Proceedings of the 2007 IEEE Symposium on
Computational Intelligence and Games (CIG 2007)

Inferring the Past: A Computational Exploration of the Strategies that May Have Been Used in the Aztec Board Game of Patolli

Andrés Gómez de Silva Garza
Instituto Tecnológico Autónomo de México
(ITAM—Mexican Autonomous Technology Institute), Río Hondo #1
Colonia Tizapán-San Ángel
01000—México, D.F., Mexico
agomez@itam.mx

Carlos Emilio Galindo Flores
Instituto Tecnológico Autónomo de México
(ITAM—Mexican Autonomous Technology Institute), Río Hondo #1
Colonia Tizapán-San Ángel
01000—México, D.F., Mexico
cgalindo@itam.mx

***Abstract*—In this paper we use computational techniques to explore the Aztec board game of Patolli. Rules for the game were documented by the Spanish explorers that ultimately destroyed the Aztec civilization, yet there is no guarantee that the few players of Patolli that still exist follow the same strategies as the Aztec originators of the game. We implemented the rules of the game in an agent-based system and designed a series of experiments to pit game-playing agents using different strategies against each other to try to infer what makes a good strategy (and therefore what kind of information would have been taken into account by expert Aztec players back in the days when Patolli was an extremely popular game). In this paper we describe the game, explain our implementation, and present our experimental setup, results and conclusions.**

Keywords: Board Games, Intelligent Agents, Evaluation of Alternate Strategies

I. Introduction

In this paper we use computational techniques to explore the Aztec board game of Patolli. This game shares some features with the European game of Goose and the Indian game of Parcheesi, yet developed independently. To the Aztecs, at least to the upper classes who had a surplus of time and possessions, playing and betting on the game of Patolli was a social occasion of extreme importance, as it was a way of communing with two of their gods: Macuiltxóchitl, god of dance, sports and games, and Ometochtli, god of pulque (an alcoholic drink made from the agave cactus) [1], [2].

Some traits of the game were documented by some of the earliest Spanish explorers before they ultimately destroyed the Aztec civilization, including the formerly widespread habit of playing Patolli [3], [4]. Some historians have suggested that the game has been lost forever [5]. Although there are almost no written descriptions of the game which have survived from Aztec times, it has been found that in some Mexican communities the game is still played [2]. The game's existence and rules have probably been passed on through the centuries via the oral transmission of traditions. However, knowing the rules of a game and knowing how to play it are two different things. Modern Patolli tournaments exist [6], though the number of people that play the game is not very high. These modern players may have developed their own expertise through long series of games, or may have been taught how to play by ancestors who have passed on their knowledge of the game (which they probably received in the same fashion). However, in either case we cannot be sure that the way they play reflects the way that the game was played 500 years ago, before the European conquest of the Americas.

We decided to implement the rules of the game of Patolli in an agent-based computer system in which we gave agents the possibility to try out alternate strategies. Each strategy differs from the others depending on the type of information it takes into account in order to decide what moves to make. We then designed a series of experiments to pit game-playing agents using different strategies against each other. The purpose of evaluating multiple alternative strategies is to try to infer, based on the results of the experiments, what kind of information was probably taken into account by expert Aztec players back in the days when Patolli was an extremely popular game.

The advantage of this multi-agent approach is that it is easy to control which aspects of agents are different (in this case, the strategies they employ to make decisions) and which are identical (in this case, everything else: the variables that describe the agents, the game-state information they have access to, etc.). It therefore becomes easy to isolate one component (the one for which we are interested in evaluating variations), leaving everything else the same (and thus not allowing anything else to influence the outcome of the evaluations). A similar multi-agent approach was followed by [7] in order to play the game Trouble®, yet that paper focuses on team-work in game playing (i.e., communication between agents during a game), whereas in our approach each agent is an opponent of the other agent(s) and therefore agents don't communicate with each other. Another discussion of intelligent agents in computer games is provided in [8].

In Section II of this paper we describe the game of Patolli and its rules. In Section III we present our system and how it implements the different rules of the game. In Section IV we describe a series of experiments we performed in order to evaluate the different game-playing strategies we implemented. Finally, in Section V we present our experimental results and draw some conclusions about the reasoning processes that expert Aztec Patolli players may have undergone when deciding what moves to make.

II. The Game of Patolli

The four cardinal directions and the center, or up-down direction, were among the most important concepts for Aztecs. Gods were associated with specific directions [2]. The agricultural calendar year of the Aztecs had four special dates, two equinoxes and two solstices, which separated the year into four seasons. There was also a fifth "season," consisting of a five-day holiday to celebrate the end of one year and beginning of the next. Due to their special significance, the numbers 4 and 5 appeared often in daily and religious life, including in the game of Patolli.

The rules for Patolli which we have programmed, described below, were put together by us based on partial descriptions found in different anthropological and historical studies [1]-[5] and modern-day websites produced by Patolli enthusiasts [6], [9]-[12]. Some of these sources contradict each other in certain details (perhaps because the Aztecs themselves did not have just one standardized way of playing the game or drawing the Patolli board layout), but we have tried to come up with a coherent set of rules and design of the board for the game that seems to us to be the most logical or most likely to be correct, based on all these sources.

A match in Patolli consists of a series of games played between two, three, or four players (or teams of players), each of which begins by placing a bet (presumably precious stones or other items valued by the Aztecs, but we will refer to this in general as "making a bid" with "goods" or "possessions"). A match ends when one player has won all of the possessions of the rest of the players (who drop out of a match as they become bankrupt until there is only one overall winner of the match). This may occur in the middle of a game or at the end of a game (*i.e.*, games can be interrupted before reaching a "natural" end if all but one player has been bankrupt in the middle of a game—how this can happen is explained below).

Games are played on a board which was originally drawn on a mat using a paint made with tar. The board is in the form of a large letter X subdivided into 52 board positions (boxes). The number 52 also had a special meaning to the Aztecs—it is the number of years in one of their centuries [2]. Board positions are classified into five types: there are four start positions (s—each one corresponding to a different player), four end positions (e—each one corresponding to a different player), eight extra-turn positions (d), eight toll or pay positions (p), and the rest are "normal" positions (n).

Each player has five beads or tokens of a certain color which have to be advanced from outside the board, onto that player's start position, and through the rest of the board to his/her end position. Four bright, red-colored beans (a species found in Mexico), each marked with a dot on one side, are tossed by a player and used as dice at each turn during a game to determine how many board positions that player may advance one of his/her tokens. The number of beans that can have a face-up dot after a toss is therefore 0, 1, 2, 3, or 4. The probability of each of these outcomes, however, might be different from that of the rest, so it is like tossing loaded or biased dice. In addition, if one, two, three, or four face-up dots are visible on the beans after a toss, the player whose turn it is can move one of his/her tokens by that number of board positions; however, when no beans have a face-up dot, and due to the special nature of this occurrence, a player is allowed to move one of his/her tokens by ten board positions. Table I shows the probabilities and number of board positions that are advanced for each of the possible outcomes of a bean-toss.

TABLE I
Analysis of the Four Beans (Used as Dice)

Total number of face-up dots	Probability	Number of board positions to advance
none	1/16	10
1	4/16	1
2	6/16	2
3	4/16	3
4	1/16	4

One restriction is that a token may only enter the board (at the corresponding player's start position) if a 1 results from a bean toss. However, once several tokens have entered the board, a player has to decide which of his/her tokens to move, given the number of board positions that he/she can advance (based on the result of the bean toss). In order to make this decision, different strategies can be thought of, but the lack of human Patolli experts that can be consulted (in order to learn the heuristics they use when playing) means that determining which strategies are better than others can best be done through a computational experiment.

Fig. 1 shows the Patolli game board and beans in a typical configuration. The p-type boxes are those shown under the boldface X's drawn near the middle of each of the "arms" of the board (shown this way because that is how the Aztecs apparently drew them).

Some additional rules with respect to the game board are the following. If a player's token falls on an s- or e-type box and the box is occupied by an opponent's token, then the opponent's token is returned to its start state (*i.e.*, it must start at the opponent's start position, but can only do so when a bean-toss results in a 1) and the opponent has to pay the player a certain number of goods. This is the only situation in which a token is allowed to land on a board position that is occupied (momentarily) by another token. If a player's token falls on that player's end-position, the token is taken out of play and the opponents have to pay the player a certain number of goods. A player's token cannot "overtake" that player's end position (*i.e.*, is not allowed to go around the board more than once). If a player's token falls on a d-type box, that player gets an extra turn (and is allowed to move any of his/her available tokens in the new move). If a player's token falls on a p-type box, the player must pay his/her opponents a certain number of goods. Finally, if a player's token falls on an n-type box, nothing special happens. A game ends "naturally" when all of one player's tokens have reached that player's end position. As can be seen, there are three types of situation in which payments may occur between players. Any of these may leave a player bankrupt (and not all occur before a game has ended in a "natural" way).

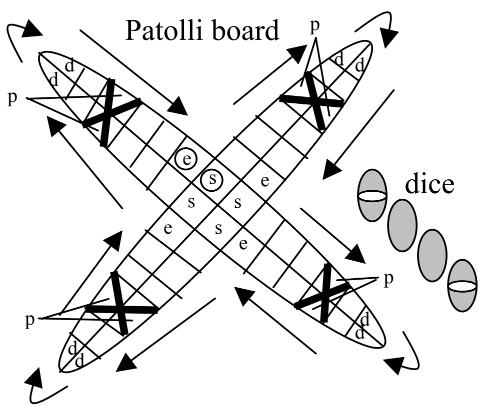

Fig. 1. The 52 boxes of the Patolli board game and four beans, each marked on one side, used as dice. Arrows show the direction in which tokens move during play. The start and end boxes shown circled are those that "belong to" one of the players.

Table II gives a summary of the different types of board positions, their semantics, and how many of them occur on a Patolli board.

III. Implementation

We have implemented the rules of Patolli in a system called GATIGO (the GAme of TIme and GOds). For the moment our implementation assumes a two-player game, though it can easily be extended in order to perform three- or four-player games. In GATIGO we start each player off with 20 goods in their possession, and each time a player has to make a payment it is just one possession which is transferred (irrespective of which of the three possible situations occurred that cause a payment to be required). These parameters can also be easily adjusted.

The GATIGO system has an agent-based architecture. In our implementation, beans are agents which, when asked to "flip," produce a Boolean value (representing either the fact that the dot-side ended up face-up, or that it ended up face-down). Players are agents which have to decide which move to make. In order to do so, they have to analyze each of their five tokens (or the subset of them which is currently on the board or able to enter the board) to decide which one to move. Only one of the tokens can be moved by the player, and the move has to be by a number of board positions which depends on the number of beans that ended up face-up after flipping them (all four bean agents are flipped in each turn). Tokens are agents that, when prompted by their player owner-agent, return a value indicating to what degree that token thinks it should move at that point in the game (by the number of board positions indicated by the beans). Each player agent therefore owns and consults five token agents.

TABLE II
Classification of Board Positions

Vulnerability	Type	Description	Special features	No. of boxes of this type on board
If occupied by a player's token, the token can be bounced back (returned to its start state, outside of the board). This occurs if an opposing player's token landing on the same board position. The player who is bounced back has to pay some of its possessions to the player that bounced him/her back. This applies to both s- and e-type boxes.	s	start	A token can only enter the board at its corresponding start box when one face-up dot appears after tossing the beans.	4
	e	end	A token must land on its corresponding end box by advancing an exact number of positions. When this occurs, the token's owner receives some goods as payment from each of the other players.	4
If already occupied by any token, the board box cannot be landed on. This applies to d-, p-, and n-type boxes.	d	extra-turn	A player who lands a token on this type of position can toss the beans again (and make a new move based on the results).	8
	p	toll or pay	A player whose token lands on one of these positions has to pay some goods to each of the other players.	8
	n	normal	None	28

There are thus two types of agent in GATIGO that have to make decisions that can influence the outcome of a game, the tokens and the players. Tokens have to decide how good or bad it would be for them to move during their owner's current turn in the game. We represent this as a value between 0 and 1, and call it the token's *move weight*. The calculation of the move weight of a given token is independent of the calculation performed at that time by any of the other tokens belonging to the same player. The research question here is what type of information is useful/necessary to take into account in order to make the calculation? Below we describe some strategies that we have implemented for calculating a token's move weight which differ in the type of information that the use to make the calculation.

Players have to decide which of their tokens to move by taking into account the feedback provided by all of their tokens. If one token returns a move weight that is clearly better than the move weights of the rest of the player's tokens, the decision is easy. However, if several of the tokens consulted by the player have returned an equally or similarly good value for their move weight, some sort of tie-breaking criterion has to be taken into account. The research question here is what kind of criterion works best for deciding amongst several equally-good tokens in the game of Patolli? Below, after our discussion of our move-weight calculation strategies, we describe some tie-breaker criteria that we have implemented for a player to decide which of its tokens should be moved in its current turn. In the discussion below we will use b_t to refer to the board position occupied by a token at time t (the current time). Therefore b_{t+1} represents the next board position that a token will occupy, and b_{t+2} represents the following position after that.

A. Strategies for Calculating a Token's Move Weight

A player in GATIGO is programmed to follow a given strategy, where the strategy determines which kind of information the player's tokens use to calculate their move weight. The strategy can be adjusted from one match to the next, but all of a player's tokens follow the same strategy within one match. The strategies we have implemented for a token to calculate its move weight are the following:

S0: this strategy automatically assigns a 1 as the move weight. The purpose of this strategy is that, if we set up the tokens of several players to follow S0, then we can ask one player to follow a different tie-breaker criterion from that followed by another player, and thus compare tie-breaker criteria to each other by themselves (independently of, and the results of the comparison thus not being influenced by, the strategy used by the players for calculating the move weight).

S1: this strategy assigns the value of the move weight depending on the type of board position on which a token will land if it is moved (*i.e.*, depending on the type of b_{t+1}). If the token will fall on an extra-turn board position, a high value (0.9) is assigned to the move weight, because getting an extra turn in the game is deemed to be a good thing. If the token will fall on a payment-must-be-made board position, a low value (0.1) is assigned to the move weight, because having to pay one's opponent is not a desirable thing in the game of Patolli. If the token will fall on its player-owner's end position, then a high value (1) is assigned to the move weight, because one doesn't always get the correct number of dots after flipping the beans in order to be able to end a token's advance through the board, so when one has the opportunity to end the advance, it must be taken advantage of. If the token will fall on a different end position or a start position and said position is occupied by one of the opponent's tokens, then a high value (1) is assigned to the move weight, because this type of game situation forces the opponent to have to pay the token's owner. If the token will fall on the same kind of positions but they are not occupied, then a medium-low value (0.4) is assigned to the move weight, because of the vulnerability fo that type of board position—there is a slight risk of the opponent then falling on the token being evaluated, thereby forcing the token's owner to pay. All other positions that a token may fall on imply that a medium-high value (0.6) is assigned to the move weight, because the decision of moving the token or not is neutral with respect to the type of board position.

S2: this strategy assigns the value of the move weight depending on the type of board position that the token is currently occupying (*i.e.*, depending on the type of b_t). If the token is currently in a start or end position, the move weight is assigned a high value (1) because of the risk inherent in occupying one of these types of position (if the opponent falls on one of "our" tokens—from the point of view of the player applying this reasoning strategy—in one of these types of position, the opponent must be paid). If the token is currently in any other type of board position, a medium value (0.5) is assigned to the move weight. This is due to the fact that other types of board positions are neutral from the point of view of deciding whether moving out of them makes sense or not. It is irrelevant to a player if one of his/her/its tokens moves away from such a position or not from the point of view of losing turns in the game or losing goods to other players (*i.e.*, the negative things that can occur to a player during a game).

S3: this strategy assigns the value of the move weight depending on an analysis of the possible board positions in which the token can land, and the probabilities of landing on them, in the player's following turn, assuming the token moves at the current time. In other words, it is like S1, but looks ahead an extra step, and assigns the move weight based on that extra information. The value assigned to the move weight depends here on the type of b_{t+2}, independently of the type of b_{t+1}.

The exact numeric values given to the board weights in the different strategies are arbitrary, but qualitatively we believe that they make sense, as explained in the description of each strategy above. Each strategy is different from the rest in that it uses a different type of information to make its calculation of the move weight, thus allowing us to evaluate which types of information seem to be better than the other types by pitting the different strategies against each other. Additional strategies can be implemented that take into account more types of information not mentioned above, such as the local neighborhood (within the game board) of a token, including which other tokens are nearby (both those belonging to the same player and to the opposing player, thus introducing the concept of blocking or unblocking the paths of certain tokens as they move around the board) or looking forward more steps into the (possible) future states of the game. We have left the implementation of these additional strategies, and the subsequent experiments that would need to be performed to evaluate them, for a future version of GATIGO, as we wanted to concentrate on the simplest possible strategies to begin with.

B. Tie-Breaker Criteria

If several of a player's tokens seem to be "good" candidates to be moved in the current turn, a criterion that is guaranteed to result in just one candidate token, independent of the strategy used to assign move weights, must be taken into account. For instance if S1 is being used and two of the player's tokens can land on an extra-turn position, given the current point-value of the beans, then somehow a decision has to be made about which of the two tokens to move. The tie-breaker criteria we have implemented are the following:

T0: this criterion chooses randomly from amongst the tied tokens in order to decide which one should move.

T1: this criterion chooses the most advanced of the tokens that are tied.

T2: this criterion chooses the token that is furthest behind, from amongst those that are tied.

These tie-breaker criteria seem very simple, but choosing one over the other can have a large influence on the overall behavior of a player's tokens. In addition, one would expect T1 to tend to favor the dispersal of a player's tokens (because, all other things being equal, the most advanced token will pull even further away from the rest of the player's tokens than it already is). T2 would be expected to tend to favor a player's tokens bunching together (because, all other things being equal, the one that is furthest behind will advance and catch up with or join the group of the rest of the tokens on the board). T0, on the other hand, due to its random nature, presumably represents a balance between dispersing and bunching up. However, the measurements we performed during our experiments do not support this overly-simplistic explanation of the characteristics of the different tie-breaker criteria, as can be seen in Section V.

IV. Experimental Setup

We initially ran the system with both player-agents (from now on, P0 and P1) programmed to follow the same strategy and the same tie-breaker criterion for ten sets of 100, 500, 1000, 5000, and 10000 matches each. As expected, both players ended up winning approximately the same amount of times in each of the sets of matches. However, the variance in the results between one set of matches and the next was greater when the number of matches in a set was less than 5000. At 5000 we had found "stability" (more robust or trustworthy results—results not biased by the random nature of the bean tossing and other factors).

After making this observation we therefore decided to perform one set of 5000 matches for each of the experiments we performed. The experiments were designed to test the performance of the different strategies for assigning move weights (S0, S1, S2, and S3) combined with the different tie-breaker criteria (T0, T1, and T2) we have implemented. Each different S-T pair was tested against each other possible S-T combination for 5000 matches in order to measure its performance.

Performance can be measured in several ways (for example, in the context of GATIGO, how successful one is at bankrupting one's opponents, or how efficient an S-T pair is at getting through a game). Because of this we decided to measure the following variables in the experiments: the total number of matches won per player, the total number of games won per player, the total number of times each player's tokens were forced to return to the beginning (associated with which is a payment to the opposing player), the total number of times each player landed on a p-type board position (and therefore had to pay its opponent), the total number of times each player won an extra turn, the total number of moves each player had to make, and the average number of tokens a player had on the board at any given time during a game.

V. Results and Conclusions

In the results we present here we highlight those that produced a clear difference in performance (at least a ratio of 3:1 in the values of the variables measured when comparing P0 and P1, except where the variation between combinations was too small, in which case a smaller ratio was highlighted). In the tables shown in the next few figures, columns correspond to P0 and rows to P1. Each column and each row represents a different S-T combination for the corresponding player, and their intersections show whether the S-T pair represented by the column (P0) or by the row (P1) "won" (given the variable that was measured in order to produce the corresponding table) in a face-to-face competition. Boxes that are shaded (blue) correspond to cases in which P0 clearly won by at least 3:1 (the winning S-T pair can be consulted in the corresponding column heading) and boxes that are very lightly shaded (yellow) correspond to cases in which P1 clearly won (the winning S-T pair can be consulted in the corresponding row heading). Columns and rows in each table are ordered according to the number of cases in which the corresponding S-T pair prevailed (against the other S-T pairs tested). Therefore the order in which the various S-T combinations appear in the rows and columns may differ from one table to the next.

Fig. 2 shows the results of comparing each possible S-T pair to all others when measuring the number of matches won. This measurement represents an S-T pair's overall success in the game of Patolli, as it is equal to the number of times the opposing player was bankrupted.

It can be observed from Fig. 2 that there are three groups of S-T combinations: three very successful ones (which are most likely to reflect the way that the Aztecs played the game), seven mediocre ones, and two really awful ones. The S3T1 pair (assigning the move weights based on b_{t+2}, and moving the most advanced token as the tie-breaker criterion) was one of the dominant S-T combinations. The other two S-T pairs in the group of successful ones both involve S1 (assigning move weights based on the destination board position b_{t+1}), but strangely enough they used T0 (choosing at random) and T2 (advancing the least advanced token), respectively, as their tie-breaker criterion. The fact that none of these was T1 (even though S3 combined with T1 was the other successful combination) means that the interactions taking place between strategies used to assign move weights and tie-breaker criteria are both interesting and complex. Even more strange is the fact that S1T1 is the worst combination overall, when one would expect that all variations of S1 would be relatively successful if the other two pairs involving S1, S1T0 and S1T2, are (and if the other successful combination, S3T1, involves T1)!

	S3T1	S1T0	S1T2	S2T1	S0T2	S0T0	S2T0	S2T2	S3T0	S3T2	S0T1	S1T1
S3T1	1	1	1	713	4999	5000	5000	5000	2499	4999	5000	5000
S1T0	1	1	1	624	1666	999	4999	4999	2499	1666	5000	5000
S1T2	1	1	1	832	1666	999	2499	1666	2499	2499	4999	5000
S2T1	713	624	832	1	2	2	3	2	2	2	12	160
S0T2	4999	1666	1666	2	1	1	1	1	2	2	13	4999
S0T0	5000	999	999	2	1	1	2	1	2	2	13	713
S2T0	5000	4999	2499	3	1	2	1	1	1	1	10	1249
S2T2	5000	4999	1666	2	1	1	1	1	1	1	9	1666
S3T0	2499	2499	2499	2	2	2	1	1	1	1	5	713
S3T2	4999	1666	2499	2	2	2	1	1	1	1	5	999
S0T1	5000	5000	4999	12	13	13	10	9	5	5	1	293
S1T1	5000	5000	5000	160	4999	713	1249	1666	713	999	293	1

Fig. 2. Comparison of the success of the different S-T combinations against all others when measuring the number of matches won.

One possible explanation for these observations is the fact that S3T1 favors the participation of several tokens in the game because the first position in which newly-entered tokens will fall, a vulnerable s-type box, is ignored by S3, which looks one additional step ahead. In contrast, S1 notices that the initial board position which a token must occupy is a vulnerable one, and therefore tends to disfavor the entrance of a token into the board. When this happens, the most advanced token is forced to move, even if it means falling onto a p-type fox, thereby causing the player to lose its goods and, after this occurring repeatedly, the player eventually loses the match. This happens when the tie-breaker criterion is T0 or T2, and is only exacerbated if it is T1 (moving the most advanced token), which is why S1T1 was the worst combination of all.

Fig. 3 shows the results of comparing each possible S-T pair to all others when measuring the number of games won.

As can be seen from Fig. 3, the results are only slightly different from, and in general consistent with, those shown in Fig. 2 (in which the number of matches won was the variable that was measured). There is no *a priori* reason to assume that a good overall S-T combination (one that bankrupts the opposing player, and therefore wins matches, often) results from winning a larger number of games than other S-T pairs. This is due to the fact that the way in which the majority of games were played (trying to minimize one's payments and maximize the opponent's) would seem to be more important than the actual number of games won. However, in general the two variables (matches won and games won) do appear to be related to each other in Patolli, as shown in Fig. 3.

	S3T1	S1T0	S1T2	S3T0	S3T2	S2T1	S0T0	S0T1	S0T2	S2T0	S2T2	S1T1
S3T1	1	5	5	177	231	111	416	505	330	424	350	5000
S1T0	5	1	1	10	9	11	16	28	16	17	18	455
S1T2	5	1	1	9	10	10	14	31	16	17	16	500
S3T0	177	10	9	1	1	1	2	2	2	2	2	91
S3T2	231	9	10	1	1	1	2	2	2	2	2	122
S2T1	111	11	10	1	1	1	2	2	2	2	2	24
S0T0	416	16	14	2	2	2	1	1	1	1	1	9
S0T1	505	28	31	2	2	2	1	1	1	1	1	83
S0T2	330	16	16	2	2	2	1	1	1	1	1	8
S2T0	424	17	17	2	2	2	1	1	1	1	1	10
S2T2	350	18	16	2	2	2	1	1	1	1	1	11
S1T1	5000	455	500	91	122	24	9	83	8	10	11	1

Fig. 3. Comparison of the success of the different S-T combinations against all others when measuring the number of games won.

Fig. 4 shows the results of comparing each possible S-T pair to all others when measuring the amount of times each player's tokens were forced to return to the beginning (which we shall call degree of bounce-back).

	S3T1	S1T1	S0T1	S0T0	S0T2	S2T0	S2T2	S1T0	S1T2	S3T0	S3T2	S2T1
S3T1	1	1.44	1.32	1.57	1.58	1.4	1.41	3.12	3.12	3.35	3.4	3.6
S1T1	1.44	1	1.43	1.62	1.64	1.51	1.5	2.52	2.51	3.62	3.64	3.61
S0T1	1.32	1.43	1	1.41	1.43	1.27	1.26	2.42	2.4	2.77	2.78	3.16
S0T0	1.57	1.62	1.41	1	1.01	1.14	1.12	2.11	2.1	2.14	2.19	2.63
S0T2	1.58	1.64	1.43	1.01	1.01	1.14	1.13	2.08	2.11	2.14	2.14	2.61
S2T0	1.4	1.51	1.27	1.14	1.14	1	1	2.38	2.4	2.47	2.49	2.96
S2T2	1.41	1.5	1.26	1.12	1.13	1	1	2.42	2.43	2.45	2.46	2.97
S1T0	3.12	2.52	2.42	2.11	2.08	2.38	2.42	1	1.01	1.28	1.22	1.63
S1T2	3.12	2.51	2.4	2.1	2.11	2.4	2.43	1.01	1.01	1.24	1.24	1.6
S3T0	3.35	3.62	2.77	2.14	2.14	2.47	2.45	1.28	1.24	1	1	1.38
S3T2	3.4	3.64	2.78	2.19	2.14	2.49	2.46	1.22	1.24	1	1	1.37
S2T1	3.6	3.61	3.16	2.63	2.61	2.96	2.97	1.63	1.6	1.38	1.37	1.01

Fig. 4. Comparison of the success of the different S-T combinations against all others when measuring the number of times each player's tokens were forced by the opponent to return to the beginning (degree of bounce-back).

According to Fig. 2, S3T1 is one of the best S-T combinations, yet Fig. 4 shows that it involves the highest degree of bounce-back. Intuitively it would seem that this contradicts the fact that S3T1 is a winning combination. This is due to the fact that each bounce-back involves paying the opposing player (the one that forced the bounce-back by landing on a player's token as it occupied an s- or e-type board position). Having to pay one's opponent frequently would not seem to be an effective way of bankrupting the opponent! Fig. 5 and Fig. 6 make things clearer with respect to this apparent contradiction.

Fig. 5 shows the results of comparing each possible S-T pair to all others when measuring the total number of times each player landed on a p-type board position and therefore had to pay the opposing player.

In Fig. 5 we can see that S3T1 is the "best" S-T combination when measuring payments made (it involves landing on a p-type box the least number of times). It turns out that this more than compensates for S3T1's poor showing in Fig. 4 (in which the variable measured was degree of bounce-back). This can be best seen in Fig. 6, which gives the result of combining (adding) the results from Fig. 4 and Fig. 5.

	S0T0	S0T1	S0T2	S1T1	S2T0	S2T1	S2T2	S3T0	S3T2	S1T0	S1T2	S3T1
S0T0	1	1.02	1	1.26	1.01	1.08	1.01	1.11	1.11	1.92	1.89	4.03
S0T1	1.02	1	1.03	1.24	1.03	1.03	1.03	1.08	1.08	1.97	1.99	4.11
S0T2	1	1.03	1	1.26	1.01	1.07	1.01	1.11	1.11	1.92	1.92	4.01
S1T1	1.26	1.24	1.26	1	1.27	1.22	1.27	1.13	1.14	2.49	2.49	4.85
S2T0	1.01	1.03	1.01	1.27	1	1.08	1	1.12	1.12	1.93	1.91	4.15
S2T1	1.08	1.03	1.07	1.22	1.08	1	1.08	1.02	1.02	1.93	1.91	3.37
S2T2	1.01	1.03	1.01	1.27	1	1.08	1	1.13	1.12	1.94	1.92	4.12
S3T0	1.11	1.08	1.11	1.13	1.12	1.02	1.13	1	1	2.12	2.09	4.45
S3T2	1.11	1.08	1.11	1.14	1.12	1.02	1.12	1	1	2.1	2.11	4.43
S1T0	1.92	1.97	1.92	2.49	1.93	1.93	1.94	2.12	2.1	1.01	1.01	2.42
S1T2	1.89	1.99	1.92	2.49	1.91	1.91	1.92	2.09	2.11	1.01	1	2.44
S3T1	4.03	4.11	4.01	4.85	4.15	3.37	4.12	4.45	4.43	2.42	2.44	1

Fig. 5. Comparison of the success of the different S-T combinations against all others when measuring the number of times each player's tokens landed on a p-type board position (and therefore had to pay the opposing player).

	S2T2	S2T0	S0T1	S1T1	S3T2	S3T0	S0T2	S0T0	S2T1	S1T2	S1T0	S3T1
S2T2	1	1	1.07	1.32	1.03	1.02	1.02	1.01	1.07	2.01	2.03	2.26
S2T0	1	1	1.08	1.32	1.03	1.03	1.02	1.02	1.08	2	2.01	2.27
S0T1	1.07	1.08	1	1.28	1.09	1.09	1.09	1.09	1.14	2.07	2.05	2.33
S1T1	1.32	1.32	1.28	1	1.37	1.36	1.33	1.32	1.42	2.5	2.49	2.9
S3T2	1.03	1.03	1.09	1.37	1	1	1.01	1.02	1.01	1.86	1.86	2.1
S3T0	1.02	1.03	1.09	1.36	1	1	1.01	1.02	1	1.85	1.86	2.11
S0T2	1.02	1.02	1.09	1.33	1.01	1.01	1	1	1.06	1.96	1.95	2.21
S0T0	1.01	1.02	1.09	1.32	1.02	1.02	1	1	1.06	1.93	1.95	2.21
S2T1	1.07	1.08	1.14	1.42	1.01	1	1.06	1.06	1	1.66	1.67	1.85
S1T2	2.01	2	2.07	2.5	1.86	1.85	1.96	1.93	1.66	1	1	1.23
S1T0	2.03	2.01	2.05	2.49	1.86	1.86	1.95	1.95	1.67	1	1	1.23
S3T1	2.26	2.27	2.33	2.9	2.1	2.11	2.21	2.21	1.85	1.23	1.23	1

Fig. 6. Comparison of the success of the different S-T combinations against all others when measuring the degree of bounce-back and the number of times each player's tokens landed on a p-type board position together (*i.e.*, total payments to the opposing player due to factors that can be directly controlled).

As can be seen from Fig. 6, the three most successful S-T combinations from Fig. 2 (S3T1, S1T0, and S1T2) are also the ones that involve paying the opposing player the least number of times in total (due to factors that can be directly controlled). The observations made previously on S3T1's behavior based on the results shown in Fig. 4 and Fig. 5 show that it is more important to avoid falling on a p-type board position than to be wary of having tokens on vulnerable s- and e-type board positions (in which the token may be bounced back by one's opponent). This might seem obvious at first because if a token falls on a p-type box the certainty of having to pay the opponent is 100%, whereas a token that is on an s- or e-type box is not certain to have to pay. However, there are many factors that make the situation more complex than it may initially seem.

First, the longer a token in a vulnerable board position stays there, the more it is exposing itself (and the chances that the token will eventually be bounced back and the opponent will have to be paid increase). Second, if the opposing player is looking to take advantage of this type of situation explicitly, the token's vulnerability increases. Finally, an additional situation in which a player may have to make a payment in Patolli is when one of the opponent's tokens reaches its end position. How frequently this happens cannot be influenced directly by one's decisions during play. One might try, indirectly, to minimize the probability that the opposing player's tokens will reach their end position (perhaps by trying to bounce them back as often as possible). However, as we have seen, guarding against bounce-back (the defense that the opponent may apply against this type of strategy) is not as important as avoiding direct payments by falling on p-type boxes.

Fig. 7 shows the results of comparing each possible S-T pair to all others when measuring the total number of times each player landed on an extra-turn board position.

	S3T2	S3T0	S2T2	S2T0	S1T2	S1T0	S0T2	S0T1	S0T0	S3T1	S2T1	S1T1
S3T2	1	1	1.1	1.1	1.1	1.1	1	1	1	1	1.1	3.4
S3T0	1	1	1.1	1.1	1.1	1.1	1	1	1	1	1.1	3.4
S2T2	1.1	1.1	1	1	1.2	1.2	1	1	1	1	1.1	3.2
S2T0	1.1	1.1	1	1	1.2	1.2	1	1	1	1	1.1	3.2
S1T2	1.1	1.1	1.2	1.2	1	1	1.2	1.2	1.2	1.1	1	3.5
S1T0	1.1	1.1	1.2	1.2	1	1	1.2	1.2	1.2	1.1	1	3.5
S0T2	1	1	1	1	1.2	1.2	1	1	1	1	1.1	3.2
S0T1	1	1	1	1	1.2	1.2	1	1	1	1.1	1	3.4
S0T0	1	1	1	1	1.2	1.2	1	1	1	1	1.1	3.2
S3T1	1	1	1	1	1.1	1.1	1	1.1	1	1	1.1	2.8
S2T1	1.1	1.1	1.1	1.1	1	1	1.1	1	1.1	1.1	1	3
S1T1	3.4	3.4	3.2	3.2	3.5	3.5	3.2	3.4	3.2	2.8	3	1

Fig. 7. Comparison of the success of the different S-T combinations against all others when measuring the number of times each player's tokens landed on an extra-turn board position.

As can be seen in Fig. 7, most combinations are comparable according to this measurement, but S1T1 is less likely than all the others to obtain extra turns as the game proceeds. This directly correlates with the fact that S1T1 was the worst overall combination according to Fig. 2. Getting extra turns increases the chances of winning matches.

Fig. 8 shows the results of comparing each possible S-T pair to all others when measuring the total number of turns the players had in their series of games. It can be seen in Fig. 8 that the same combination which wins the least number of extra turns, S1T1, is also the combination that has the less number of turns in a game (shown in Fig. 7). This was the most efficient game-playing combination (it made the least number of moves during its games), as shown in Fig. 8, but efficiency turns out not to be desirable if what one wants is to succeed at bankrupting one's opponent. This again reinforces the fact that striving to get more turns helps maximize one's chances of winning matches.

	S0T0	S0T1	S0T2	S1T0	S1T2	S2T0	S2T1	S2T2	S3T0	S3T1	S3T2	S1T1
S0T0	1	1	1	1.02	1.02	1	1.01	1	1	1.01	1.01	1.1
S0T1	1	1	1	1.02	1.02	1	1.01	1	1	1.01	1	1.11
S0T2	1	1	1	1.02	1.02	1	1.01	1	1.01	1.01	1	1.1
S1T0	1.02	1.02	1.02	1	1	1.03	1	1.03	1.02	1.02	1.02	1.12
S1T2	1.02	1.02	1.02	1	1	1.03	1	1.03	1.02	1.02	1.02	1.12
S2T0	1	1	1	1.03	1.03	1	1.01	1	1.01	1	1.01	1.1
S2T1	1.01	1.01	1.01	1	1	1.01	1	1.01	1.01	1.02	1.01	1.1
S2T2	1	1	1	1.03	1.03	1	1.01	1	1.01	1	1.01	1.1
S3T0	1	1	1.01	1.02	1.02	1.01	1.01	1.01	1	1.01	1	1.11
S3T1	1.01	1.01	1.01	1.02	1.02	1	1.02	1	1.01	1	1.01	1.09
S3T2	1.01	1	1	1.02	1.02	1.01	1.01	1.01	1	1.01	1	1.11
S1T1	1.1	1.11	1.1	1.12	1.12	1.1	1.1	1.1	1.11	1.09	1.11	1

Fig. 8. Comparison of the success of the different S-T combinations against all others when measuring the number of turns each player had during the series of games.

Fig. 9 shows the results of comparing each possible S-T pair to all others when measuring the average number of tokens the players had on the board at a given time during the series of games.

	S3T1	S0T1	S1T1	S2T1	S2T2	S2T0	S0T2	S0T0	S3T2	S3T0	S1T2	S1T0
S3T1	1	1.16	1.06	1.41	1.39	1.37	1.42	1.41	2.46	2.45	2.28	2.28
S0T1	1.16	1	1.12	1.29	1.32	1.32	1.35	1.34	1.74	1.73	1.77	1.78
S1T1	1.06	1.12	1	1.31	1.3	1.3	1.32	1.32	2.08	2.07	1.88	1.87
S2T1	1.41	1.29	1.31	1	1.03	1.04	1.01	1.01	1.38	1.38	1.39	1.39
S2T2	1.39	1.32	1.3	1.03	1	1	1.04	1.03	1.45	1.43	1.58	1.58
S2T0	1.37	1.32	1.3	1.04	1	1	1.04	1.04	1.46	1.45	1.58	1.57
S0T2	1.42	1.35	1.32	1.01	1.04	1.04	1	1	1.42	1.4	1.53	1.53
S0T0	1.41	1.34	1.32	1.01	1.03	1.04	1	1	1.42	1.41	1.53	1.53
S3T2	2.46	1.74	2.08	1.38	1.45	1.46	1.42	1.42	1.01	1	1.17	1.17
S3T0	2.45	1.73	2.07	1.38	1.43	1.45	1.4	1.41	1	1	1.17	1.18
S1T2	2.28	1.77	1.88	1.39	1.58	1.58	1.53	1.53	1.17	1.17	1	1
S1T0	2.28	1.78	1.87	1.39	1.58	1.57	1.53	1.53	1.17	1.18	1	1

Fig. 9. Comparison of the success of the different S-T combinations against all others when measuring the average number of tokens each player had on the board at a given time during the series of games.

As Fig. 9 shows, S3T1, one of the most successful combinations according to Fig. 2, had an equal or higher number of tokens on the board, on average, than all the other S-T pairs. However, S1T0 and S1T2, which were the other two very successful overall combinations, are among those S-T pairs with the lowest number of tokens. Therefore, the number of tokens that a player maintains on the board does not seem to be a predictor of success. If we compare S0T0, S0T1, and S0T2 in Fig. 9, we can also see that our prediction that T1 would cause dispersal of a player's tokens (and therefore a smaller number of them on the board, on average), T2 would cause bunching up of the tokens (and therefore a larger number of them on the board, on average), and T0 would be somewhere in between these two alternatives, was completely wrong.

After this analysis of our experimental results we can conclude that even when very simple strategies are used to assign move weights to tokens, and when very simple criteria are used to choose amongst several tokens that may end up with the same (best) move weight, these two factors involved in deciding which token to move interact in very complex ways. These interesting interactions sometimes lead to unexpected results.

We have tried to explain the results we obtained, but intend to continue exploring these issues further. Perhaps we can think of other variables to measure in order to acquire a better understanding of the game strategies we have implemented. More complicated strategies (in which a token would analyze more or different types of information in order to calculate its move weight) and tie-breaker criteria can also be implemented and tested. It would also be a good idea to compare the results we have obtained for Patolli with the behavior of other, similar, multi-token board games. Finally, we need to explore the interactions that occur when more than two players are involved in a game. In the meantime, our conclusion is that the most successful Aztec players probably made their decisions on which moves to make during the game of Patolli in a way similar to our S3T1, S1T0, or S1T2 combinations (described above).

REFERENCES

[1] A. Caso, "Un Antiguo Juego Mexicano: El Patolli", Ed. El Pirata, Escuela Nacional de Antropología e Historia, Xalapa, Ver., México, 1987.

[2] A. Caso, *El Pueblo del Sol.* México, D.F.:Fondo de Cultura Económica, 1971, pg. 21, pg. 106.

[3] B. de Sahagún, *Historia General de las Cosas de Nueva España,* 2nd ed., libro VIII, cap. X, Porrúa, México , 1969, pg. 300.

[4] J. de Torquemada, *Monarquía Indiana*, vol. IV, libro XIV, Universidad Nacional Autónoma de México, México, 1964, pp. 344-345.

[5] D. Durán, *Historia de las Indias de Nueva España e Islas de Tierra Firme*, tomo II, México, 1880, pg. 238.

[6] Federación Mexicana de Juegos y Deportes Autóctonos y Tradicionales, A.C. (Mexican Federation of Traditional Games), http://www.codeme.org.mx/autoctonoytradicional/Destreza/Patolli.html.

[7] M.J. Findler, M.E. Fendley, S. Narayanan, and R. Hill, "Using Trouble® to Compare Egocentric and Network-Centric Strategies," in *Proceedings of the Eighth Annual Conference on Industrial Engineering—Theory, Applications, and Practice*, Las Vegas, Nevada, U.S.A., November 10-12, 2003, pp. 754-760.

[8] R. Mikkulainen, "Creating Intelligent Agents in Games," in *Proceedings of the National Academy of Engineering 2006 Conference on Frontiers of Engineering*, http://www.nae.edu/nae/naefoe.nsf/weblinks/GBAN-6UQNMK?OpenDocument.

[9] http://www.azteca.net/aztec/nahuatl/game.shtml.

[10] http://www.xtec.es/~rbernau1/tauler/cst/patolli.htm.

[11] P. S. Neeley, http://www.xmission.com/~psneeley/Shareware/aztecpat.htm.

[12] Patolli, *Enciclopedia Libre Universal en Español*, http://enciclopedia.us.es/index.php/Patolli.