deceptive actions made in the Truco match.

Attributes	Deception type	Description
Deceptive actions performed with success	Bet/Raise Slow playing Bet/Raise with complete game information	Number of deceptive moves that achieved the goal.
Deceptive actions performed without success		Number of deceptive moves that didn't have the expected effect.
Revealed deceptive actions		Due to the cards played by the agent, number of deceptive actions that could have been detected by the opponent.
Opponent's deceptive actions		Due to the cards played by the opponent, number of deceptive actions that were discovered by the agent.

With respect to the deceptive actions performed by Truco players, case attributes to represent the deception information were used to measure the similarity of the cur-

rent game situation in relation to the cases stored in a LEARNING case base. The purpose was to determine whether such case base had enough records of problem opportunities for using deceptive actions in order to solve the game problems encountered in the matches in which the active learning tasks were executed.

3.2 Game Actions and Deception

Truco has various kinds of game actions. To support the analysis of deceptive Truco behaviors, we classified as *aggressive* the Truco playing actions involving betting or raising an opponent's bet. Similarly, *passive* are the actions in which the player should decide either accepting or denying an opponent's bet. In addition, *aggressive* game actions can be labeled as either *honest* or *deceptive*. In *aggressive* moves, the player can most effectively employ deception. In *passive* moves, the player has the opportunity to detect the opponent's deception since the opponent is either betting or increasing a bet. In Table 4, we analyze such deceptive game actions in Truco.

Table 4. Possible types of deceptive game actions in Truco.

Game action	Condition	Goal	Deception characteristic
Bet/Raise on ENVIDO-type bets	Low probability of having a better hand than the opponent	To give the impression that the deceiver has a strong hand. Consequently, to induce the opponent not to accept the bet or raise the bet	Concealment/simulation; omit the reality and show the untruth
Bet/Raise on TRUCO-type bets			
Bet/Raise with perfect information on TRUCO-type bets	Deceivers are sure they cannot defeat the opponent's card in the last hand round		
Slow playing on ENVIDO-type bets	High probability of having a better hand than the opponent	To induce the opponent to bet so that the deceiver can raise the bet in order to maximizing the number of points been disputed	Concealment; hides/omits the truth
Slow playing on TRUCO-type bets			
Bet/Raise with perfect information on ENVIDO-type bets	Low probability of having more points than the opponent	To induces the opponent not to accept the bet	Concealment/simulation; omit the reality and show the untruth

3.3 Hand Strength

Truco is played with 40 of the 48 cards of the Spanish deck, where there are 9,880 possible hands. With this, it is possible to sort and classify each hand according to their strength for the ENVIDO and TRUCO disputes. The ENVIDO hand strength is directly based on the ENVIDO points. To calculate the strength of a TRUCO hand, the relative strength and importance of each card that forms the hand have to be considered. A method to calculate such hand strength can be derived from the analysis of two components: a) the strength of the two highest value hand cards and b) the

strength of the two lower value hand cards. This method considers the Truco rules since a hand dispute is played in a best of three rounds.

The two highest hand cards a player possesses (high and medium cards, see Table 2) are more important in the estimation of final hand strength. To have two high cards in a hand tends to increase the player's chances of winning in the best-of-3 competition. A low card among these higher hand cards has a high negative impact on the final hand strength. On the other hand, a low card between the two lowest hand cards (medium and low cards) should also have a negative impact on the final hand strength. However, this impact is not as severe in the calculation of the hand strength as it is the impact of owning a low card between the two highest ones.

The method explores the calculation of means between the hand card numerical encodings (i.e. the non-linear encoding from 1 to 52). The first calculates a harmonic mean (1) between the two highest hand cards (high and medium cards). When one value much lower than another is used in this type of harmonic mean calculation, the final result of the computed mean tends to be reduced toward the lowest value.

$$M_1 = \frac{2}{1/HighCard + 1/MediumCard} \quad (1)$$

The second uses the calculation of a weighted arithmetic mean (2) between the two lower hand cards (medium and low cards). In this case, the weight attributed for the highest card between these two lowest hand cards was set to double the weight of the lowest card. The use of a weighted arithmetic mean also allows expressing the impact of a low card on the hand strength. However, the weight of having a high card between the two lower ones should be greater than the weight of having a lower card among these two lower cards.

$$M_2 = \frac{(2 * MediumCard) + LowCard}{2 + 1} \quad (2)$$

To reach the final value of the hand strength, a weighted arithmetic mean (3) is calculated between the results obtained from the two mean values computed with (1) and (2).

$$M_3 = \frac{(2 * M_1) + M_2}{2 + 1} \quad (3)$$

According to numerical tests, it was not possible to identify hand situations in which our calculations of hand strength presented unsatisfactory results. Qualitatively, either higher or lower values than those obtained by the use of our method could be argued as relevant in some situations. In these situations, even without the use of our method, the strength of the considered hands is subject to debate, especially when we considered the Truco rules and the different ways of deceptively playing in this game.

3.4 Triggering the Expert Consultation

As part of the proposed active learning approach, two strategies to trigger a human specialist consultation are proposed in the work.

First, the coverage of a case base is used as a trigger to consult the expert. To do it, the similarity between the current game situation and the cases learned through active learning is computed when a query is emitted. The query is performed on the case base containing the newly retained cases: the LEARNING case base. Such query considers the case attributes that are relevant for each type of decision in the game. In addition, the case attributes referring to the previously taken deception decisions in the match (Table 3) are considered in the similarity computations. The 98% similarity threshold was used to determine whether the LEARNING case base had sufficient coverage to resolve the current query situation.

Second, the trigger for the expert consultation is also directed to the identification of a problem opportunity to play deceptively. To define whether a particular decision-making scenario is characterized as an opportunity for such bluffing, the number of possibilities of certain Truco events is computed. With this, for example, it is possible to determine the probability of an opponent having more ENVIDO points than the points of the agents ENVIDO, using the card that had already been played by the opponent and the position of the agent on the table. Moreover, in each moment of the hand competition and in each type of game move, whenever there is a probability of success lower than 50%, the game situation can be classified as an “opportunity for deception”. Similarly, when the probability of success is higher than 85%, the agent may also adopt a slow playing deceptive move. Such estimate of the winning odds is computed in each new decision state of the game. It is updated according to the information revealed throughout the hand dispute.

The following example shows how this probability calculation is performed. Given the following agent’s cards: (3♠, 12♦, 6♦), which are removed from the deck, it is possible to calculate that the opponent can have $C_{37,3} = 7,770$ possible hands. Then the strength of each possible opponent hand is compared with the strength of the agent’s hand in each of these 7,770 card combinations. To do so, our method for calculating the hand strength is used. The result is that the agent has a better hand than the opponent in 5,106 hands. By computing the probability, there is a 66% chance of having a better hand than the opponent’s hand.

4 Experiments and Results

The developed experiments aimed to evaluate the effectiveness of the proposed active learning and CBR approach in the collection and exploration of deception cases in the stochastic and imperfect information game of Truco. These experiments were organized as follows: a) case learning, covering the acquisition of cases through active learning, b) agent performance, referring to the analysis of agent victories with and without the use of the collected cases, and c) agent behavior, concerning the evaluation of the set of decisions taken by the agents with and without the use of the collected cases. The tested Truco playing agents were implemented according to the four different solution reuse policies listed in Table 1. Different case bases were used by them: a) the initially collected case base (BASELINE, storing 3,195 cases), collected from matches played amongst human players, b) the resulting case base later built in

this work (ACTIVE, storing 5,013 cases), which increased the BASELINE case base with the new cases collected through active learning. To analyze the different game playing strategies adopted by the agents, according to the reuse policies along with their respective case bases, a number of evaluation attributes was used (Table 5).

Table 5. Game attributes observed in the analysis of the implemented agents.

Game attributes	Description
Matches	Total number of disputed matches.
Wins	Total number of victorious matches by each agent.
Game actions	Total number of ENVIDO/TRUCO game actions (passive + aggressive) by each agent.
Aggressive game actions	Total number of ENVIDO/TRUCO aggressive game actions by each agent.
Bluffs	Total number of played bluffs by each agent.*
Successful bluffs	Total number of successful played bluffs by each agent.*

* Bluffs: ENVIDO/TRUCO game actions in which the hand has either less than 50% or more than 85% (slow playing) winning chances.

Using the attributes described in Table 5, the analyzed strategies were the following: i) *Honest-deceptive*: indicating the rate of deceptiveness, it expresses the relationship between the total number of deceptive game moves and the total number of game moves. The higher the value is, the more deceptive the agent behavior is; ii) *Successful bluff*: indicating the rate of bluff effectiveness, it corresponds to the relationship between the number of successful bluffs and the total number of bluffs; and iii) *Passive-aggressive*: indicating the rate of aggressiveness, it captures the relationship between the number of aggressive moves of the ENVIDO/TRUCO-type and the total number of ENVIDO/TRUCO-type game moves, including when the agent does not bet. The higher the value is, the more aggressive the agent behavior is.

4.1 The Active Learning Experiment

To build the case base via active learning, 148 Truco matches were played between the agents implemented according to the tested reuse policies. The reuse policies used by each agent were randomly chosen at the beginning of each match. Only one of the players in each match had the ability to consult the expert human player, where the expert player was the first author of this paper.

During the collection of such cases via active learning, the agents computed their decisions by using the BASELINE case base. In these learning matches, whenever the learning criterion related to the detection of deception problem opportunities was satisfied, the learning algorithm presented information about the game to the specialist. Then, the expert reviewed whether the reused game action provided an effective solution for the current problem. With that, the expert player decided whether to maintain the decision recommended by the automated reuse policy or to perform another game action, deceptive or not. The expert decision was stored as a new case in a separate case base containing situations and decisions (LEARNING case base). New

cases were stored in this case base only when the specialist performed an intervention by changing the game action suggested by the reuse policy. In total, 1,818 new cases were stored in the LEARNING case base. When the reviewed game actions were used by the agents, they won 79% of the disputes played during this learning experiment.

4.2 The Evaluation Experiments

Due to luck and randomness, the quality of the Truco cards received by each player is likely to have a large variation. To reduce this imbalance in the evaluation experiments, the dispute model described by the Annual Computer Poker Competition (ACPC) [25] was adopted. This model employs duplicate matches, in which the same set of hands is distributed in two sets of matches. In doing so, players reverse their positions at the table when playing the second match. Because all players receive the same set of cards, this dispute model allows a fair assessment of agents' ability.

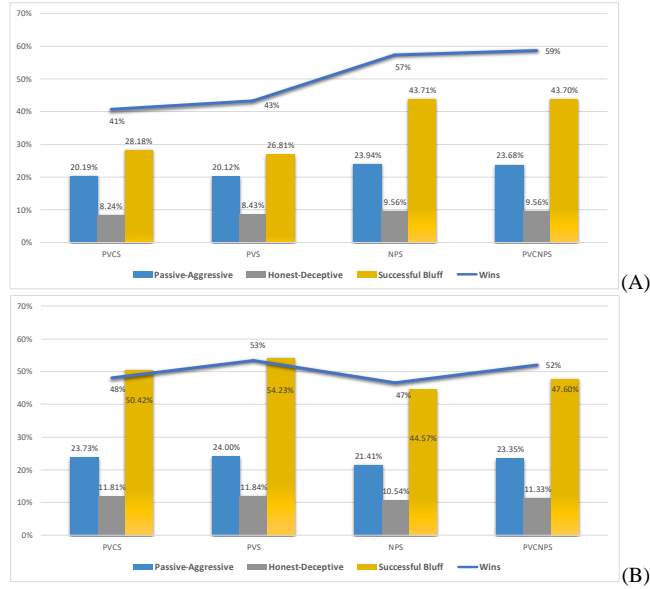


Fig. 1. Competition results between (A) agents implemented with the BASELINE case base only and (B) agents implemented with the ACTIVE case base only.

In the first set of tests, a competition between the four implemented agents was developed, where all of them competed against the others in a total of 300 Truco matches. The agents only used the BASELINE case base in all these matches. The results in Fig. 1 (A) indicate that the PVCNPS and NPS agents achieved the best performance. Regarding the analysis of their deceptive characteristics, even the BASELINE case base, which did not yet retained the cases collected via active learning, allowed these agents to deceptively play. In fact, that BASELINE case base col-

lected from human players already stored deceptive problem-solving experiences which were reused by the agents throughout these matches.

In the second set of tests, the test setup was similar to the previous one. However, the tested agents only used the ACTIVE case base. So the aim was to analyze whether the cases collected through the proposed active learning approach permitted to improve the agents' deceptive capabilities, and how such behavior change was expressed in the different kinds of tested agents. The results in Fig. 1 (B) indicate that the use of new cases collected via active learning enabled the PVS and PVCS agents to have the most significant performance improvement. While only the PVS and PVCS agents improved their aggressiveness rates, all tested agents increased their deceptiveness and successful bluff rates.

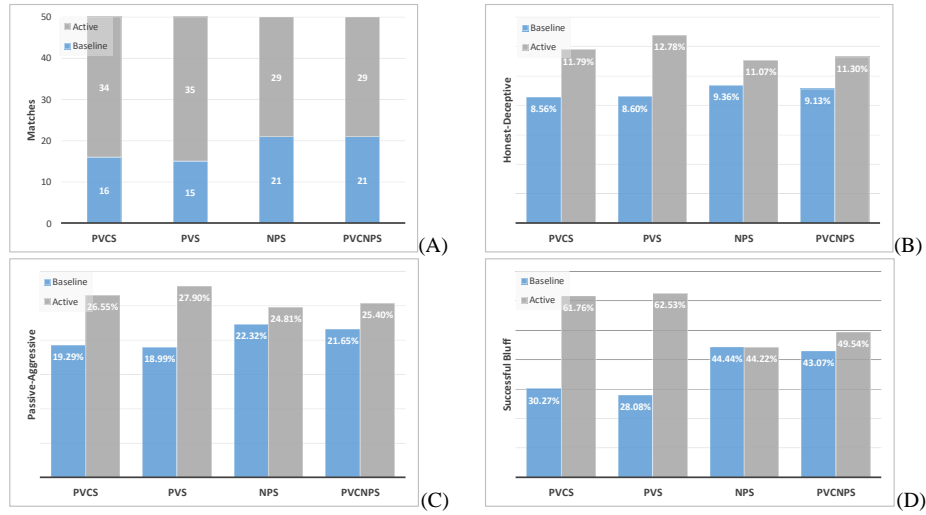


Fig. 2. Competition results between agents implemented with the BASELINE case base versus agents implemented with the ACTIVE case base.

In the third set of tests, each one of the four implemented agents was now configured to use different case bases: BASELINE and ACTIVE. In a total of 200 played matches, each kind of agent implemented with the use of one of these case bases played against its correspondent using the other case base. The results in Fig. 2 (A) indicate that the agents implemented with the use of the cases collected via active learning achieved a superior performance in relation to the others. The tests also permitted to observe the behavior of the implemented agents according to their reuse policies and the different case bases used to compute their game decisions. Fig. 2 (B) allowed analyzing the tested agents according to the honesty level, showing that the agents with the ACTIVE case base were more deceptive than the others. Fig. 2 (C) allowed comparing the agents according to their aggressiveness, showing that the ACTIVE case base enabled the tested agents to be more aggressive behaviors. Fig. 2 (D) allowed analyzing the assertiveness rate of performed bluffs by each one of the agents and case bases. Again, the results show that the agents with the ACTIVE case

base deceived better than their BASELINE’s correspondents (with a single exception: the NPS agent). In addition, the relationship between the agents’ performance and the adopted game behaviors is apparent since the reuse policies that obtained the best performance (PVS and PVCS) were those that played more aggressively, were more deceptive and performed a larger number of successful bluffs. Despite losing their matches, such behavior could also be observed with the better performing agents implemented with the BASELINE case base: the NPS and PVCNPS agents.

5 Final Remarks

This work investigates the integration of active learning and CBR, two different but complementary AI techniques, aiming to permit card playing agents to make better decisions when faced with problem opportunities to deceive. The experiments show that the actively learned cases allowed the tested agents to achieve a better game playing performance. Regarding the agents’ playing behaviors, the collected cases allowed them to more assertively act in deceptive problem situations. The CBR reuse policies that benefited the most, improving their deceptive behavior, were the ones that implemented the “Probability Victory” criterion (PVS and PVCS). As future studies, we can suggest the analysis of how deception could be related to other CBR techniques, e.g. in the execution of deceptive similarity computations. Further tests involving the implemented agents playing against human players are also relevant to improve the techniques proposed in this paper.

References

1. Buller, D.B., Burgoon, J.K.: Interpersonal Deception Theory. *Communication Theory* 6, 203-242 (1996)
2. DePaulo, B.M., Lindsay, J.J., Malone, B.E., Muhlenbruck, L., Charlton, K., Cooper, H.: Cues to deception. *Psychological Bulletin* 129, 74-118 (2003)
3. Ekman, P.: *Telling lies: Clues to deceit in the marketplace, politics, and marriage*. W. W. Norton & Company, Inc. (2009)
4. Billings, D., Davidson, A., Schaeffer, J., Szafron, D.: The challenge of Poker. *Artificial Intelligence Journal* 134, (2002)
5. De Mantaras, R.L., McSherry, D., Bridge, D., Leake, D., Smyth, B., Craw, S., Faltings, B., Maher, M.L., T COX, M., Forbus, K.: Retrieval, reuse, revision and retention in case-based reasoning. *The Knowledge Engineering Review* 20, 215-240 (2005)
6. Rubin, J., Watson, I.: Computer Poker: A review. *Artificial Intelligence* 175, 958-987 (2011)
7. Sandven, A., Tessem, B.: A case-based learner for Poker. *The Ninth Scandinavian Conference on Artificial Intelligence (SCAI 2006)*, Helsinki, Finland (2006)
8. Paulus, G.B., Assunção, J.V.C., Silva, L.A.L.: Cases and clusters in reuse policies for decision-making in card games. *IEEE 31st Int. Conf. on Tools with Artificial Intelligence (ICTAI 2019)*, pp. 1361-1365, Portland, OR, USA (2019)

9. Paulus, G.B.: Cases and clusters for the development of decision-making reuse policies in card games (written in Portuguese). Programa de Pós-Graduação em Ciência da Computação, vol. Master in Computer Science, pp. 132. Universidade Federal de Santa Maria (2020)
10. Moral, R.C.B., Paulus, G.B., Assunção, J.V.C., Silva, L.A.L.: Investigating case learning techniques for agents to play the card game of Truco. XIX Brazilian Symposium on Computer Games and Digital Entertainment (SBGames 2020), pp. 107-116, Recife, Brazil (2020)
11. Winne, L.L.: Truco. Ediciones Godot, Ciudad Autónoma de Buenos Aires (2017)
12. Settles, B.: Active Learning Literature Survey. Department of Computer Sciences, University of Wisconsin–Madison (2009)
13. Richter, M.M.: Knowledge containers. In: Watson, I. (ed.) Readings in Case-Based Reasoning. Morgan Kaufmann Publishers, San Francisco (2003)
14. Neto, H.C., Julia, R.M.S.: ACE-RL-Checkers: decision-making adaptability through integration of automatic case elicitation, reinforcement learning, and sequential pattern mining. Knowledge and Information Systems 1-32 (2018)
15. Floyd, M., Esfandiari, B.: Supplemental observation acquisition for learning by observation agents. Applied Intelligence 48, 4338-4354 (2018)
16. Ontanon, S., Floyd, M.: A comparison of case acquisition strategies for learning from observations of state-based experts. The 26th Int. Florida Artificial Intelligence Research Society Conf. (FLAIRS 2013), Florida, USA (2013)
17. Ross, S., Bagnell, D.: Efficient reductions for imitation learning. In: Yee Whye, T., Mike, T. (eds.) The Thirteenth Int. Conf. on Artificial Intelligence and Statistics, vol. 9, pp. 661-668. PMLR (2010)
18. Ross, S., Gordon, G., Bagnell, J.A.: A reduction of imitation learning and structured prediction to no-regret online learning. The 14th Int. Conf. on Artificial Intelligence and Statistics (AISTATS), pp. 627-635, Ft. Lauderdale, FL (2011)
19. Packard, B., Ontanon, S.: Policies for active learning from demonstration. 2017 AAAI Spring Symposium Series, Stanford University (2017)
20. Packard, B., Ontanon, S.: Learning behavior from limited demonstrations in the context of games. The 31st Int. Florida Artificial Intelligence Research Society Conf. (FLAIRS 2018), Florida, USA (2018)
21. Miranda, M., Sanchez-Ruiz, A., Peinado, F.: Towards human-like bots using online interactive case-based reasoning. In: Bach, K., Marling, C. (eds.) The 27th Int. Conf. on Case-Based Reasoning (ICCBR 2019), pp. 314-328. Springer International Publishing, Otzenhausen, Germany (2019)
22. Floyd, M.W., Esfandiari, B.: An active approach to automatic case generation. The 8th Int. Conf. on Case-Based Reasoning (ICCBR 2009), vol. 5650, pp. 150-164. Lecture Notes in Computer Science, Seattle, WA, USA (2009)
23. Rubin, J., Watson, I.: Case-based strategies in computer Poker. AI Communications 25, 19-48 (2012)
24. Sobrinho, M.G.: Manual do jogo do Truco Cego (Flor de Abóbora). Martins Livreiro Editora Ltda., Porto Alegre (2004)
25. ACPC: Annual Computer Poker Competition. <http://www.computerpokercompetition.org/> (2018)