UNIVERSITY OF CALIFORNIA
Los Angeles

Game Theoretical Solutions
in Blackjack and Chess
via a Markov Decision Process Algorithmic Analysis

A thesis submitted in partial satisfaction
of the requirements for the degree
Master of Applied Statistics

by

Livingston Arthur Albritten

2022

ABSTRACT OF THE THESIS

Game Theoretical Solutions
in Blackjack and Chess
via a Markov Decision Process Algorithmic Analysis

by

Livingston Arthur Albritten
Master of Applied Statistics
University of California, Los Angeles, 2022
Professor Frederic R. Paik Schoenberg, Chair

In this paper I will first describe the theoretical background of a certain kind of random process, namely, Markov Processes, and subsequently, the potential for their application in a game theoretic context. Consequently, these random processes can be used in a variety of game theoretic contexts to construct solutions to games whose state space can be realized as a discrete graph of nodes and the edges between them. Once the details of how Markov Decision Process based strategies may be implemented to develop convergent solutions are settled, this method will be tested by evaluating the performance of these solutions in Tic-tac-toe, Blackjack and Chess.

The thesis of Livingston Arthur Albritten is approved.

David Anthony Zes

Nicolas Christou

Hongquan Xu

Frederic R. Paik Schoenberg, Committee Chair

University of California, Los Angeles

2022

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

Introduction

A principal component of the requirements of completion of the UCLA Graduate MAS curriculum is the requirement that students perform statistical research on a data set which demonstrates the application of statistics to the domain from which the data set is sourced. In accordance with this requirement the work here aims to present the structure of the data, algorithmic aims, statistical considerations, and experimental results of the application of Markov Decision Processes to construct solutions to a variety of games: Tic-Tac-Toe, Blackjack and Chess. In previous literature the Markov Decision Process framework has been used to train elementary navigation systems, but the method here proposes a slight adaptation of the original interpretation of the upper confidence bounds computed to inform the MDP in order to develop mixed-strategy solutions to each game considered here rather than deterministic solutions as yielded by the original online expert selection implementation. The adaptation of the discrimination of win rate statistics considered in this work helps motivate important code implementation considerations as well as the choice of the domain of the parameter settings. In any case, the bulk limit of a number of winning rate statistics should be collected/observed in order to form a memory which is used to generate the Markov measure associated with the MDP. Notably, the experimental evolution of a given game solution can be understood as a sequence of memory updates, corresponding to the consolidation of novel game data as winning rate statistics in the memory, creating a new pushforward Markov measure with each update.

The concept of a mixed-strategy equilibrium was introduced by John von Neumann and Oskar Morgenstern in their 1944 book The Theory of Games and Economic Behavior, with the analysis being restricted to zero-sum games, their work showed that a mixed-strategy equilibrium exists for any zero-sum game with a finite set of actions. Tic-tac-toe and Blackjack are in fact zero sum games with finite set(s) of actions, however Chess is differentiated as a zero sum game with an uncountable set of actions. The discussion and experiments here hope to demonstrate the tractability of the development of automated mixed-strategy solutions with the Markov Decision Process in the context of the three games, tic-tac-toe, blackjack and chess.

In the evaluation of the performance of the strategies developed in this work's experiments, it is good to note that there are not any guarantees of optimality (at least over the entire space of mixed-strategy solutions). The mixed-strategy solutions developed are local optima, well characterized by the implementation enabled parameterization selected before experimentation, but they are not necessarily global optima since the whole state space is not explored in the convergence of a given solution. The results of the experiments show that for each unique solution given by a unique parameter setting, the performance starts out as average or sub-par and later when the MDP strategy is turned on (an intermediary/late phase of experimentation with a strictly positive, non-zero parameter value), the performance creeps upward as new games are played and the strategy is gaining experience, however at some point the performance plateaus as there are fewer and fewer novel experiences for the algorithm to incorporate into the mixed-strategy solution. The size of this upward move is influenced by the choice of parameter settings, and so there is a best choice of parameter setting that yields the best local solution.

In the traditional consideration of Nash equilibria, a unilateral change is tested by an researcher seeking to validate the optimality of the equilibrium as it is presented, but in the context of this work any potential strategy changes are inspired by the arrival of new information from novel game data which indicates that bulk limit of the winning rate of a particular action may be higher than what the MDP memory statistics currently give evidence for. In this case, the novel game data will yield an empirical win rate statistic which is higher than the empirical win rate statistic recorded in the MDP memory and so when the new data is added to the memory, the action will be selected more often than before because the updated win rate was supplemented by the novel game data. The idea is that new data adds to underestimated winning rates and dampens overestimated winning rates until the true bulk limit of each winning rate is discovered. Importantly, every winning rate is a function of how frequently the action is chosen, so the parameter settings that are responsible for determining this frequency are notably influential in the convergence of the bulk limits of the winning rates. As a particular MDP solution is undergoing its evolution from average/sub-par to locally optimal, it is because the mixed-strategy is being updated to perform more rewarding moves, more often and because the games here are zero-sum games, each novel win can hopefully provide important information on how rewarding a move can be.

John Nash was able to build on Morgstern and von Neumann's equilibrium result to find that there exists an equilibrium to every game with a finite number of players and finite number of pure strategies by characterizing this solution as a Brouwer fixed point in the entire space of mixed-strategies under a map that moves solutions closer to optimality functioning on the utility gain of unilateral changes to the strategy, since known as a Nash equilibrium. It is important to note that the map used in the actualization of the markov process to construct solutions results in a Brouwer fixed point which is only a local optimum while the traditional Nash equilibria are global optima under all candidate unilateral changes. In this work, the locally optimal final solutions are found here by using an increasing sequence of parameter settings to drive the observed win rates of each individual action to a bulk limit. Different choices of the parameter settings will determine different bulk limits for the individual actions, the sequence of allowable actions dictated by the graph underlying the game's allowable states (vertices) and actions (edges) determines how altering a mixed-strategy to perform one action may also alter the bulk limit of the win rates of other actions that may be performed subsequently within a single game. Still, the convergence conditions of the MDP algorithm used in the experiments here that generates automated solutions is equivalent to the existence of a Brouwer fixed point in the form of a mixed-strategy, represented by the concentration of measure of the bulk limits of the winning rates of allowable actions.

Really, this describes how the strategy seeks to build a sub-graph from the graph underlying a given game, with the incidence of an edge in the newly forming sub-graph corresponding to the bulk limit of the win rate of the game-action associated with that edge being larger than a threshold determined by the bulk limits of all other edges sharing an out node with the edge of interest. The domain choice of the parameter settings, which will be described in detail more below, determine why it is that the convergence of the MDP mixed-strategy solutions fails to explore the whole space of mixed solutions. There are multiple ways to map the win rates of all allowable actions to a probability distribution over all allowable actions, but a given choice of a parameter setting will only employ one of these maps. The map that the MDP uses to construct probability distributions (Markov measures on the graph underlying the game), or mixed-solutions, is a many-to-one mapping. The details of how this computation is performed will be explained more in Chapter 2 and 3, but, if one set of winning rates is any positive real multiple of another set of winning rates then those two sets of winning rates will map to the same probability distribution. Or more specifically, if two distinct sets of winning rates of the allowable actions are constant multiples of each other, when partitioned into subsets of actions which share a common starting point, and this multiple is determined by the common starting point (for any and every possible common starting point, or stage of the game $s \in V$) $\exists \ d_s > 0 \ | \ w_{s,i}^{(01)} = d_s * w_{s,i}^{(02)} \ \forall \ i \ \in AA_s \, , \, s \in V$ then these two sets of winning rates will map to the same Markov measure from which actions will be sampled to construct a mixed-strategy to a given game. Note that $W^{(01)}$ and $W^{(02)}$ are distinct sets of winning rates (generated from two distinct sequences experiments) on the same actions, all allowable actions i at every reachable stage s of the game.

$$W_s^{(01)} = \{w_{s,i_1}^{(01)} \, , \, w_{s,i_2}^{(01)} \, , \, ...w_{s,i_{|AA_s|}}^{(01)}\}, \ \ W_s^{(02)} = \{w_{s,i_1}^{(02)} \, , \, w_{s,i_2}^{(02)} \, , \, ...w_{s,i_{|AA_s|}}^{(02)}\} \, , \forall \ i \ \in AA_s \, , \, s \in V$$

$$AA_s = \{i_1 \, , \, i_2 \, , \, ...i_{|AA_s|}\}$$

The MDP strategy is tractable because it actualizes a percolation process on the directed graph underlying the game to build a sub-graph whose edges correspond to actions which are included in the final mixed-strategy at the conclusion of the experiments. In a possibly unfamiliar step, this percolation process has a variable edge inclusion probability which is determined by the relative sizes of the winning rates of all allowable actions (all outgoing edges from a given vertex, including the edge/action being considered as an addition to the percolated sub-graph) at a given stage (vertex) and the parameter settings as well as whether or not it is reachable from vertices which have already been nabbed in a previous percolation bubble (the determination of this variability is what makes the MDP map 'data dependent'). As a 'variable edge inclusion probability percolation process', the edge inclusion probability is more of an edge specific indicator function on the characteristics of that given edge, like the edges which it shares an outgoing vertex with, the reachability of the given edge from vertices visited in old data, the parameter settings and also stochastically evolving characteristics like the winning rates of the edges of interest, but in this format we can view each pushforward MDP mixed-strategy/memory update as an empirical data dependent percolation process recreating a new sub-graph with each memory update or parameter value update. A new choice of parameter settings can yield a new percolated-sub graph, differentiated from the old by a different variable edge inclusion probability but also a potentially different set of winning rates and history of visited vertices if the experimenter wishes to change the parameter settings *and* collect new data. However, this is not always necessary and a new percolated sub-graph can be created with a new set of parameter settings by operating on any aggregation of data previously collected by a number of different previous parameter settings of MDP experiments (a hint toward the idea that after enough data has been collected, particularly in the uniform sampling phase, this data-dependent variable edge inclusion probability percolation process should have an expected behavior, somehow).

As long as this edge inclusion probability is low enough (given by large parameter settings far away from 0) then the percolated sub-graph will be sufficiently small (compared to the full graph) and this enables the practical tractability of our solution. The uniform sample (all parameter setting $= 0$) that initializes the MDP memory serves as a memoryless BFS (edge inclusion probability $= 1$ for all edges reachable from previously visited vertices) over the graph underlying the game. This is an attempt to introduce the strategy memory to all reachable stages of the game $s \in V$ by performing (almost-all) of the allowable actions, or traversing all edges in the graph associated with the given game $i_k \in AA_s \subset E, \cup_{s \in V} AA_s = E$. When the MDP is turned on and used to select actions/edges, the percolated sub-graph can be characterized by one or more of its unique minimum spanning trees (when operational, the MDP parameter setting $a_{t_{MDP}} \to \infty$ will be sufficient to ensure that the edge inclusion probability approaches a low value far from 1 but strictly greater than and concurrently close to 0). The edges of the sub-graph are like a to-do list for testing if the pushforward Markov measure has exhibited concentration of measure to a local equilibrium and so the development of the percolated sub-graph serves as a canopy over characteristic spanning trees which narrow the focus of the MDP to an important corner of the state space of allowable probability distributions (mixed strategies).

However, if the approach is changed to try to cover the whole state space of probability distributions over the graph then these future experiments will have a much longer run time because there are no conditions by which to narrow the search, it must simply mix pure strategies and all prior candidate mixed strategies together until it has an approximately full sample of what is at best a discretization of the continuous state space of probability distributions over all allowable actions.

If we wanted to explore the whole state space of mixed-strategies we would have to choose a one-to-many mapping from winning rates to probability distributions or we could simply enumerate the whole state space of probability distributions (mixed-strategies) as described above, but this will require a lot more work with the code implementation and data management and so it is easy to see that the brute force search of the entire state space of mixed-strategies may not be sensible. The mapping from winning rates to mixed strategies used by the MDP is an important piece of machinery because the instantiation of this map is a convenient and quick way for the MDP to test candidate changes to the mixed strategies. Since this map is a many-to-one function, it focuses the MDP's attention on a small corner of the entire space of mixed-strategies and so while the sequence of mixed-strategies produced by alternating between collecting winning rates as the results of game experiments and generating probability distributions as candidate solutions to future game experiments is not at all equal to the continuous pushforward map under which Nash equilibria are Brouwer fixed points, the MDP map hopes to accomplish a similar goal while also cutting down on computation costs with a more data dependent approach to morphing candidate mixed-strategies. After all, the experiments running the MDP strategies hope to produce a final probability distribution that is its own kind of Brower fixed point under the data dependent MDP pushforward map described above, and also in more detail in Chapter 3.

CHAPTER 2

Background

In order to satisfy the thesis requirement of the UCLA Statistics Graduate MAS program I have conducted a research project which explores the application of probabilistic/statistical measurement of Markov Decision Process' to Blackjack and Chess. In particular, computer generated Nash equilibria yield a very interesting class of decision/optimization problems to be studied. Solutions to games whose state space can be realized as a graph $\mathcal{G} = \{V, E\}$ (discrete mathematical structures defined by vertices or nodes and relations between them or edges) are valuable representations of candidate solutions which may exist in the Pareto frontiers of computer generated Nash equilibria of zero sum games.

The game theoretic formulation of both Blackjack and Chess with their respective "state spaces" given by the space of all allowable card combinations/board arrangements offers an interesting platform on which powerful probabilistic/statistical analyses can be performed. For example, consider that the rooks graph (the graph of all allowable rook moves on a chess board) is isomorphic to the latin square graph. Inside each single rank and file of the chess board, every square is reachable with the rook, so the cartesian product of the K_8 clique (or the complete graph on 8 vertices) with itself, is a concrete representation of the rooks graph. Also, if any two vertices that share an edge in the resulting cartesian product are forbidden to share a labeling chosen from a set of size/cardinality equal to 8, then this graph representation can be matched to a set of appropriately labeled isomorphic latin square graphs. The rooks graph is a subset of the total state space of the game of chess and so the concrete algebraic representation of the rooks graph can help motivate a more rigorous statistical analysis which partially explains this work's affinity for the convenience of being able to take disjoint unions over the set of same color pieces on a chess board or the sets of allowable actions with a hand value of 17: {hit}, {stay} in blackjack to produce a candidate probability distribution representing a solution to the respective games, as well as the theoretical abstraction it lends to the background discussion.

An unfortunate feature of the statement above is that the rooks graph is only a *subset* of the total state space in chess (like the hand value 17 in blackjack), but still, this observation lends itself to a natural partition of the state space. Importantly enough, the Markov Decision Process formulation can also be interpreted as a form of what is known as the Multi Armed Bandit problem in which a problem is partitioned into stages and the performance of the algorithm is maximized by maximizing the utility of which arm is pulled at each stage. Essentially, the MDP/MAB formulation takes advantage of the isomorphic mapping from chess' state space to markov chain's underlying graph whose vertices derive their out degree from the cardinality of the allowable moves associated with the board arrangement matched to the given vertex. This framework is particularly convenient because every unique allowable move is assigned to a corresponding unique directed edge in a graph whose vertices correspond to the states of a given game and whose directed edges correspond to the allowable actions pointing from the states where the actions are allowed and pointing to the states that result from selecting the action/move.

Ultimately, the MDP/MAB formulation succeeds by observing that given an empirical sample of a sequence of games, we can formulate a strategy by simply choosing the arm that has the highest empirical reward given our sample. In the experiments here, these rewards are winning rate statistics, more specifically these statistics are an element of $[0, 1]$ of the form $\sum_{n_j} \mathbb{1}_{k,win} n_j^{-1}$.

The simple closed form of this statistic lends itself to an interesting Martingale probabilistic analysis that helps set the foundation for the convergence conditions of the Markov Decision Process in previous literature and here. Once the bulk limit of the winning rate statistics have converged, the expected performance of the MDP strategy is a Martingale. The convergence of the Markov chain constructed for the MDP, combined with some thought about making sure that this Markov chain can focus on the most rewarding actions are motivation for the theoretical background discussion that hopes to inform the experimental results. As, $\sum_{n_j} \mathbb{1}_{k,win} n_j^{-1} < B = f(1, n_j)$, a high probability bound can be placed on the greedy maximal arm selection being "optimal" in some sense. This high probability bound allows us to calculate the divergence between our own empirical/greedy maximal arm selection and the optimal solution in the limit to infinity. Part of the cleverness of the solution is that the instantiation of winning rate statistics as random variables in the high probability inequality invoked makes the computation of divergence from optimality tractable in the limit to infinity. It can be shown that the ratio of the distance between our algorithmic solution and its optimal form and the number of games played by the algorithm approaches zero. In other words, the more novel game data we allow our algorithm to collect, the more it (via the provable convergence of a high probability inequality of interest) resembles it's own most optimal instantiation. Each implementation of the MDP strategy in the game theoretic contexts of the experiments in tic-tac-toe, blackjack and chess corresponds to a unique instance of an MDP solution, and under further consideration, there are different evaluations of the high probability inequality that indicate the paramaterization of further potential instantiative uniqueness. Because we have a way of thinking about how each instance of a strategy (or selection policy) is unique, it is convenient to think of optimality as being unique to each different instance of the MDP strategies developed in the experiments (this locally optimal solution is best described as a pushforward invariant Markov measure).

2.1 Application of MDP Strategy in Game Theoretic Contexts

Note that the MDP/MAB formulation when applied to chess amounts to choosing first *which* piece to move, and also, *where* to move this piece. And so, importantly, it incorporates a similar graph isomorphism similar to the one inspired by the rook graph's isomorphism to the latin square graph. Whereas of course the total state space of chess enumerated is isomorphic to the underlying graph of a unique markov chain, the rook's graph subset of this state space still provides an intersting perspective on the problem at hand. So long as a player has at least one uncaptured rook, at each subsequent stage that player has the option to move the uncaptured rook, and this is what makes the latin square graph isomorphism relevant to the MDP/MAB formulation. Interestingly enough, the latin square graph also arises in experimental design, which may partially explain its relevance to the MDP algorithmic design.

Typically, experimental design dictates which *data* is recorded, while algorithmic design dictates the functional characteristics of *operators* of that data. In this example, the data turn out to be the state space enumerated by the sequence of games played by the algorithm, and the functional characteristics of operators of the data are best represented by the Markov measure associated with a given MDP algorithm, and our ability to sample from this measure. Importantly enough, these two objects have similarity, but their semantic definition prevents them from truly being identical. Yes, the Markov measure is an artifact of algorithmic design, and the enumeration of the state space is a vital representation of the relevant data for this project (also known as unbiased, uniform experimental design), and yes a Markov measure corresponding to a game theoretical mixed strategy is a representation of the enumeration of a given game's state space, but the full state space cannot be enumerated for the game of Chess (this is not true for Tic-Tac-Toe and Blackjack which is why it is not a terrible idea to start our algorithmic analysis here). This correspondence between algorithmic design and experimental design and the parallel compatibility between the Markov measure and the enumeration of the state space is good to keep in mind as it helps explain the importance of the code that is responsible for querying the MDP algorithm: algorithmic operations on previously collected data uniquely determine the aggregation of new data.

Two popular axioms from set theory establish the logical background of the development of the Markov measure. The axiom of dependent choice informs our ability to sample from the discrete distribution by comparing its values, and the axiom of choice informs the basis of the existence of the discrete Markov measure attached to the union over edges of the underlying graph of the game.

However, much of the action is well described by what is not happening. Or more directly, the strength of the MDP algorithm lies in ability to construct a solution which places much of the state space in its complement. The algorithm is guaranteed to make an acceptable move at each stage of the game, making it a complete solution in some sense, but the final Markov measure's algorithmic representation of the state space is actually small in comparison to the state space in totality; once we enter the intermediary phase of experimentation, say by increasing the parameter from 0 to 1, then the resulting probability distribution over allowable actions is truly only supported by a subdivision of the whole space of allowable actions.

In many cases, the non-decreasing parameter values associated with each successive phase of experimentation should produce a sequence of mixed-strategy distributions whose entropy is non-increasing. Let $H(\hat{M}_{V_{end}^{(N)}})$ be the entropy of the empirical measure assigned to every reachable final stage of a given game after N games have been recorded in the memory. Then for a sequence of entropies associated with a sequence of candidate mixed-strategies each the result of successive, distinct phases of experimentation each with length n_b, the entropies should often be non-increasing (provided the initial uniform sampling phase is long enough). Note that this is an empirical measure over all possible final states of the game, which can be observed by keeping a histogram of the final state of each game recorded in the MDP memory during experimentation.

$$A_T = \{a_{t_0}, a_{t_1}, a_{t_2}, a_{t_3}...\}$$

$$H(\hat{M}_{V_{end}^{(N_t)}, a_{t_0}}) \geq H(\hat{M}_{V_{end}^{(N_t+n_b)}, a_{t_1}}) \geq H(\hat{M}_{V_{end}^{(N_t+2n_b)}, a_{t_2}}) \geq H(\hat{M}_{V_{end}^{(N_t+3n_b)}, a_{t_3}}) \geq ...$$

$$\mathcal{G}_{game} = \{V, E\}$$

$$V_{end} \subset V \,|\, AA_v = \emptyset \, \forall \, v \in V_{end}$$

The reduction of entropy and resulting concentration of measure toward the actions with high empirical winning rates is exciting; the strength of the MDP heuristic can be described as its ability to transform what are initially "average" solutions into approximately "optimal" solutions. What's more, is that upon close inspection, we may investigate different loss functions to quantify the strength of a given solution or even the strength of a process which yields a sequence of solutions. No matter their analytic properties, these loss functions must translate a given solution into a single scalar whose intensity is indicative of the solution's average behavior in some sense.

Consider the simple classic loss function $L(y,t) = c(y-t)^2$ or the vectorized version $L(y,t) = c(y-t)^T(y-t)$ where y is a given solution and t is a target or aim. While we may realize that there is no such thing as a truly final MDP solution, especially for games whose state space cannot be enumerated in the span of any known lifetime, experimental performance in the form of win/loss results from a sequence of games played by our MDP solution can be used to compute an empirical average that is meant to approximate the expected value of this loss function (in distribution) for the given solution. The computation/approximation of this empirical average will be thought about more in the experimental plan section with the discussion of $\mathbb{E}[L(y,t)]$. As an aside, the empirical measure over all final states mentioned in the consideration of entropy earlier is also sufficient data for approximating $\mathbb{E}[L(y,t)]$.

In practice, we may not even have a representation for t the target/aim or optimal solution, but we do have a way of knowing when the MDP solution representation has stable convergence to a final distribution, via the stable convergence of the bulk limit of action winning rate statistics. So, in summary, while we may not be able to exactly instantiate the classic loss function, we may be able to leverage the fact that we expect it to still be relevant to the MDP formulation in some sense.

One may observe that a state space enumeration can be performed for the game of blackjack, with the Markov chain's underlying graph denoting the allowable moves/arm choice associated with hitting or staying with a given card combination. The tractability of the enumeration of the blackjack state space makes it a good diving board for our algorithmic analysis, where the same analysis can be performed in deeper waters in the context of the game of chess.

2.2 Experimental Plan

Keeping in this train of thought I will perform a rigorous statistical study of the MDP/MAB algorithm first in Tic-Tac-Toe and Blackjack and then in Chess. This study will be composed of writing Python code for the burn in phase/iterations of the state spaces of the respective games. Given sufficient functional ability, this code will yield the empirical statistics needed for the MDP/MAB algorithm to demonstrate its convergence to the locally optimal solution(s). These empirical statistics, combined with the observed algorithmic performance of MDP/MAB will allow us to confirm the validity of the theoretically proven efficiency of our ”close to optimal in high probability” solution for all of the games. I have chosen to begin with Tic-Tac-Toe and Blackjack because these games can demonstrate the effectiveness of the strategy in a one player setting. In Tac-Tac-Toe, the strategy's effectiveness can be shown to help the O-player overcome the disadvantage of always having to move second to the X-player's first move. How this effectiveness translates to the game of blackjack can also be explored by allowing our strategy to play against the dealer in Blackjack, who has a fixed strategy according to the rules of the game. In each of these cases, only one player (in number and type) has access to the MDP memory (Player O in Tic-tac-toe and a single player in Blackjack). For subsequent experiments in chess, both players may have access to a single memory and train a sensible MDP strategy by recording the moves of both players in the MDP memory.

In particular, different evaluations of the high probability inequality of interest correspond to different convergence rates for the MDP/MAB algorithm. I would like to compare the performance of a number of different evaluations of inequality to understand the observed advantage of one algorithm over another. Given the theoretical foundations of the MDP strategy formulation, as well as the Python implementation of MDP memory data structures for Tic-Tac-Toe, Blackjack and Chess and the data sets accompanying each distinctly tuned MDP strategy, the following analysis can focus on verifying the algorithmic performance of the MDP strategy according to our expectations given the parameters.

In the experimentation I may not be able to exactly verify the accuracy of the convergence of the algorithm to the optimal solution in the limit $\lim_{n\to\infty} n^{-1} D_{MDP,OPT}(n)$ to show that the performance exhibited by the MDP/MAB algorithm does match $D_{MDP,OPT}(n)$ as formulated in the Probabilistic Design/Analysis of the algorithm. However, I can still seek to gain a more qualitative understanding of the strength of the solution yielded by the MDP algorithm by observing it's most populated late stage strategies. When performing the game experiments, the MDP algorithm needs a seed–a sequence of previously played games, the

MDP's seeded memory, and the results of it's newly selected strategies combine to create a newly formed memory which dictates the collection of new data which may be consolidated into the memory in a future memory update.

Each time the MDP selects a move according to its seeded memory, which generates the MDP distribution over edges of the game's underlying graph, it updates the memory: a collection of winning rates for every allowable action that has been used by the strategy in a previous round of experimentation.

$$W_{j,n_j} = \frac{\sum_{k=0}^{n_j} \mathbb{1}_{j(k)=win}}{n_j} \rightarrow \frac{\sum_{k=0}^{n_j+1} \mathbb{1}_{j(k)=win}}{(n_j+1)} = W_{j,n_{j+1}}$$

The variable $\mathbb{1}_{j(k)=win}$ is an indicator function conditioned on the outcome of the k-th time action j is performed by the strategy being a win, which is crucial to the calculation of the winning rate of a given action, after it has been performed a given number of times W_{j,n_j}.

Earlier it was noted that we can start the algorithm with a burn in phase which uniformly samples the state space to generate a sequence of games. Later, we noted the importance of the MDP's ability to transform "average" solutions into "optimal" solutions, and this phenomenon is put into practice immediately as the initial strategy of using a uniform sample to select actions in games is almost exceptionally average, but still, with its affinity for the most rewarding actions, the MDP is likely to turn the solution toward a locally optimal solution, and away from disadvantageous solutions. The alternation between uniform sampling and the MDP selection procedure can also extend beyond the first stage of experimentation if it seems that the uniform sampling procedure can still supply novel game data to the MDP memory.

When we start the algorithm with the uniform sampling procedure, the strategy is simply looking at the board and choosing any allowable move with a probability relative to the inverse of the cardinality of allowable moves associated to the given board state $\hat{\mathbb{P}}(i) \sim \frac{1}{|AA_s|} \forall i \in AA_s | s \in V$. For as long the MDP is being supplied with new uniform samples (average solutions) it will be able to transform the game data into optimal solutions (so long as they hold information content, i.e. we hope the uniformly sampled moves are winning moves, and possibly even moves we have not considered before). When the experiments are focused on the game of chess, it may be useful to think more carefully about the transition from the uniform sampling procedure to the MDP selection strategy. In the experimentation with tic-tac-toe and blackjack, we can "close out" the uniform sampling procedure, meaning that at some point we do not have to update the memory with any moves chosen by the uniform sampling procedure any longer. Once enough experiments have been run in tic-tac-toe and blackjack, the memory "covers" the whole state space, meaning that the memory has an empirical score statistic for every possible board arrangement, or card combination. Note that while support of the memory being a covering space of the whole state space of the game is a very good goal (as it helps the memory be maximally, and uniformly informative to the map which produces mixed-strategies in the form of Markov measures), especially for finite action and state space games, the final MDP solution should still arrive at selection policy that only employs a select few of the whole set of allowable actions.

However, in the experimentation associated with chess, the size of the total state space of acceptable moves is too large for us to cover the whole space, which hints at the idea that it may not ever be clearly advantageous to "close out" the uniform sampling procedure in the implementation of the MDP chess strategy. A 100% uniform sampling strategy is a comfortable start, but how should the uniform sampling strategy and MDP selection strategy be comingled to produce a happy medium in which the strategy can utilize the focus of the MDP to yield more wins and also the exploration of new states provided by the uniform sampling procedure?

Furthermore, the longer the algorithm runs, the more we expect it to resemble the optimal instantiation of its parameters. Once the algorithm has gained enough experience, we can sample from the Markov measure more and sample from the uniform distribution less. Note that $\mathbb{E}[L(y,t)] \sim Var(y) + (\mathbb{E}(y) - t)^2$. And here the relevance to the classical loss function is observed, I will claim that resemblance to the optimal solution will yield a minority contribution to the $(\mathbb{E}(y) - t)^2$ term. However, in some sense, our understanding of the representation of t, the target solution, is dependent on the choice of parameterization, so it will be important to think about the conditions of convergence under paramterization.

Since we observed that the uniform sample was somewhat crucial to the (partial) enumeration of the state space necessary to kick start the MDP algorithm we set ourselves back in preparation to take steps forward. The initial burn in phase, a supply of uniform samples, is a reliable steady stream to $Var(y) = \mathbb{E}[(y - \mathbb{E}(y))^2]$. Being awakened to the MDP's ability to morph novel data from these average solutions into nearly optimal solutions, we can count on observing an empirically decreasing $(\mathbb{E}(y) - t)^2$ term, and so the only thing left to do is minimize $Var(y)$. We can accomplish this by making the uniform sample mixed-strategy a smaller fraction of the chosen moves over time (the uniform distribution contributes a lot to the loss as the equal use of every action available makes for a notably high variance over the whole span of available empirical winning rates, however it is also the maximum entropy distribution over actions). Currently the plan is to start with a $100\% = \frac{1}{K}, K = 1$ uniformly sampled initial burn in phase with a transition to a $\frac{1}{K}, K > 1$ uniform sample paired with a strategy which samples from a data dependent Markov measure with frequency $\frac{K-1}{K}, K > 1$. This plan can be updated to include a longer sequence of transitions where the uniform sample is stepped down from $1/K_0$ to $1/K_0^{\alpha_1}$, $1/K_0^{\alpha_2}$, $1/K_0^{\alpha_3}$, ... until it is a negligible proportion of the solution (where $\alpha_i > \alpha_j \iff i > j$).

The $(1/(K^q)) * U + (1 - (1/(K^q))) * MDP$ is an adequate description of every instance of the strategies used in this project's experiments. Where $(1/(K^q)) * U$ is the $1/K^q$ fraction of the strategy which is selected using a uniform sampling distribution and $(1 - (1/(K^q))) * MDP$ is the $(1 - (1/(K^q)))$ fraction of the strategy which is selected using the Markov measure. In particular, it is an effective way to paramaterize the loss function's interaction between the solution's variability and its contradicting search for optimality via concentration of measure. We can tune the rate of uniform sampling and MDP parameter settings to give the algorithm the path of least resistance to the optimal MDP based solution and to manage the trade off between deciding which term contributes to the cumulative loss, and when.

In particular, we are okay with the short term setbacks of a nearly maximal contribution from $Var(\hat{y})$ to the cumulative loss when we uniformly sample, observe the result and add to the memory, with the understanding that future contributions to the cumulative loss from $(\mathbb{E}(\hat{y}) - t)^2$ can always be made smaller with novel data, and also a particular range of increasing parameter values. Experimentation is meant to capture as much novel data as possible about $\hat{y}$, potential empirical performance of the MDP. This also shares an important connection to the consideration of entropy over final states and also an understanding of what is meant by the 'most optimal' instantiation the MDP for a given parameter.

The decreasing entropy over final states should also correlate with a decreasing $Var(\hat{y})$. The change from the initial uniform sample to the MDP Markov measure mixed strategy is the first occurrence of decreasing $Var(\hat{y})$ and the next occurrences are driven through their correlation to the non-decreasing parameters and corresponding non-increasing sequence of entropies of endgame measures.

The sequential reduction of entropy in the development of an MDP mixed-strategy is also meant to coincide with a path of resemblance to a locally optimal solution. Of course, an action's selection frequency influences it's winning rate, so there is no guarantee that the final empirically highest winning rates are the *true* highest winning rates in a more general sense, but we hope our initial uniform sample is big enough and unbiased enough to supply a good deal of information about action winning rates. The idea behind our attempt to find a locally optimal solution whose assigned measure on endgames yields a nearly minimal $\mathbb{E}[L(\hat{y}, t)]$, is to populate the memory with information about these endgame strageties and use the MDP parameterization to concentrate on lucrative endgame strategies which minimize $(\mathbb{E}(\hat{y}) - t)^2$.

The successively increasing parameters (integer powers of pushforward measures) should induce a concentration of measure toward the actions with the highest winning rates, meaning that these actions are a more consistent fixture of the strategy's cumulative action set, coinciding with the decreasing entropy and variance. If these actions are empirically valuable, then the strategy will continue to improve it's cumulative winning rate, or it has converged to its most optimal instatiation and so its empirical performance is a valid approximation of its expected performance. Later in Chapter 3 the concentration of measure induced by the sequence of increasing parameters will be considered as a convergence to an invariant Markov measure. Importantly, there is no guarantee Markov measure will produce a stationary distribution, but the parameter dependent convergence conditions that drive the Markov measure to exhibit Martingale behavior are in fact enough to consider the relevance of the unique pushforward invariant Markov measure used for the MDP strategy.

Note that each MDP strategy is naturally endowed with a probability measure on the state space of a given game, and so this probability measure is what gives substantial meaning to the strategy's "most populated" late stage strategies. In particular, the pushforward invariant Markov measure associated with each MDP parameterizaton should have an associated empirical stationary distribution on the leaves of the graph representing the late stage state space of a given game. These leaves can also be referred to as endgames. (For the sake of experimentation we can always construct an empirical stationary distribution by freezing the parameter value for a sufficiently long block of experimentation and using the empirical histogram over endgames)

In the case that the Markov measure satisfies enough regularity conditions to have a stationary distribution on endgames, the "most populated" late stage strategies are those strategies yielded by the MDP in the board arrangements/card combinations where the MDP has already been observed to prefer those arrangements over similar arrangements (i.e. arrangements that share many of the same previous moves but diverge later in the development of the game). In other words, the winning moves most often chosen by the MDP algorithm should provide a sensible lens through which a qualitative understanding of the tendencies of the strategies yielded by the rigorous MDP algorithmic analysis may be gained. In the next chapter, the end of experimental analysis section and also the blackjack strategy implementation in particular may help explain more how the most populated late stage stategies and also the resulting histograms of final game actions and final game states are relevant to the consideration of the approximation of $\mathbb{E}[L(y,t)]$.

CHAPTER 3

Experimental Analysis

In this section I will review results of the experimental analysis process I have described in the previous section. The experimental analysis will be composed of a sequence of evaluations of the MDP strategy in a few games, Tic-tac-toe, Blackjack, and Chess. The experiments rely on the implementation of the games and recording of the memory for the empirical Markov measure samples given in response to queries to the relevant players for their move selections. The background section has information on the theoretical underpinnings of the Markov Decision Process strategy formulation as well as ideas about evaluating different parameterizations of the MDP. This section provides more information about how the theoretical development of solutions are translated into experimental results.

A better understanding of how the MDP theory may be expressed to render automated solutions may help give context to which parameters are relevant, and will be explored in more detail in this section. For now, we can focus on the additive constant in the high probability inequality of interest. Because we seek to minimize the probability that the move we select is not far from the optimal move, at least in its reward (in the form of a winning percentage) as a winning move, we can realize that the additive constant that accomplishes this most effectively from the perspective of our high probability inequality, is the logarithm of the number of times a move is chosen. This thought can be investigated further, we may simply allow our desired error to be a polynomial function of the number of games played. Importantly, for any given action, the additive constant is placed in the numerator of the empirical winning rate, so our winning percentages take the form $\frac{(n_j W_{j,n_j} + log(n_j))}{n_j}$. We should hope to amass enough experiments to observe the sequence of logarithmic additive constants become negligible (and consistently more so) $\frac{(log(n_j))}{n_j} \rightarrow 0$, while the empirical winning rate

W_{j,n_j} must stay between 0 and 1. At any rate, we are interested in studying the behavior of the bulk limits of the empirical winning rates of each allowable action j in a given game:

$$\lim_{n_j \to \infty} W_{j,n_j} \; \forall \, j \in \bigcup_{s \in V} AA_s$$

While there are other additive constant that can be considered with the MDP like square root $(n_j)^{1/2}$ and polynomial $(n_j)^k$, we can be confident that the logarithmic additive constant that falls out of the algebraic manipulation of our high probability inequality is the best implementation of the additive constant parameterization of the MDP, it helps guarantee that the MDP will most often choose the move that has the highest empirical reward. However, more can be done with this understanding in the implementation stage, we can drop the additive constant altogether. As a function of the number of times an action is chosen, the logarithmic additive constant is the absolute slowest growing function we can choose, and once we divide by the number of times the action is chosen to yield a winning percentage, this becomes even more important. If we drop the additive constant altogether, it best approximates the MDP implementation with the logarithmic additive constant. There are many choices we could make to specify the additive constant in the implementation stage of the project. But in truth, the constant is a function of one variable, the number of times an action is chosen; the best additive constant is logarithmic, and the two slowest growing closed form algebraic functions are the constant 0 function and the logarithm. Since the constant 0 function is the best approximation to the logarithm in the limit to infinity, we might as well drop the additive constant in our experimental implementation. We can choose other additive constants, but they would be exact (or approximate) implementations of parameterizations of the MDP that are not as good as the logarithm (from the perspective of the chosen high probability inequality concerned with the convergence in probability of the maximum value winning rate(s) among actions associated with each stage of the game) and so we proceed with our empirical winning rates.

3.1 Experimental Evolution of Probability Distribution over Allowable Moves

The exploration of the behavior of the winning rates and an understanding of how they generate the probability distributions used by the automated MDP strategy to make selections may also motivate a closer look at the measure theoretic implications of the MDP selection procedure. In particular, the equation that references the memory update can be investigated a bit further.

$$W_{j,n_j} = \frac{\sum_{k=0}^{n_j} \mathbb{1}_{j(k)=win}}{n_j} \rightarrow \frac{\sum_{k=0}^{n_j+1} \mathbb{1}_{j(k)=win}}{(n_j+1)} = W_{j,n_j+1}$$

For any game with an accompanying sequence of experiments generated by the uniform and/or MDP selection procedure, every allowable move that has been visited in a sequence of previously observed experiments has a unique data set that generates its unique empirical winning percentage W_j. Because these winning percentages take the form of a closed form algebraic expression, we can think about the behavior of our algorithm through the lens of the evolution of these unique empirical winning percentages. Since a non-trivial set of winning percentages can generate an empirical probability distribution to sample allowable moves, the update process described above shows how each memory update to the selected moves creates a pushforward measure. Note that any given memory update always occurs after the conclusion of one or more games, so each update augments more than one action's empirical winning percentage with the results of the most recent game(s) as novel data. In particular, each memory update concurrently adds to the visit totals and win totals of more than one allowable action. This is also why the uniform sample burn in stage of experimentation helps drive the MDP selection policy to yield favorable convergence properties, a set of one or more complete games and a ledger of all of the actions chosen to yield the results of those games will always be candidate novel data to augment (necessarily) more than one allowable action empirical winning percentage's.

At any given stage of the game, the memory has winning percentages recorded for some subset of the allowable moves/actions currently available.

$$R_{mem,s} = \{W_i \, \forall \, i \in AA_s \mid \mathbb{1}_{n_i>0}\}$$

$$\hat{\mathbb{P}}_s(i) \sim W_i \mathbb{1}_{n_i>0} \; \forall \; i \in AA_s$$

$$\mathcal{G}_{game} = \{V, E\}, \;\; E = \bigcup_{s \in V} AA_s$$

Where AA_s is the set of allowable moves available and $R_{mem,s} \subset AA_s$ is the subset for which the memory has winning percentages recorded (at stage s of the game). $\mathbb{1}_{n_i>0}$ is an indicator function which ensures that a positive number of game data has been recorded in the memory for a given move i, and the following ensures that the empirical distribution $\hat{\mathbb{P}}$ assigns measure 1 to the set of allowable moves at every reachable stage s.

$$\sum_{i \in R_{mem,s}} \hat{\mathbb{P}}_s(i) = \sum_{i \in R_{mem,s}} M_{i,1} = 1 \;\; \forall \, s \in V$$

With the introduction of the pushforward measure, it is important to note that the type of measure we introduced actually belongs to a larger class of distributions. This set of interest for this discussion is the set of non-negative integer powers of pushforward measures.

$$\hat{\mathbb{P}}_s^{\mathbb{N}} = \{\hat{\mathbb{P}}_s^k \, \forall \, k \geq 0\} \; \hat{\mathbb{P}}_s^k \sim (W_i \mathbb{1}_{n_i>0})^k \; \forall \; i \in AA_s, \; k \geq 0$$

$$\hat{\mathbb{P}}_s^k(i) = M_{i,k} = \frac{(W_i \mathbb{1}_{n_i>0})^k}{\sum_{j \in AA_s}(W_j \mathbb{1}_{n_j>0})^k} \; \forall \; i \in AA_s, \; k \geq 0$$

The specification of the pushforward measures on any given state/stage (board state or card combination) s of a game is important because the MDP assigns measure to the allowable actions at each stage rather than to the states themselves. When thinking about the MDP strategy we should associate one measure, a probability distribution (or a set of probabilities) to each stage, and we should associate a single probability to each allowable action/move. This should help contribute to the understanding that the probability distribution generated by the MDP strategy is a markov chain on the allowable actions of the game.

And so while we may want to think about the stationary distribution on the vertices of the markov chain's underlying graph, i.e. the stages of the game, we should remember to save this thought for the conclusion of the experiment(s). Truthfully, this is a great way to observe how the increased parameter settings help drive a sequence of decreasing entropy stationary distributions over the union over all final stages of a given game. While the experiments are testing the performance of the MDP strategy, however, it is more important to consider the collection of distributions formed by composing the operation of sampling from the discrete distributions at each stage of the game to select each action (or move) in sequence by comparing the empirical, data-dependent winning rate values associated to each edge.

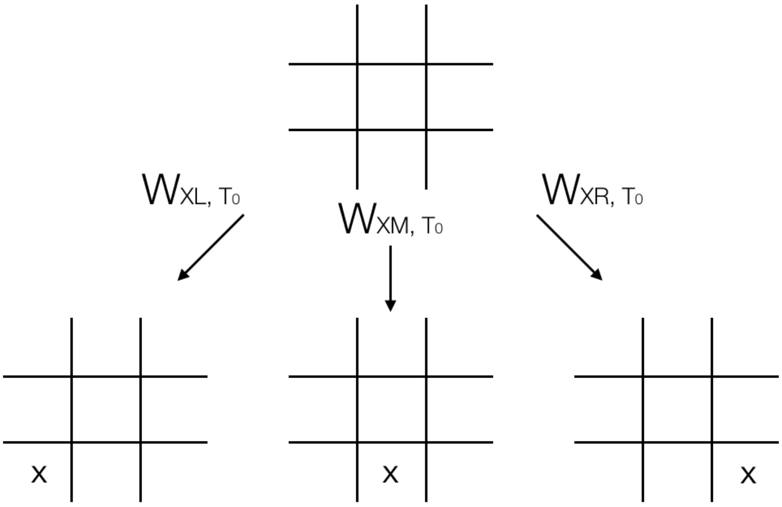


Figure 3.1: Graphical representation of markov chain with underlying graph associated to opening stage of tic-tac-toe

When employing one of the members of the set of non negative integer powers of the empirical distributions to construct a solution to a game in a sequence of experiments, the choice of the power has an important effect on the evolution of the distribution over edges that is associated to the markov chain's underlying graph (note that choosing zero as the power corresponds to a uniform sample). For values approaching infinity, the integer power of the empirical distribution makes the selection procedure choose the maximum value action (the largest empirical winning rate of the available allowable actions) with probability approaching 1. It is certainly feasible for a strategy to select the most valuable available action at every stage of the game (as is the goal in earlier MDP literature).

However this strategy can yield a number of different results, depending on the initial conditions. A good way to think about this with these experiments is that there are many different almost everywhere non-decreasing sequences, and each of these would yield a different final solution whose selection of the maximum value action available would be dependent on an underlying distribution unique to the non-decreasing sequence (as long as the MDP memory is translated into updated pushforward distributions with the arrival of new data). The $\hat{\mathbb{P}}^*$ distribution mentioned below is meant to represent the limit of the distribution arrived at by exponentiating the distribution with a power approaching infinity (the Kronecker delta distribution on the maximum value of the distribution δ_{i,AA_s^*} where AA_s^* is the maximum value action at stage s of a given game).

$$A_T = \{a_{t_j}\,;\, t_j \in \mathbb{N}\,|a_{t_m} \geq a_{t_n} \iff t_m > t_n\}$$

$$\lim_{t\to\infty} \hat{\mathbb{P}}_s^{a_t} = \hat{\mathbb{P}}_{s,A_T}^* = \delta_{i,AA_s^*}$$

The A_T set above is a non-decreasing sequence, any of which are acceptable candidates to be chosen as the set of parameters for a sequence of MDP experiments. Each single element of the non-decreasing set in order is a parameter for a single phase of a block of planned experiments. At a given phase, the corresponding parameter is the exponent applied to the winning rates to generate the markov measure (or probability distribution) which represents a mixed-strategy ($\hat{\mathbb{P}}_s^{a_t}$), and so importantly, when the experiment moves from one phase to another, the increased parameter selected from the non-decreasing sequence is meant to generate a new mixed-strategy which favors the most lucrative winning rate actions more than that of the previous phase of experimentation.

While we may not ever carry out an experiment with all of the non-decreasing sequences described above, our experiments will follow the path of a non-decreasing sequence that starts at 0 and increases until it holds its position at a finite value. The convergence properties of the non-decreasing sequence chosen influences the kind of convergence exhibited by the probability distribution $\hat{\mathbb{P}}_s$ and also $\hat{\mathbb{P}}_s^* = \delta_{i,AA_s^*}$.

It is important that we are aiming to generate a probability distribution because this means that the best way to characterize the kind of convergence exhibited by the probability distribution is convergence in distribution of the selection policy over allowable actions.

$$\lim_{n\to\infty} F_n(x) = F(x),\ \lim_{n\to\infty} E[g(\hat{\mathbb{P}}_{s,(n)})] = E[g(\hat{\mathbb{P}}_s)],\ g \in BC^0$$

$$BC^0 = \{W \subset C^0 \,|\, g \in W \iff |g(x)| < G < \infty\}$$

Where F_n is the CDF of the random variable i_n and F is the CDF of the random variable with support $x \in [0,1]$ (the PDF of i is the final mixed-solution assigned Markov measure M, $F(x) = \int f = \sum_{i:W_i<x} M_i$) and C^0 is the class of all continuous functions. In the analysis of the strategies here, the random variable i_n represents the strategy's allowable action selection after its n-th memory update and i is that of the final Markov measure (at a given stage of a game $i, i_n \in AA_s$). The logarithmic additive constant that we dropped above is relevant if one is primarily concerned with convergence in probability, but since the construction of the MDP memory stratifies the empirical winning percentages by allowable actions associated to a vertex matching a board state or card combination, our analysis is concerned with the convergence in distribution whose domain mapping covers a set of values rather than the convergence in probability, shown below, that tests the value of one statistic (the maximum value action).

$$\lim_{n\to\infty} \mathbb{P}(|\hat{W}_{AA_s^*,n} - W_{AA_s^*}| > \epsilon) = 0\,\forall\,\epsilon > 0$$

This is a crucial adaptation of the interpretation of the high probability inequality of interest. By considering the convergence in distribution rather than the convergence in probability, we are able to examine a wider variety of solutions. Now, the instantiative uniqueness of each solution rests on the characterization of an integer power of a distribution of values corresponding to a solution in the form of a probability distribution, rather than the sequences of maximum values of the relevant sets of statistics that correspond to a singular deterministic solution (represented by a finite number of singletons associated to the maximum value actions). Conveniently, the deterministic solutions represented by singletons are well described by the empirical distributions with integer powers approaching infinity, meaning they can also be evaluated as probability distributions with the tools of convergence in distribution described above.

As an aside, I think the view that deterministic solutions represented as singletons can also be approximated by empirical distributions of winning rates with integer powers approaching infinity is an important one; the original analysis seeks to characterize the behavior of individual winning rate statistics as the convergence in probability of random variables, and so this is why the high probability inequality of interest includes an additive constant to the winning rates to help account for the fact that less frequently chosen actions are less well informative to the strategy, and so they may actually be more lucrative than the strategy would have currently expected. However, the novelty of the adaptation of the analysis here is important as well; recognition of the view that we can approximate the greedy selection heuristic of always choosing the maximum value allowable action represented by singletons with very high integer powers of winning rate distributions over allowable actions also supports the shift in the analysis to characterize the behavior of the automated mixed-strategy at any given stage of a game, now treating the automated strategy's choice of action as the random variable of interest, and studying the convergence in distribution of this random variable.

We may still explore how unilateral changes in the direction of the most lucrative actions alters our expectation of a win while employing the strategy, but now we are considering more gradual changes to probability distributions (or mixed-strategies), rather than simply trying to collect enough information to decide which pure strategy is the best.

It is good to note the characteristics of the powers of the distributions to understand the spectrum of behavior that may be observed when using these distributions to construct strategies. An important feature of considering this class of distributions is that we can describe how increasing the power changes the distribution.

$$\hat{\mathbb{P}}_s^k(i) = M_{i,k} = \frac{(W_i \mathbb{1}_{n_i>0})^k}{\sum_{j\in AA_s}(W_j \mathbb{1}_{n_j>0})^k} \, \forall \, i \in AA_s$$

$$\frac{M_{i_1,l_j}}{M_{i_2,l_j}} = \frac{(W_{i_1}\mathbb{1}_{n_{i_1}>0})^{l_j}}{(W_{i_2}\mathbb{1}_{n_{i_2}>0})^{l_j}} > \frac{(W_{i_1}\mathbb{1}_{n_{i_1}>0})^{l_k}}{(W_{i_2}\mathbb{1}_{n_{i_2}>0})^{l_k}} = \frac{M_{i_1,l_k}}{M_{i_2,l_k}} \iff l_j > l_k,\, M_{i_1,1} > M_{i_2,1}$$

The equations above demonstrate that increasing the powers of the distribution is order preserving. If $(W_{i_1}\mathbb{1}_{n_{i_1}>0})^{l_j} > (W_{i_2}\mathbb{1}_{n_{i_2}>0})^{l_k}$ then when we sample from $\hat{\mathbb{P}}_s^{l_k}$ to select an action at stage s, we will select move i_1 more frequently than we select i_2. Additionally, when we increase the exponent from l_k to l_j, we increase the ratio corresponding to the frequency of choosing i_1 over i_2. The change in this ratio is described by the given relation of the ratio of assigned measure for moves i_1 and i_2 when taken to powers $l_j > l_k$.

$$\frac{M_{i_1,l_j}}{M_{i_2,l_j}} = \frac{(W_{i_1}\mathbb{1}_{n_{i_1}>0})^{l_j}}{(W_{i_2}\mathbb{1}_{n_{i_2}>0})^{l_j}} = \frac{(W_{i_1}\mathbb{1}_{n_{i_1}>0})^{l_k}}{(W_{i_2})\mathbb{1}_{n_{i_2}>0})^{l_k}} \cdot \frac{(W_{i_1}\mathbb{1}_{n_{i_1}>0})^{l_j-l_k}}{(W_{i_2}\mathbb{1}_{n_{i_2}>0})^{l_j-l_k}}$$

The indication that increasing the power is order preserving is important because we can uniquely characterize the probability distributions by their level sets. We know that if two actions at a given stage s_n have equivalent winning percentages, then a sample from $\hat{\mathbb{P}}$ will always choose these actions equally often, no matter what integer power of the distribution (of values) we choose. When we increase the power of the distribution, we assign more mass to the higher values. The exact functional characteristics of the change induced by raising the power of the distribution depends on the distribution of values, but the inequalities listed here show how the ordering of the magnitude of values is preserved by this operation (at least for non-negative powers) and how the largest powers assigning more mass to the higher values converges to a distribution that always selects the maximum value action available (because the largest winning rate(s) become larger in measure in proportion to the rest of the winning rates as the integer power of the distribution increases).

Now that we have a description of the kinds of distributions we will use in the experiments and also their measure theoretic implications as pushforward measures, exploring the relevance of invariant measures seems to be a good idea as well. I have previously mentioned that the stationary distribution of a game's underlying graph is an important lens for characterizing the results of experiments which aim to perform random walks over these graphs, particularly toward the end of the experiments. With the introduction of the relevance of pushforward measures to the theoretical treatment here, the memory update's effect on the empirical distributions has an important interpretation. The memory update creates a new pushforward measure, using the previous empirical distribution combined with new data, and this pushforward measure appreciably changes the empirical distribution. However, the conclusion of the experiment (ideally) can be best understood as the stage at which the new empirical pushforward measure is not different from the previous empirical distribution. Someone who is measure theoretically inclined will tell you that this distribution is known as an invariant measure, because it is invariant under the pushforward memory update.

Once the invariant pushforward measure is found, it is because the calculation (and generation) of the bulk limits of the winning rates of all allowable actions have converged to a stable empirical average through the aggregation of data in the MDP memory by the previous sequence of candidate mixed-strategies that generated the novel game data that was incorporated into the final pushforward invariant mixed-strategy. With the change of perspective by considering a wider set of mixed-strategies (probability distributions) versus simply considering the handful of strategies that may be yielded by implementing a deterministic hard cut off of the maximum winning rates we are able retain the focus afforded by restricting the strategy's attention to maximally rewarding actions, while also using that same focus to point the strategy exploration in the direction of a probability distribution, rather than a sequence of deterministic decisions. Under these heuristics, the algorithm is focused on a select few of the maximally valuable allowable actions, so as long as these empirical statistics are reliable/stable, our strategy seeks greedy improvement by considering empirical unilateral changes to mixed-strategies over these select few allowable actions. When the parameter settings are at 0 in the initialization phase and also still relatively small in the early to intermediary stages of the experiments, the hope is that these high entropy associated parameter settings will generate a stable and diverse foundation of winning rates to generate a candidate probability distribution which is a strong solution to its native game when a low entropy aiming, higher winning rate focused parameter setting is applied to yield a final invariant pushforward measure.

In short, we don't want the algorithm to start focusing on the corner of the state space under the canopy of the percolated sub-graph of the highest empirical winning rate edges of the game until we are sure that the winning rate statistics that generate the percolated subgraph are actually reliable. The goal is that in the beginning we want to stay approximately close to the uniform sample in order to make sure that we capture any kind of behavior that may be potentially demonstrated by any and all candidate mixed-strategies. As long as the sample of game data recorded in the memory in this initial phase is appreciable in size in comparison to the state space of the game, then the higher value allowable actions will have yielded evidence for this with their winning rate statistics and so the focus of the MDP which builds a strategy on top of a percolated sub-graph will be useful and well informed. After all, once the MDP parameter settings are stepped up high enough, the strategy will restrict its attention to a small percentage of the allowable actions, and so we want to be sure that these are actually high value winning rate allowable actions. The decrease in entropy of the mixed-strategy as a result of the higher parameter settings is an important mathematical description of the focus sought out by this strategy formulation. However, an updated, lower entropy, more aggressively win rate focused mixed-strategy should also result in better empirical performance, especially when measured in overall winning percentage. If not, it is possible, that we turned our parameter settings up too quickly and so our strategy began to perform low value actions more frequently because it mistook them for high value actions by cutting its initial phase of experimentation too short and not collecting enough information. The initial phase of experimentation with the parameter settings at 0 or small values close to 0 should be sufficiently long enough so that when the strategy parameter settings are stepped up to amplify the selection frequency of the highest value actions the updated memory/mixed-strategy has an increased expectation of a win, improving the cumulative empirical winning percentage of the strategy.

Since all of the games considered in this project are zero sum games, and because we seek to maximize the expectation of a win for the MDP strategy, the endgame scenarios penultimate to the conclusion of a game are crucially important pieces of the game data. As long as the strategy memory is robust enough to populate all of the endgame scenarios (the leaves of the graph), penultimate to the union over all final stages of the game then there is always a chance that the MDP strategy can move one of these penultimate stages more frequently than before (if it is deemed necessary). However, if the initial phase of experimentation is not robust enough to include all of these endgames, then there is no guarantee that the algorithm will be open to the consideration of all of the potential penultimate state strategies that may improve its performance.

This is a good part of the motivation for the initial phase of experimentation being sufficiently long in the context of its particular game implementation. Some sufficiently large number of game experiments is enough to ensure that enough data is collected in the initial phase of experimentation so that at the very least the MDP has memory for every penultimate stage strategy whose selection frequency may be bumped up in the consideration of a change of penultimate stage strategy and so the newfound affinity for lucrative actions in the final phase of experimentation with the higher parameter settings is actually well informed by winning rate statistics on any and all of the veritable candidate penultimate stage mixed-strategies that may trickle down to the opening and intermediary stage strategies.

However, if for some reason the final phase of experimentation fails to yield a stable convergence upward toward better performance, this is when it may be advantageous to keep running the initial phase to collect a bigger memory that has more reliable information on the canopy (sub-graph of most valuable actions and states in a given game) under which the MDP may select strategies, or this may be in a context like Chess where there is an uncountable space of allowable actions and so in theory, the first phase of experimentation never ends, and runs behind the MDP strategy in parallel, always providing novel game data that can hopefully provide new information on candidate unilateral penultimate stage strategy changes (and the actions which are close neighbors in the intermediate and opening stages).

When a penultimate stage action is updated to be performed more frequently because novel game data shows it to have a high empirical winning rate, the opening and intermediate stages of from which that penultimate stage is reachable can be improved in future experiments simply by selecting those opening and intermediate stage actions and then selecting the newly updated penultimate stage action. Of course, this is never explicitly sought out by the algorithm since it is a random process, but once a penultimate stage strategy is updated as an action with a high empirical winning rate, future experiments whose random walks pass through the opening and intermediate stages from which the penultimate action is reachable are more likely than before to make their way to the recently updated penultimate action, with its increased winning rate.

In practice, once the final phase of experimentation (highest parameter value) is reached the evolution of the algorithm is reminiscent of stochastic gradient descent, updating the winning rates of the most lucrative actions in order of their estimated winning potential and also adding these positive results to the winning rates of the actions penultimate to the final result of a win or loss all the way back to the most lucrative opening and intermediate stage actions from which the lucrative final stage game actions were reached by the strategy employing a Markov random walk obeying the rules of the game (with the hope that increases in the frequency of any particularly valuable action choice results in an increase in the expectation of a win). The algorithm has converged once the empirical winning rates of all opening, intermediate and final stage actions have arrived at a stable bulk limit after being updated to incorporate all of the novel game data that may be collected.

Note that since the frequency of a given action choice can also influence the winning rate of that action choice, we hope to increase the frequency of the most lucrative action choices, but we must also hope that this increase in frequency also results in a material increase in the expectation of a win. The action choice frequencies can be increased either by increasing the power of the distribution used as the Markov measure candidate mixed-strategy or by a novel material increase in the bulk limit of the winning rate of the given action. The final phase of experimentation most likely favors the former with the high parameter value serving to create a Markov measure assigning nearly all of its mass to a select few most valuable actions (on the other hand, it is the hope that the winning rates arrive at stable bulk limits with the initial and intermediary phases of experimentation). Although the winning rates do undergo a different mode of evolution in the later phases of experimentation, these changes are most often caused by the increased parameter settings producing sequentially lower entropy candidate mixed-strategies whose concentration of measure (in selection frequency) toward the most valuable actions resulting in a material increase in the winning rates of the opening and intermediary stage winning rates from which the lucrative endgames are reachable. As the evolution of the MDP mixed-strategy exhibits its concentration of measure toward the actions which have yielded stable bulk limits of the most lucrative winning rates, this is when an experimenter may hope to give some thought to the confirmation of a material increase in the expectation of a win for the strategy in the context of $\mathbb{E}[L(y, t)]$.

Interestingly, all sequences of actions corresponding to game data may be considered novel when 1: the sequence has not been observed before *and* 2: the actions in the sequence have not been performed by the particular mixed-strategy that selected the action(s). And so by definition, the final pushforward invariant mixed-strategy is in one-to-one correspondence with a set of statistics corresponding to a stable converged set of bulk limits of the winning rates of allowable actions. This is an important characterization of the status of convergence of the algorithm. A converged mixed-strategy cannot yield novel game data and the bulk limit of winning rates cannot be reached until all novel game data has been collected. Thus, there is an important relationship between the collection of novel game data and the construction of the Markov measures that are used to select actions in novel game data which yields the bulk limit of the winning rates of all allowable actions and the corresponding final converged mixed-strategy yielded by the algorithm. Also, the arrival of novel game data is strong evidence that the Markov measure mixed-solution will not yet exhibit Martingale behavior, for example when trying to approximate $\mathbb{E}[L(\hat{y}, t)] > 0$.

As a relation concerning increasing events in a random process the Fortuin Kasteleyn Ginibre (fkg) inequality helps explain the relevance of the indicator function to the construction of the pushforward measures. A consequence of the fkg inequality is that if X and Y are increasing events in the actualization of a random process, then $\mathbb{P}(X \cap Y) \geq \mathbb{P}(X)\mathbb{P}(Y)$. For example, if we take X to be the event that $W_{j,n_j} < W_{j,2n_j}$ and Y to be the event that the cumulative winning percentage given by the empirical distribution on endgames increases by at most 0.01 points over $N \sim (2n_j)^5$ games played with the MDP strategy, then an interpretation of the fkg inequality supports the idea that over the the course of the $N \sim (2n_j)^5$ experiments, conditioning on the sub-sequence of n_j memory updates that increase the winning percentage of a given allowable action j, increases the expectation of the event of a win.

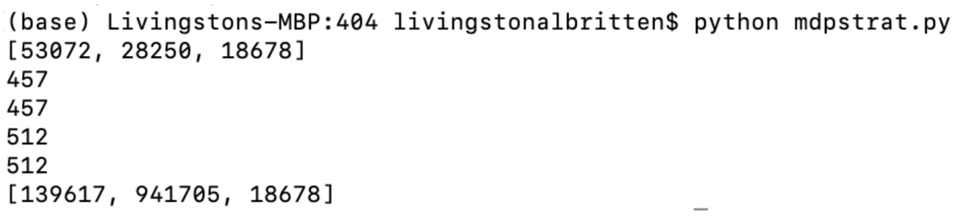
If the Markov chain is friendly enough to yield a stationary distribution after a sufficient sequence of experiments performing a random walk along the edges of the underlying graph, then a pushforward memory update from an MDP strategy which uses the probability distribution over edges to perform a random walk generating a new sequence of games to be consolidated into the memory will yield an invariant measure. Actually, the pushforward memory update may go through phases of experimentation that yield a stationary distribution on endgames, and this is not dependent on the regularity of the underlying markov chain. Anecdotally, this is primarily because the late stage parts of the underlying graph, leaves or endgames, may be populated enough by the random walk generated by the MDP strategy to yield a stationary distribution on endgames. However, once the final phase of experimentation is reached, the spanning forest graph given by the endgames should help coax the pushforward memory update into yielding a stable invariant measure. As the last move of a game always determines the outcome as a win or loss, this helps explain why the endgames are most relevant to the approximation of $\mathbb{E}[L(y,t)]$, and also the convergence to an invariant measure. So, sometimes the markov chain's stationary distribution over all vertices is relevant to the invariant measure, but it is better to primarily think about the pushforward memory updates effect on the leaves of the graph that are penultimate to the final results and potential convergence to an invariant measure.

3.2 Tic-Tac-Toe MDP Strategy Implementation

The tic-tac-toe experiments were composed of an initial 100K game burn in followed by another 1M games with O using the MDP strategy selection. This MDP strategy selection selects the maximum winning rate move twice as often as it selects a move sampled from the Markov measure of winning rates of available moves (2:1). In the first 100K games, the uniform sampling only yields a 28% winning rate for O, while that rate is raised to 91.2% for the final 1M games to yield a cumulative winning rate of 85.5% for the whole 1.1M games. While the experiment was one sided, in the sense that X was not using the MDP strategy and continues using the uniform sample strategy, the 93.1% winning rate in the 1M games at the very least shows that the MDP strategy is capable of performing well, possibly even converging to or approximating optimal solutions against an innocuous strategy like a uniform sample.

$$A_T = \{0, 1, 10^{10}\}$$

$$\hat{\mathbb{P}}_{M0,s} \sim \hat{\mathbb{P}}_s^{a_{t_0}}, \, \hat{P}_{M1,s} \sim (1/3)\hat{\mathbb{P}}_s^{a_{t_1}} + (2/3)\hat{\mathbb{P}}^*_{s,A_T^\infty}$$



```
(base) Livingstons-MBP:404 livingstonalbritten$ python mdpstrat.py
[53072, 28250, 18678]
457
457
512
512
[139617, 941705, 18678]
```

Figure 3.2: Terminal output from Tic-tac-toe experiments

3.3 Blackjack MDP Strategy Implementation

There were two blackjack experiments. Both use the same sequence of non-negative integer powers of Markov measures, with the second experiment testing the final instance of the MDP solution longer than its counterpart in the first experiment. The first was composed of an initial burn in of 100K games followed by another 200K games with the game's single player using the MDP selection strategy (80:1), followed by 100K games with the single player using the MDP strategy (25:1) and a final 600K games with the singular player using the MDP selection strategy (7:1). The second was composed of an initial burn in of 100K games followed by another 200K games with the game's single player using the MDP selection strategy (80:1), followed by 1.2M games with the single player using the MDP strategy (25:1) and a final 8.3M games with the singular player using the MDP selection strategy (7:1). The ratio's following the strategy indicate that the strategy selects a move sampled from the Markov measure generated using winning rates of available moves 80 times more often as it selects the maximum winning rate move in the first 200K following burn in (for example).

In experiment 1, in the first 100K games, the blackjack strategy yields a 35.9% winning rate, while in the middle 200K, the winning rate improves to 47.2%, and in the final 700K the winning rate settles to 46.9% to yield a cumulative winning rate of 45.9%. Below is a description of the sequence of non-negative integer powers used to generate the Markov measures as well as closed form representations of the Markov measures that are sampled by the instances of the MDP strategies tested in the experiments.

$$A_T = \{0, 13, 10^{10}\}$$

$$\hat{\mathbb{P}}_{M0,s} \sim \hat{\mathbb{P}_s}^{a_{t_0}}, \, \hat{P}_{M1,s} \sim (80/81)\hat{\mathbb{P}}_s^{a_{t_1}} + (1/81)\hat{\mathbb{P}}^*_{s,A_T^\infty}$$

$$\hat{\mathbb{P}}_{M2,s} \sim (25/26)\hat{\mathbb{P}}_s^{a_{t_1}} + (1/26)\hat{\mathbb{P}}^*_{s,A_T^\infty}, \, \hat{P}_{M3,s} \sim (7/8)\hat{\mathbb{P}}_s^{a_{t_1}} + (1/8)\hat{\mathbb{P}}^*_{s,A_T^\infty}$$

In experiment 2, in the first 100K games, the blackjack strategy yields a 35.4% winning rate, while in the middle 200K, the winning rate improves to 47.1%, and in the final 9.5M the winning rate holds at 47.1% to yield a cumulative winning rate of 45.9%.

The winning rates yielded by the two experiments do provide some good information about the convergence of the MDP solution in the context of blackjack. The winning rate over the final 700K and final 9.5M games of experiment 1 and 2 respectively are comparable, both within 0.1% of 47.0%. The final winning rates of show that the quality of the convergence is attractive enough to consider the relevance of the invariant measure. We may not be able to guarantee that the final MDP solution is optimal, but at least we can confirm that the convergence in the final 9M games is consistent enough to yield a solution whose convergence to an invariant measure can be confirmed over a sufficiently lengthy sequence of games.

Figure 3.3 provides information on the relative frequency of winning actions chosen by the MDP strategy in the blackjack experiments. The card combinations with hand values of 20 or 21 are the most lucrative, as the actions that correspond to staying on these values are responsible for $> 40\%$ of the algorithm's wins. Additionally, 88% of the algorithm's wins are captured by 15% of the winning moves available to the strategy and $> 98\%$ of the algorithm's wins are captured by 70% of the winning moves available to the strategy. The top heavy distribution of the algorithm's selection of winning moves helps explain the importance of raising the empirical distribution to an integer power, by increasing the odds of selecting the higher winning rate moves, we are able to drive our algorithm to convergence toward a distribution which favors the most lucrative actions.

Figures 3.4 and 3.5 give summaries of the blackjack experiments that test the MDP strategy in two different blackjack settings (the winning percentages reported above refer to the first setting). In the first setting, the payout for a natural blackjack is 150% of the player's original bet, while in the second setting the payout for a natural blackjack is 200% of the player's original bet. A natural blackjack is any hand where the first two cards result in a hand value of 21 with one card being an ace and the other having a value of 10. Importantly, the increased payout in the second setting is enough to boost the cumulative value of the MDP strategy's total winnings above that of the dealer.

Figures 3.6 and 3.7 give some information on the statistics and strategic rules that the MDP strategy and Thorpe's Beat The Dealer card counting strategy follow. In Figure 3.6, Thorpe's card counting strategy keeps track of the ratio of others/tens in the deck to make decisions dependent on the number of tens in the deck. The black squares indicate that the strategy will split the pair, the white squares indicate that the strategy will not split the pair and the squares with numbers give thresholds on the others/tens ratio for acceptable when to split the pair. Since the MDP strategy does not consider the dealer's face up card or the composition of cards in the deck, it is best to consider the polarity of the histograms in Figure 3.7 for different hands in the MDP strategy. When the polarity of the histogram favors splitting the pair, we know that the 10th power of the distribution will split the pair fairly often; these are the pairs that we should expect Thorpe's strategy to implement hard splits on for most of the dealer's prospective face up cards (the {07, 07} pair, or the {08, 08} pair). Conversely, if the histogram favors hitting or staying on the pair, we know that the 13th power of the distribution will favor selecting one of these actions rather than the split, and we should expect Thorpe's strategy to avoid advocating for a hard split on any of the dealer's prospective face up cards (the {04, 04} pair, the {05, 05} pair or the {JK, JK} pair).

Notably, the comparison between the MDP strategy evaluated here and the others/tens strategy considered by Thorpe should be given a disclaimer: it should be very hard to quantify the true distance between the strategies, for there may not be a space that effectively houses them both. In the games considered in Thorpe's book, players are allowed to look at the dealer's face up card, and the use of the others/tens ratio means that the deck is not shuffled after every game. In the experiments here, however, neither are the case, the player only considers his own hand when evaluating his strategy selection, and the deck is shuffled after every game $> 70 - 80\%$ of the time, so the players in Thorpe's book are supplied with more information than the automated MDP players in my own blackjack experiments. Of course, the goal is that everyone is able to maximize the utility of all of the information available to them, and so we must consider that players in Thorpe's book and the automated MDP players are motivated by different interests, but I only meant to say that despite their differences of manifest variability, there are situations, like the hand values considered above, where the quantitative features of the MDP strategy can be interpreted as qualitative direction, or attitude that may be approximately mirrored by players in Thorpe's book.

Figure 3.8 is an image of a jupyter environment cell implementing the functionality to perform blackjack experiments in Python code. Fortunately, because the code handles the player's strategy queries first (according to the rules of blackjack) if a multi player version of the experiment was implemented, each player could be assigned an order and queried in succession before the dealer is queried for his strategy selection. For the sake of this experiment, the games only involve the dealer and one player. But if players were to be added in future experiments it could be done in a fashion described above. It should be noted that if our one player experiment is capable of approximating an optimal solution (or even simply converging to a comprehensible solution), then the multi player experiments with each player employing the MDP strategy should yield a similar set of solutions for every player in the multi player setting. After all, the dealer's hand is the only other hand considered when evaluating a player's performance as a win or a loss.

The (card combination, action) pairs referenced in the terminal output give information on the size of the memory after each phase of experimentation. When the MDP strategy is queried at each stage of a game in blackjack, it takes the card combination and pairs it with each allowable action associated to the hand value. The MDP memory's recorded winning percentages for each allowable action that can be attached to a given card combination to form a (card combination, allowable action) pair are the values used to generate the Markov measure that the MDP samples to select actions.

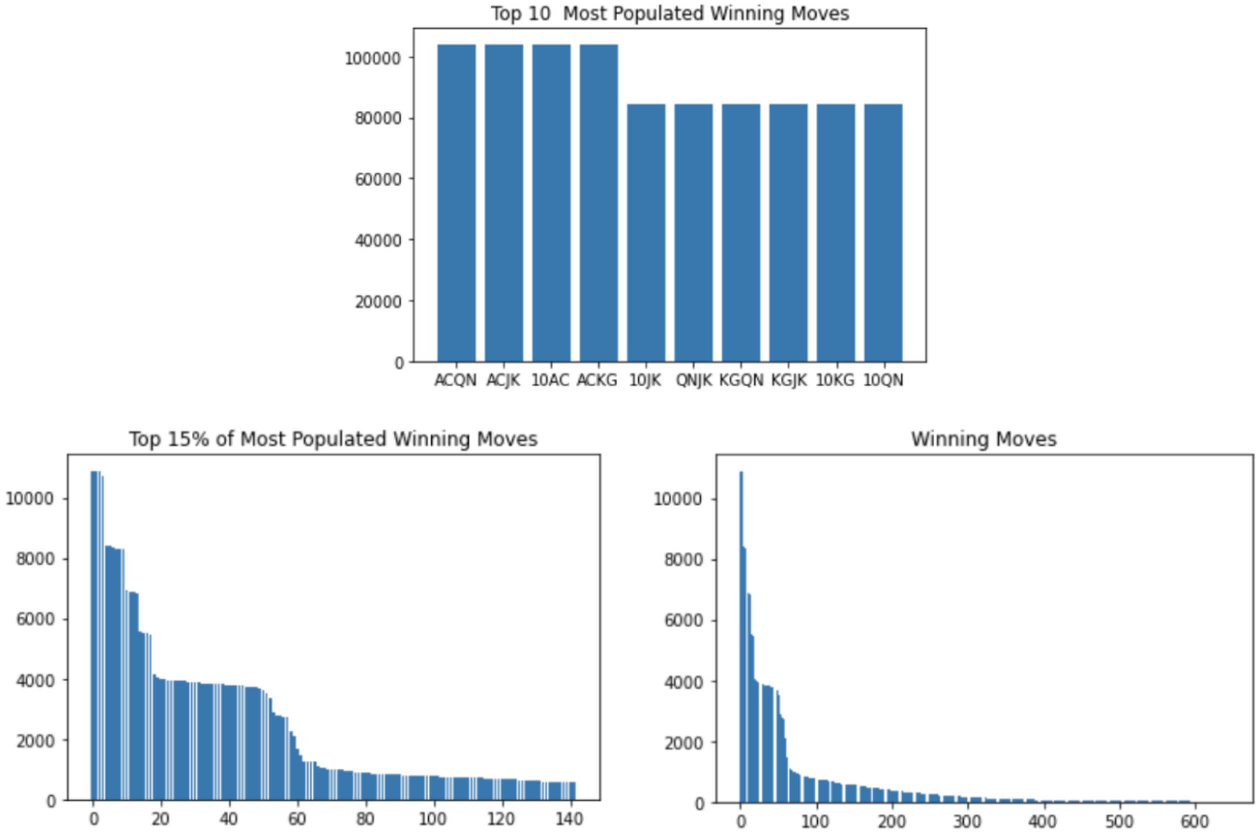
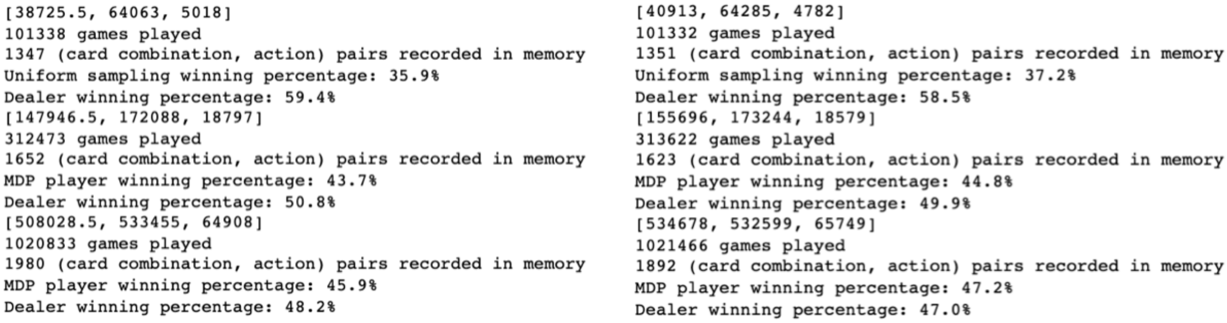


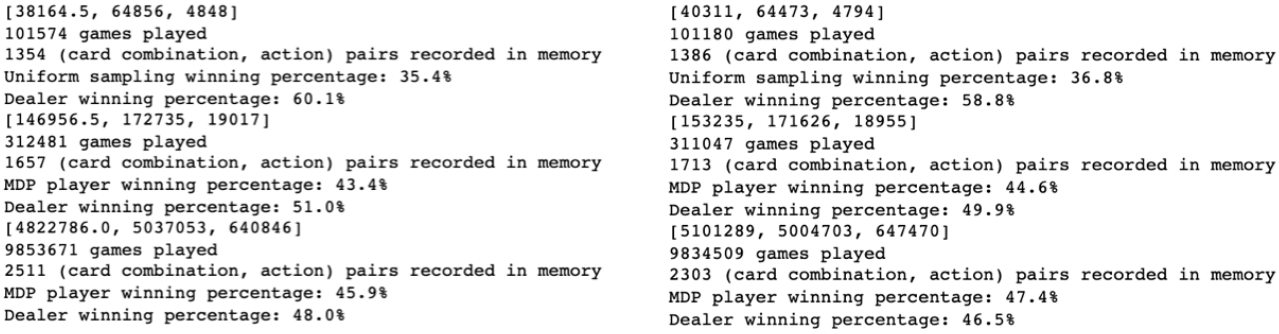
Figure 3.3: Histograms of the top 10 most populated winning moves, the top 15% most populated winning moves after 1M blackjack experiments, and the full distribution of winning moves played by the MDP strategy (each move corresponds to a {card combination, allowable action} pair)



```
[38725.5, 64063, 5018]
101338 games played
1347 (card combination, action) pairs recorded in memory
Uniform sampling winning percentage: 35.9%
Dealer winning percentage: 59.4%
[147946.5, 172088, 18797]
312473 games played
1652 (card combination, action) pairs recorded in memory
MDP player winning percentage: 43.7%
Dealer winning percentage: 50.8%
[508028.5, 533455, 64908]
1020833 games played
1980 (card combination, action) pairs recorded in memory
MDP player winning percentage: 45.9%
Dealer winning percentage: 48.2%
```

```
[40913, 64285, 4782]
101332 games played
1351 (card combination, action) pairs recorded in memory
Uniform sampling winning percentage: 37.2%
Dealer winning percentage: 58.5%
[155696, 173244, 18579]
313622 games played
1623 (card combination, action) pairs recorded in memory
MDP player winning percentage: 44.8%
Dealer winning percentage: 49.9%
[534678, 532599, 65749]
1021466 games played
1892 (card combination, action) pairs recorded in memory
MDP player winning percentage: 47.2%
Dealer winning percentage: 47.0%
```

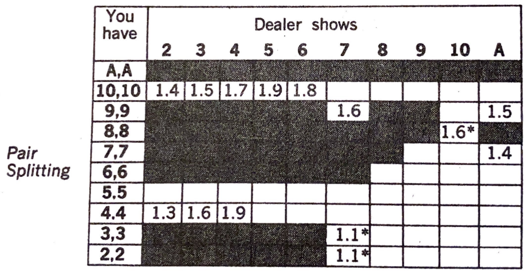
Figure 3.4: Terminal output from first 1.0M Blackjack experiments



```
[38164.5, 64856, 4848]
101574 games played
1354 (card combination, action) pairs recorded in memory
Uniform sampling winning percentage: 35.4%
Dealer winning percentage: 60.1%
[146956.5, 172735, 19017]
312481 games played
1657 (card combination, action) pairs recorded in memory
MDP player winning percentage: 43.4%
Dealer winning percentage: 51.0%
[4822786.0, 5037053, 640846]
9853671 games played
2511 (card combination, action) pairs recorded in memory
MDP player winning percentage: 45.9%
Dealer winning percentage: 48.0%
```

```
[40311, 64473, 4794]
101180 games played
1386 (card combination, action) pairs recorded in memory
Uniform sampling winning percentage: 36.8%
Dealer winning percentage: 58.8%
[153235, 171626, 18955]
311047 games played
1713 (card combination, action) pairs recorded in memory
MDP player winning percentage: 44.6%
Dealer winning percentage: 49.9%
[5101289, 5004703, 647470]
9834509 games played
2303 (card combination, action) pairs recorded in memory
MDP player winning percentage: 47.4%
Dealer winning percentage: 46.5%
```

Figure 3.5: Terminal output from 9.8M Blackjack experiments



	You have	Dealer shows									
		2	3	4	5	6	7	8	9	10	A
Pair Splitting	A,A	■	■	■	■	■	■	■	■	■	■
	10,10	1.4	1.5	1.7	1.9	1.8					
	9,9	■	■	■	■	■	1.6	■	■		1.5
	8,8	■	■	■	■	■	■	■	■	1.6*	■
	7,7	■	■	■	■	■	■	■			1.4
	6,6	■	■	■	■	■	■				
	5,5										
	4,4	1.3	1.6	1.9							
	3,3	■	■	■	■	■	1.1*				
	2,2	■	■	■	■	■	1.1*				

Figure 3.6: Table representing strategy described in Thorpe's Beat the Dealer

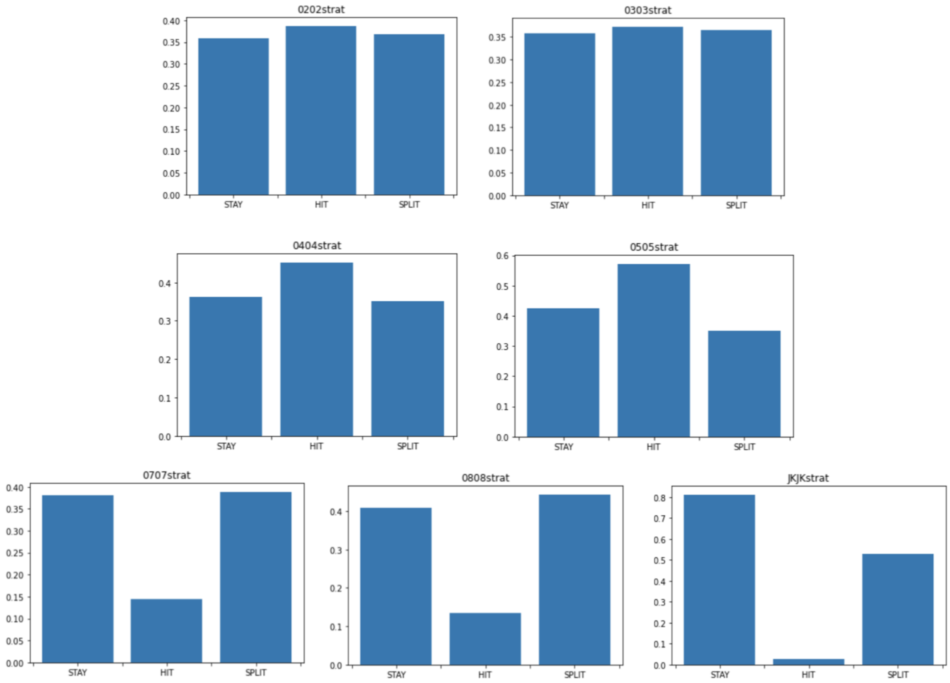
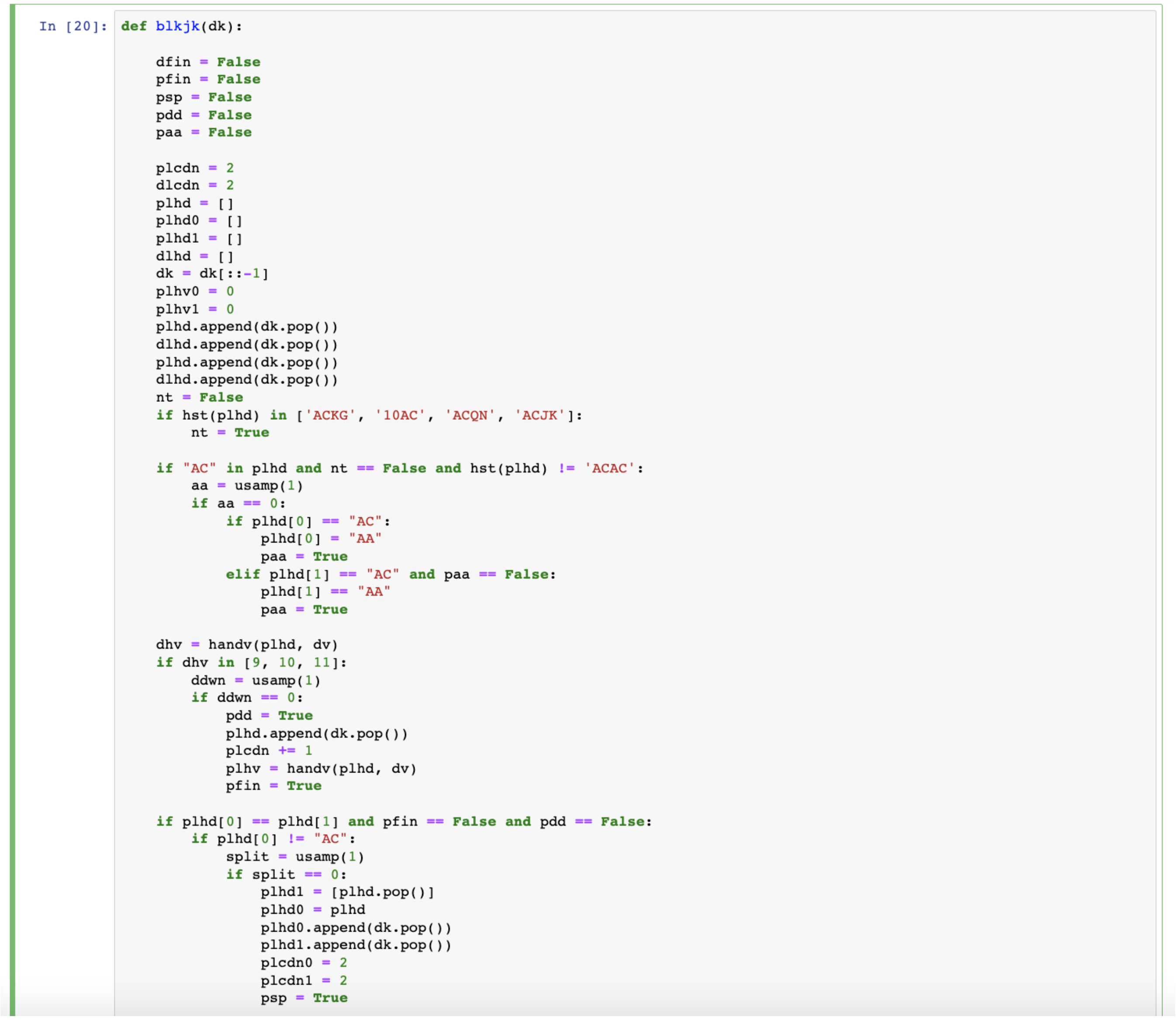


Figure 3.7: Histograms of the empirical distribution over allowable actions in blackjack for various pairs of cards. Note that each histogram entry corresponds to a (card combination, allowable action) pair winning percentage $W_{cc,aa}$

3.4 Chess MDP Strategy Implementation

The implementation of the MDP Strategy in Chess again requires code functionality that provides the ability to sample the allowable moves. Additionally, the chess implementation was assembled with code functionality that allows for connections to a SQL database. There were 400K games played in the first stage of experimentation, with both sides using the uniform sampling procedure to select moves. In these experiments, with castling and promotion enabled, black wins $> 49.9\%$ of games and white wins $< 50.1\%$ of games.



```python
def blkjk(dk):

    dfin = False
    pfin = False
    psp = False
    pdd = False
    paa = False

    plcdn = 2
    dlcdn = 2
    plhd = []
    plhd0 = []
    plhd1 = []
    dlhd = []
    dk = dk[::-1]
    plhv0 = 0
    plhv1 = 0
    plhd.append(dk.pop())
    dlhd.append(dk.pop())
    plhd.append(dk.pop())
    dlhd.append(dk.pop())
    nt = False
    if hst(plhd) in ['ACKG', '10AC', 'ACQN', 'ACJK']:
        nt = True

    if "AC" in plhd and nt == False and hst(plhd) != 'ACAC':
        aa = usamp(1)
        if aa == 0:
            if plhd[0] == "AC":
                plhd[0] = "AA"
                paa = True
            elif plhd[1] == "AC" and paa == False:
                plhd[1] == "AA"
                paa = True

    dhv = handv(plhd, dv)
    if dhv in [9, 10, 11]:
        ddwn = usamp(1)
        if ddwn == 0:
            pdd = True
            plhd.append(dk.pop())
            plcdn += 1
            plhv = handv(plhd, dv)
            pfin = True

    if plhd[0] == plhd[1] and pfin == False and pdd == False:
        if plhd[0] != "AC":
            split = usamp(1)
            if split == 0:
                plhd1 = [plhd.pop()]
                plhd0 = plhd
                plhd0.append(dk.pop())
                plhd1.append(dk.pop())
                plcdn0 = 2
                plcdn1 = 2
                psp = True
```

Figure 3.8: Python code implementing the uniform sample strategy for the player in the initial burn in stage of the MDP memory consolidation

CHAPTER 4

Potential Extensions And Conclusion

Potential extensions for work done after this point may incorporate connections to the Bernoulli shift isomorphisms native to Ornstein theory. The associated algebraic analysis of Bernoulli shift isomorphisms provide a new language for understanding the relations between the processes that actualize each different Markov process as well as alternative ways to interface with the full state space of the data beyond the algorithmic compression via strategy formulation. Also analyses associated with the Diffusion Wavelet decomposition may help demonstrate the concentration of measure from the uniform distribution to the unique MDP Markov measure.

Looking forward, advanced algebraic analyses of the composition of the graphically based game theoretic solutions presented in this project will provide a better understanding of the actualization of the random processes that correspond to a given solution. For example, the uniformity over deck arrangements supports a more rigorous consideration of the strategies implemented in the blackjack experiments as it confirms that each new card drawn from the deck is essentially independent of the last (as long as the deck is shuffled frequently enough between games). In the publication "Trailing the Dovetail Shuffle to it's Lair" authors Bayer and Diaconis invoke Markov Chains on the symmetric group to perform their analysis of the dovetail shuffle's convergence to the uniform distribution over all deck arrangements. Their analysis focuses on the evolution of the complexion of rising sequences in the deck over the course of a series of riffle shuffles. The convergence to the uniform measure can also be evaluated by observing the empirical distribution of the size of the cycles in the permutations that correspond to the transformation of the deck before and after the bridge shuffle used in the blackjack experiments in this project. At the very least this demonstrates that more advanced algebraic analyses of the data/games may provide either faster algorithmic convergence, stronger algorithmic performance, or both and so potential extensions and conclusions that incorporate explanations of and references to these analyses that contextualize the observed algorithmic performance of the MDP are worthwhile.

REFERENCES

[1] Von Neumann, J. and Morgenstern, O. (1953) *Theory of Games and Economic Behavior*, Princeton University Press.

[2] Mazumdar, E., Dong, R., Royo, V. R., Tomlin, C., and Sastry, S. S. (2017), *A Multi-Armed Bandit Approach for Online Expert Selection in Markov Decision Processes*, arXiv preprint arXiv:1707.05714

[3] Wasserman, L. (2006) *All of Statistics*, Springer.

[4] Dudley, R.M. (2004) *Real Analysis and Probability*, Cambridge University Press.

[5] Wikipedia, *Pushforward measures*, available at https://en.wikipedia.org/wiki/Pushforward_measure?oldformat=true

[6] Caltech, *Caltech CS150, FKG Inequality*, available at http://users.cms.caltech.edu/∼schulman/Courses/18cs150/lec7.pdf

[7] Thorpe, E.(1966). *Beat The Dealer*, Vintage Books.

[8] Bremer, J. C., Coifman, R. R., Maggioni, M., and Szlam, A. D. (2006). *Diffusion Wavelets*, Applied and Computational Harmonic Analysis, 21(1), 95-112.

[9] Bayer, D. and Diaconis, P. (1992)., *Trailing the Dovetail Shuffle to its Lair*, The Annals of Applied Probability, 2, 294-313.

[10] UC Riverside, *Random Permutations*, available at https://math.ucr.edu/home/baez/permutations/permutations_4.html