**Parabola Volume 56, Issue 3 (2020)**

# Brute Force Learning in Sequential Games

**Eric Gao**[1]

## 1 Introduction

Much work has been done in investigating games of imperfect or incomplete information. This paper analyzes a different conceptualization of not fully knowing how a game operates, where the player knows possible payoffs and their possible moves but does not know the mechanism of the game. We then analyze how learning may occur wherein the player formulates an idea of how the game mechanism assigns payoffs based on the player's move without actually knowing more about the mechanism of how the game works. Thus, instead of acquiring more information about the mechanism of the game, an actor only gets better at guessing their way through each repetition based on their knowledge of prior results.

We now look at related work in the existing literature. The game in consideration in this paper is similar to a multi-armed bandit problem,[2] except that each of the results is deterministic ($W$in or $L$ose) instead of a probability to win. However, the game under consideration requires the player to make multiple decisions before receiving any payoff, motivating the approach used in this paper.

Sandomirskiy [8] considers repeated zero sum games with one player only having incomplete information. Before play starts, the type of zero sum game is selected by chance from a class of almost-fair games with the game chosen only known by player one. Under certain conditions, the total payoff to player two for a large number of repetitions $n$ is directly proportional to $\sqrt{n}$, while under other conditions of the games considered, the payoff is asymptotically constrained instead. Although the game model utilized in this paper is different, the result that learning progression converges to a certain payoff holds.

Valluri [9] applies a similar reinforcement-based learning algorithm to a two player sequential prisoners' dilemma game and finds that in simulations, agents eventually learn to cooperate. Similarly, Littman [5] shows convergence to Nash equilibrium given conditions of existence as well as assumptions about the type of play an opponent uses (friend or foe). While the exact situations are slightly different, both aforementioned papers involve an agent getting better at playing given an uncertain payoff function. The learning mechanism proposed in this paper, however, is more direct and does not rely on remembering states or opponents' previous moves. For an overall

[1] Eric Gao is a rising Senior at Northwood High School in Irvine, California (eric.boxuan.gao@gmail.com).

[2] For a review of work done on the multi-armed bandit problem, see Mahajan and Teneketzis [6].

review on reinforcement learning and its applications in game theory, see Chapter 14 of Now, Vrancx, and De Hauwere [7].

Finally, behavioral models of learning in games similarly focus on two-player games and how an agent responds to the actions of other player(s) instead of to the game mechanism itself.[3] Experimentally, while two player games do not always converge to Nash equilibria, studies such as Van Huyck, Battalio, and Beil [4] have shown that as play goes on through multiple iterations of the same game, agents get better and better at predicting game dynamics.

Slightly different from the rich literature concerning models of learning in games, this paper shows how even a computationally simple mechanism can result in convergence given a sufficiently straightforward game.

# 2 A Basic Game

## 2.1 Model

We consider a single player, three turn game. In the first and second turns, the player has three choices of what to move. From the player's point of view, they do not know what happens after each move due to not knowing the mechanism of how the game works, but the player can still differentiate between the moves. To the player, the game propagates as such:

$$move \rightarrow some\ indeterminate\ thing\ the\ game\ mechanism\ does \rightarrow result\,,$$

with the player being unsure of whether or not the result is uniquely determined by their move or by the game. However, observers know that of the three moves, one move leads to the player winning ($W$), one move leads to them losing ($L$), and the final move leads to the game going on to the next decision ($C$). If the game reaches the third turn, the player then has two choices (once again, the player can differentiate between the two moves but does not know the outcome of the moves), one leading to a win and the other to a loss.

As the player does not know the outcome of their choices in the beginning, they have no choice but to guess and play randomly, making adjustments to their strategy as they repeatedly play the game. Then, the probability the player obtains a particular result from the move they chose can thus be denoted as:

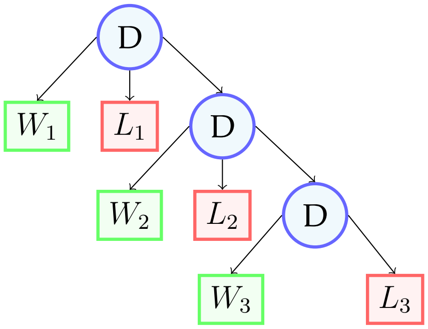
$$X_i^r \in [0,1]$$

where $X \in \{W, L, C\}$ denotes the outcome of the player's move, $r$ denotes the $r$th round the player is currently playing, and $i \in \{1,2,3\}$ the $i$th move in the current round. When $i = 3$, $X \in \{W, L\}$ as there is no continuation of the game.

[3] For a survey of both passive and active learning, see Fudenberg and Levine [2].

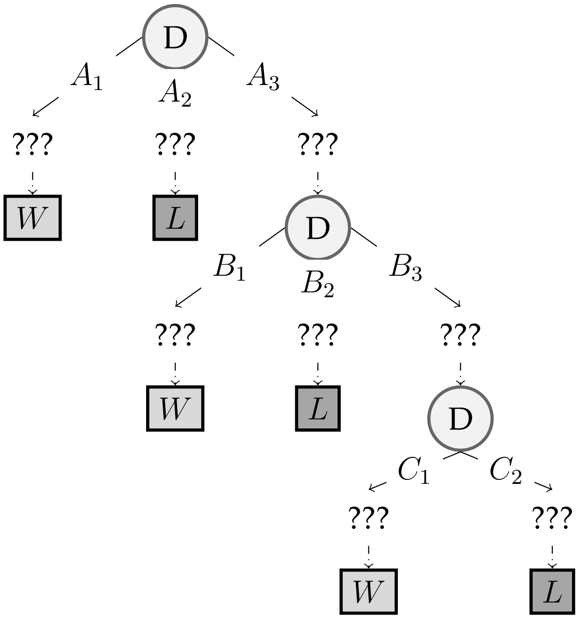
This game can also be represented as the following directed graph for someone who has perfect information about how the game works, where $D$ represents an opportunity for the player to make a decision:

Figure 1: Perfect Information Point of View



However, to the player who does not have perfect information, the game appears as such:

Figure 2: Player's Point of View



The probability of winning in round $r$ can then be expressed as:

$$P(r) = W_1^r + C_1^r[W_2^r + C_2^r W_3^r]\,. \tag{1}$$

Players "learn" through successive iterations of this game due to them knowing the outcome of previous attempts, even if they are oblivious to the mechanism that dictates the result of the game. Let $n \in [0, 1]$ be a real number representing the player's willingness to learn and adapt, where smaller $n$ means a higher propensity to change their move based on previous results, and vice versa for higher $n$.

The value $n = 1$ trivially results in no learning happening; going forward, we will assume that $n \in [0, 1)$ instead. In the case $n = 0$ (perfect learning), once a player wins, they would keep playing the same strategy over and over when repeatedly playing the game. However, it becomes much more interesting in the imperfect case with $n > 0$.

Now, we consider the effects of learning based on the current iteration of the game on the next iteration of the game by analyzing the change in the probabilities to win (1). When a player wins ($W_1^r, W_2^r, W_3^r$), they play the same move with a higher probability when they play the next iteration of the game, which also results in a decrease in the probabilities of the other moves. If a player loses in the current game ($L_1^r, L_2^r, L_3^r$), they play the move immediately before the loss with a lower probability, which results in an increase in the probabilities of the other moves. If a player's move results in the game continuing ($C_1^r, C_2^r$), then nothing changes to probabilities at that decision node.[4] Due to there being multiple possible outcomes of the current game, what happens in the following iteration of the game is thus broken up into multiple cases. Furthermore, due to the decision space being different between the first two moves and the third move, those two cases need to be treated separately as well.

**Moves 1 and 2** ($i \in 1, 2$)

$$X_i^r = W: \quad \begin{aligned} L_i^{r+1} &= nL_i^r = L_i^r + (n-1)L_i^r \\ C_i^{r+1} &= nC_i^r = C_i^r + (n-1)C_i^r \\ W_i^{r+1} &= W_i^r + (1-n)(1-W_i^r) \end{aligned}$$

$$X_i^r = L: \quad \begin{aligned} L_i^{r+1} &= nL_i^r = L_i^r + (n-1)L_i^r \\ C_i^{r+1} &= C_i^r + \tfrac{C_i^r}{1-L_i^r}(1-n)(L_i^r) \\ W_i^{r+1} &= W_i^r + \tfrac{W_i^r}{1-L_i^r}(1-n)(L_i^r) \end{aligned}$$

$$X_i^r = C: \quad \begin{aligned} W_i^{r+1} &= W_i^r \\ L_i^{r+1} &= L_i^r \\ C_i^{r+1} &= C_i^r \end{aligned}$$

**Move 3** ($i = 3$)

$$X_i^r = L: \quad \begin{aligned} L_i^{r+1} &= nL_i^r \\ W_i^{r+1} &= W_i^r + (1-n)(1-W_i^r) \end{aligned}$$

$$X_i^r = W: \quad \begin{aligned} L_i^{r+1} &= nL_i^r \\ W_i^{r+1} &= W_i^r + (1-n)(1-W_i^r) \end{aligned}$$

[4] However, note that there inevitably will be either a $W$ or $L$ move played; continuing the game just results in the change of move probabilities occurring at a later stage.

## 2.2 Preliminary Results

We now derive two results from our game and learning process.

**Theorem 1** (Eventual Perfection)**.** *Brute force learning based on the model defined in Section 2 eventually converges to perfect play for all $n \in [0,1)$ and $X_i^1 > 0$ for all $X \in \{W, C, L\}$ and $i \in \{1,2,3\}$. That is, $\lim\limits_{r\to\infty} \mathbb{E}(P(r)) = 1$ where $\mathbb{E}$ denotes expected value.*

**Lemma 2.** *$P(r)$ is strictly increasing in $r$.*

*Proof.* Note that for each $i$ the sequence $X_i^r$ is discrete, only defined when $r \in \mathbb{Z}^+$. We first reconfigure the sequence to make $X_i^r$ a continuous function. Observe how $X_i^{r+1}$ takes the form $X_i^r + \Delta(X)$ for each $X$, where $\Delta(X) = X_i^{r+1} - X_i^r$. Now, define $X_i^{r+\epsilon} = X_i^r + \epsilon\Delta(X)$ where $\epsilon \in [0,1]$. This function "connects the dots" between each discrete value of $r$: $X_i^{r+\epsilon} = X_i^r$ when $\epsilon = 0$ and $X_i^{r+\epsilon} = X_i^{r+1}$ when $\epsilon = 1$.

This new function is continuous in all $r \in \mathbb{R}^+$ and is differentiable when $r \notin \mathbb{Z}^+$. The derivative of $X_i^r$ is then:

$$\frac{dX_i^r}{dr} = \lim_{\epsilon\to 0}\frac{X_i^{r+\epsilon} - X_i^r}{\epsilon} = \lim_{\epsilon\to 0}\frac{X_i^r + \epsilon\Delta(X) - X_i^r}{\epsilon} = \Delta(X)$$

for each $r \notin \mathbb{Z}^+$. We then have:

$$\begin{aligned}\frac{dP(r)}{dr} &= \frac{d}{dr}\Big(W_1^r + C_1^r\big(W_2^r + C_2^r(W_3^r)\big)\Big)\\ &= \frac{dW_1^r}{dr} + \big(W_2^r + C_2^r(W_3^r)\big)\frac{dC_1^r}{dr} + C_1^r\Big(\frac{dW_2^r}{dr} + C_2^r\frac{dW_3^r}{dr} + W_3^r\frac{dC_2^r}{r}\Big)\\ &= \frac{dW_1^r}{dr} + (W_2^r + C_2^r W_3^r)\frac{dC_1^r}{dr} + C_1^r\frac{dW_2^r}{dr} + C_1^r C_2^r\frac{dW_3^r}{dr} + C_1^r W_3^r\frac{dC_2^r}{dr}\,. \qquad (2)\end{aligned}$$

Next, we consider each of the six different paths the game can take

$$W_1\,,\; L_1\,,\; C_1 \to W_2\,,\; C_1 \to L_2\,,\; C_1 \to C_2 \to W_3\,,\; C_1 \to C_2 \to L_3$$

and apply (2) and show that, in each case, $\frac{dP(r)}{dr} \geq 0$. Before starting to work through each case, take note that

$$\begin{aligned}1 &= W_1^r + L_1^r + C_1^r\big(W_2^r + L_2^r + C_2^r(W_3^r + L_3^r)\big)\\ &= W_1^r + L_1^r + C_1^r W_2^r + C_1^r L_2^r + C_1^r C_2^r W_3^r + C_1^r C_2^r L_3^r\,,\end{aligned}$$

Thus, the sum of any proper subset of terms on the far right hand side must be less than 1, from our assumption that $X_i^1 > 0$ for all $X \in \{W, L, C\}$.

**Case 1:** $W_1$

$$\begin{aligned}\frac{dP(r)}{dr} &= \frac{dW_1^r}{dr} + (W_2^r + C_2^r W_3^r)\frac{dC_1^r}{dr} + C_1^r\frac{dW_2^r}{dr} + C_1^r C_2^r\frac{dW_3^r}{dr} + C_1^r W_3^r\frac{dC_2^r}{dr} \\ &= (1-n)(1-W_1^r) + (W_2^r + C_2^r W_3^r)(n-1)C_1^r + C_1^r(0) + C_1^r C_2^r(0) + C_1^r W_3^r(0) \\ &= (1-n)\big(1-(W_1^r + C_1^r W_2^r + C_1^r C_2^r W_3^r)\big)\,.\end{aligned}$$

From our previous observation, $W_1^r + C_1^r W_2^r + C_1^r C_2^r W_3^r < 1$. Also, $1-n > 0$, so $1-(W_1^r + C_1^r W_2^r + C_1^r C_2^r W_3^r) > 0$; therefore, $\frac{dP(r)}{dr} > 0$ in this case.

**Case 2**: $L_1$

$$\begin{aligned}\frac{dP(r)}{dr} &= \frac{dW_1^r}{dr} + (W_2^r + C_2^r W_3^r)\frac{dC_1^r}{dr} + C_1^r\frac{dW_2^r}{dr} + C_1^r C_2^r\frac{dW_3^r}{dr} + C_1^r W_3^r\frac{dC_2^r}{dr} \\ &= \frac{W_1^r}{1-L_1^r}(1-n)L_1^r + (W_2^r + C_2^r W_3^r)\frac{C_1^r}{1-L_1^r}(1-n)L_1^r + 0 + 0 + 0 \\ &= \frac{(1-n)L_1^r}{1-L_1^r}(W_1^r + C_1^r W_2^r + C_1^r C_2^r W_3^r) \geq 0\end{aligned}$$

as $n \in [0,1]$ and $L_1^r \in (0,1)$.

**Case 3:** $C_1 \to W_2$

$$\begin{aligned}\frac{dP(r)}{dr} &= \frac{dW_1^r}{dr} + (W_2^r + C_2^r W_3^r)\frac{dC_1^r}{dr} + C_1^r\frac{dW_2^r}{dr} + C_1^r C_2^r\frac{dW_3^r}{dr} + C_1^r W_3^r\frac{dC_2^r}{dr} \\ &= 0 + (W_2^r + C_2^r W_2^r)\times 0 + C_1^r(1-n)(1-W_2^r) + 0 + C_1^r W_3^r(n-1)C_2^r \\ &= C_1^r(1-n)\big(1-(W_2^r + C_2^r W_3^r)\big) > 0\,.\end{aligned}$$

**Case 4:** $C_1 \to L_2$

$$\begin{aligned}\frac{dP(r)}{dr} &= \frac{dW_1^r}{dr} + (W_2^r + C_2^r W_3^r)\frac{dC_1^r}{dr} + C_1^r\frac{dW_2^r}{dr} + C_1^r C_2^r\frac{dW_3^r}{dr} + C_1^r W_3^r\frac{dC_2^r}{dr} \\ &= 0 + (W_2^r + C_2^r W_3^r)\times 0 + C_1(\frac{W_2^r}{1-L_2^r}(1-n)L_2^r) + 0 + C_1^r W_3^r\frac{C_2^r}{1-L_2^r}(1-n)L_2^r \\ &= \frac{C_1^r L_2^r(1-n)}{1-L_2}(W_2^r + C_2^r W_3^r) > 0\,.\end{aligned}$$

**Cases 5–6:** $C_1 \to C_2 \to W_3$ or $L_3$[5]

$$\begin{aligned}\frac{dP(r)}{dr} &= \frac{dW_1^r}{dr} + (W_2^r + C_2^r W_3^r)\frac{dC_1^r}{dr} + C_1^r\frac{dW_2^r}{dr} + C_1^r C_2^r\frac{dW_3^r}{dr} + C_1^r W_3^r\frac{dC_2^r}{dr} \\ &= 0 + (W_1^r + C_2^r W_3^r)\times 0 + 0 + C_1^r C_2^r(1-n)(1-W_3^r) + 0 \\ &= C_1^r C_2^r(1-n)(1-W_3^r) > 0\,.\end{aligned}$$

□

[5] We combine these two cases due to the change occurring in step three of the game being the same regardless of win or loss.

Now, given the appropriate tools, Theorem 1, the type of brute force learning identified in this paper eventually reaches perfect play, can finally be proven.

*Proof.* If an action on move $1$ of round $r$ results in a win, then call that action a move of type $I$. Let $J_r$ be the subset of the first $r$ repetitions of the game such that a type $I$ move is played for all $i \in J_r$.

Next, if there are $j$ moves of type $I$ in the first $r$ repetitions of the game, then denote that to be event $E^r_j$.

Then, define two random variables on the events $E^r_0, E^r_1, E^r_2, \dots, E^r_r$:

- $W^{r+1}_1(E^r_j)$ is the probability the player plays the move $W^{r+1}_1$ after $r$ repetitions of the game with $j$ moves of type $I$ in the first $r$ rounds.
- $\Upsilon^r$ is the number of moves of type $I$ in the first $r$ rounds. Note that $\Upsilon^r(E^r_j) = j$.

Finally, let $\Phi^r$ be the binomial random variable with $r$ trials and $W^1_1$ probability of success in each trial. Then,

$$E(\Upsilon^r) = \sum_{j=1}^{r} jP(j = |J_r|) \geq \sum_{j=1}^{r} j\binom{r}{j} W^1_1 = E(\Phi^r) = rW^1_1$$

as $W^r_1$ is strictly non-decreasing. Also,

$$\begin{aligned} W^{r+1}_1 &\geq W^1_1 + \sum_{i \in J_r}(1-n)(1-W^i_1) \\ &\geq W^1_1 + \sum_{i \in J_r}(1-n)(1-W^r_1) \\ &= W^1_1 + (1-n)(1-W^r_1)\sum_{i \in J_r} 1 \end{aligned}$$

as $1 - W^i_1 \geq 1 - W^r_1$ for all $i$ and since $(1-n)(1-W^r_1)$ is independent of $i$. Taking the expected value of both sides yields

$$\begin{aligned} \mathbb{E}(W^{r+1}_1) &\geq \mathbb{E}\big(W^1_1 + (1-n)(1-W^r_1)|J_r|\big) \\ &= W^1_1 + (1-n)\big(1-\mathbb{E}(W^r_1)\big)\mathbb{E}(\Upsilon^r) \\ &\geq W^1_1 + (1-n)\big(1-\mathbb{E}(W^r_1)\big)rW^1_1 \, . \end{aligned}$$

Since $\mathbb{E}(W^r_1)$ is a non-decreasing sequence bounded above, $\lim\limits_{r\to\infty} \mathbb{E}(W^r_1)$ exists. Call it $\mathfrak{L}$. We then have

$$\mathfrak{L} \geq W^1_1 + (1-n)(1-\mathfrak{L})rW^1_1$$

and, therefore,

$$\mathfrak{L} \geq \frac{W^1_1 + (1-n)rW^1_1}{1 + (1-n)rW^1_1} \, .$$

As the right hand side approaches $1$ as $r \to \infty$, the expected value of $W^r_1$ approaches $1$ as well. Recalling (1), we have the desired result. □

**Theorem 3** (Introspection)**.** *It is strictly better to blame yourself rather than the game mechanism to learn faster; that is, learning happens faster when $n \to 0$.*

*Proof.* Notice that for Cases 1–6, $\frac{dP(r)}{dr}$ can be expressed in the form

$$\frac{dP(r)}{dr} = (1-n)\vartheta(W_1^r, L_1^r, C_1^r, W_2^r, L_2^r, C_2^r, W_3^r, L_3^r)$$

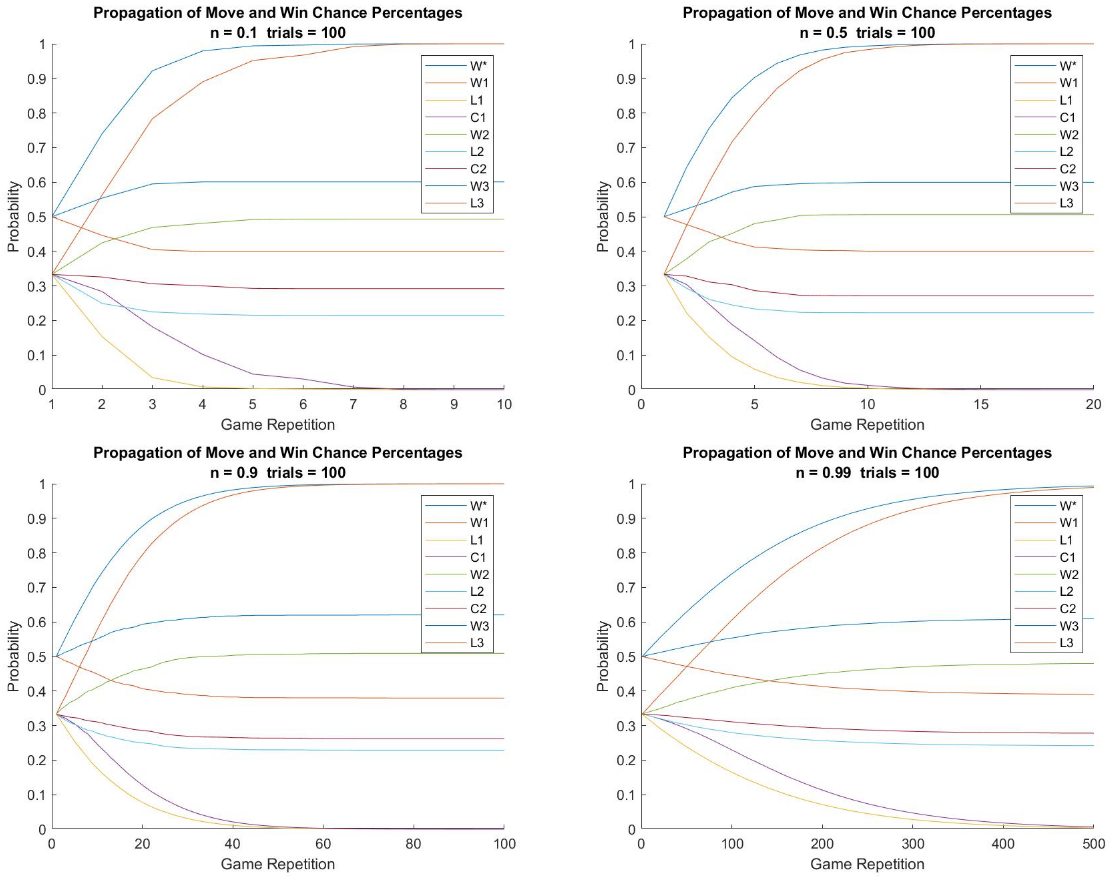
where $\vartheta(W_1^r, L_1^r, C_1^r, W_2^r, L_2^r, C_2^r, W_3^r, L_3^r)$ is a strictly positive function of choice probabilities at every step. Therefore,

$$\begin{aligned}\frac{\partial}{\partial n}\left[\frac{dP(r)}{dr}\right] &= \frac{\partial}{\partial n}[(1-n)\vartheta(W_1^r, L_1^r, C_1^r, W_2^r, L_2^r, C_2^r, W_3^r, L_3^r)] \\ &= -\vartheta(W_1^r, L_1^r, C_1^r, W_2^r, L_2^r, C_2^r, W_3^r, L_3^r) \le 0\,.\end{aligned}$$

Thus, it is seen that as $n$ decreases, $\frac{dP(r)}{dr}$ increases. □

## 2.3 Simulations of a Basic Game

We simulate the game and observe how the probability of each decision propagates through each iteration of the game. For each graphed result, we use initial values $W_1^1 = L_1^1 = C_1^1 = \frac{1}{3}$, $W_2^1 = L_2^1 = C_2^1 = \frac{1}{3}$, and $W_3^1 = L_3^1 = \frac{1}{2}$, while the value for $n$ varies. The results are as follows:



Thus, the reader can easily see that in each of the cases, $W_1$, $W_2$, and $W_3$, as well as $W^*$, the overall probability of winning the game, are strictly increasing. Similarly, as $n$ decreases, the number of repetitions of the game it takes to reach near perfect play $W^* > 0.9$ also decreases.

# 3 Generalizations

## 3.1 General Definitions

Having analyzed how play propagates in the original basic game, we now define the general class of games under consideration and generalize the learning mechanism to incorporate a richer variety of situations.

**Definition 4** (Simple Sequential Game)**.** A *Simple Sequential Game G* can be defined as follows. Let $D = \{d_1,\ d_2,\ \ldots,\ d_n\}$ be a nonempty, finite set of elements called decision nodes. We call $d_1$ the initial decision node. There exists a non-empty subset of $D$ made up of terminal decision nodes. For each decision node $d_j \in D$, there is a nonempty finite set $A_j$ that represent the set of actions available at that decision node.

There is a function $O$ defined on the set of all possible actions with the following properties:

- for each non-terminal decision node $d_j$ and each action $a \in A_j$, the outcome $O(a)$ is either $W$, $L$, or $C$, with at least one action mapped to each outcome;
- for each action $a \in A_j$ with $O(a) = C$, there exists a "next node" $d_k$ with $k > j$ such that no nodes/actions can share the same "next node";
- for each terminal decision node $d_\ell$ and each action $a \in A_\ell$, the outcome $O(a)$ is either $W$ or $L$, with at least one action mapped to each outcome.

There is a single player in this game who

- knows when they are at a decision node $d_j$ and can differentiate between decision nodes;
- can differentiate between actions $\{a_1, a_2, \ldots, a_i\} \in A_j$;
- does NOT know the mapping $O$ between actions and outcomes, but does know that the possible outcomes of the each move are $\{W,\ L,\ C\}$ and that the game ends in either $W$ or $L$;
- strictly prefers $W$ to $L$.

This finishes the definition of a Simple Sequential Game.

We also define a path $P_j$ from initial decision node $d_1$ to the decision node $d_j$ to be a set of decision nodes $\{d_1,\ d_{i_1},\ d_{i_2}, \ldots,\ d_{i_k}, d_j\}$ with $1 < i_1 < i_2 < \ldots < i_k$ and actions $\{a_1,\ a_{i_1},\ a_{i_2}, \ldots,\ a_{i_k}, a_j\}$ satisfying $O(a_m) = C$ for all $m \in \{1,\ i_1,\ i_2, \ldots, i_k\}$ and $O(a_j) \in \{W, L\}$. Each decision node in the list (except $d_1$) is the next node for the previous node and action. Let $\overline{A}_j$ be the set of actions in the path $P_j$.

One can see that the game defined in Section 2.2 is a Simple Sequential Game.

**Remark 5.** *The definition used for a Simple Sequential Game generates a subset of* extensive form games.

Based on the definition of an extensive form game introduced in Brihaye et al. [1], is easy to see that a Simple Sequential Game has a set of players $N$ with $|N| = 1$, a finite action profile $A$, and a non-empty set of outcomes $O$. Noticing that the game only continues if the outcome corresponding to the player's move at any decision node is $C$ leads to the conclusion that, for every node, it is possible to trace a path of $C$s to the very first decision node. Similarly, for all non-terminal nodes, the one player playing the game is always the one making the moves, while the mapping $O$ in Definition 4 associates an outcome with every action at any terminal node. Lastly, among the two outcomes, the preference ordering for the player is $W \succ L$.[6]

**Definition 6** (Brute Force Learning Mechanism)**.** A *Brute Force Learning Mechanism* $M$ for playing Simple Sequential Games can be defined as the triple $\langle G,\ n,\ I\rangle$ where

- $G$ represents a Simple Sequential Game;
- $n \in [0, 1)$ represents the agent's disposition against learning (inertia of the status quo; smaller $n$ correlates to more drastic learning and vice versa);
- $I$ represents the initial probability of playing each move[7]. Letting $I(a_i)$ denote the initial probability of playing move $a_i$, $I$ must satisfy $\sum\limits_{a_i \in A_j} I(a_i) = 1$ for all $d_j \in D$.

In this new general form of the learning mechanism, let $p(a)$ be the probability of playing move $a$. The learning mechanism $M$ is then a function with decision node $d_j$, move played $a^*$, and probabilities $p(a)$ that correspond to each of the moves $a \in A_j$ that outputs new probabilities $p'(a)$ (that once again, correspond to each of the moves $a \in A_j$). Similar to Section 2.2, we now work through each of the cases based on the outcome of the game stemming from the move chosen.

**Case One**: $O(a^*) = W$

$$p'(a) = \begin{cases} np(a) = p(a) - (1-n)p(a) & ,\ a \in A_j \setminus \{a^*\} \\ p(a^*) + \sum\limits_{a \in A_j \setminus \{a^*\}} (1-n)p(a) & ,\ a = a^* \end{cases}$$

The current repetition of the game ends here.

**Case Two:** $O(a^*) = L$

$$p'(a) = \begin{cases} p(a) + \dfrac{p(a)}{1-p(a^*)}(1-n)p(a^*) & ,\ a \in A_j \setminus \{a^*\} \\ np(a^*) = p(a^*) + (n-1)p(a^*) & ,\ a = a^* \end{cases}$$

The current repetition of the game ends here.

[6] The difference between a Simple Sequential Game and a perfect information extensive form game is that the player in this game does not know the mapping $O$ from actions to outcomes.

[7] Think of this as the initial conditions.

**Case Three:** $O(a^*) = C$

$$p'(a) = p(a) \qquad \text{for all} \quad a \in A_j \, .$$

And the game now moves onto a new node $d_k$[8], in which the learning mechanism follows the same algorithm at that new node.

It is easy to see that the learning mechanism defined in Section 2.1 is a Brute Force Learning Mechanism. This finishes the definition of Brute Force Learning Mechanisms.

**Remark 7.** *A Brute Force Learning Mechanism $M$ has the Markov property of the $(r+1)$th iteration of the game being dependent on only the $r$th iteration; the history of learning does not matter when determining how the game propagates.*

## 3.2 Generalized Theorems

We now generalize the two theorems presented in Section 2.2 to all Simple Sequential Games with a Brute Force Learning Mechanism. For future reference, let $P(G, r)$ denote the probability of winning Simple Sequential Game $G$ in the $r$th repetition.

**Theorem 8.** *(Stronger Eventual Perfection) Any Brute Force Learning Mechanism $M$ will eventually converge to perfect play given any Simple Sequential Game $G$ given nonzero initial probabilities to play each move and $n \in [0, 1)$. That is, $\lim\limits_{r \to \infty} P(G, r) = 1$.*

**Lemma 9.** *$P(r)$ is strictly increasing in $r$ for all Simple Sequential Games.*

*Proof.* Let $P(G, r+1 | O(a^r) \in \{W, L\})$ denote the probability of winning the game in the $r+1$th repetition given that the terminal action $a^r$ in the $r$th repetition resulted in either a win or a loss. By definition, any actions that result in a continuation of the game do not result in a change in $p(a)$ for any $a \ \in A_j$ and thus only terminal actions need to be considered.

Starting off, note that the probability to win is equal to the summation over all decision nodes of the probability to get to a certain node multiplied by the probability to win at that node:

$$P(G, r) = \sum_{d \in D} \Bigg( \prod_{a \in \overline{A}_d} p(a) \sum_{a \in A_d \, : \, O(a) = W} p(a) \Bigg) \, .$$

Now, let $S_{d^*}$ denote the decision nodes in the subgame that lie on a path starting at decision node $d^*$. Note that every possible outcome only changes the move probabilities at a specific decision node. Now, let $d^*$ denote any specific decision node and $P(G, r, d^*)$ denote the probability of winning the subgame starting at $d^*$. Let $n(a, d^*)$ denote the decision node reached after choosing action $a$ at node $d^*$ where $O(a) = C$. Now, $P(G, r)$ can be expressed as the summation of the probability to win over all

[8] $k \neq j + 1$ as $d_j$ could have multiple actions that result in the continuation of the game.

nodes not in the subgame $S_{d^*}$ plus the probability to reach the subgame $S_{d^*}$ multiplied by the probability to win in the subgame. For some arbitrary fixed $d^* \in D$, we have:

$$\begin{aligned} P(G,r) = &\sum_{d \in D \setminus S_{d^*}} \Bigg( \prod_{a \in \overline{A}_d} p(a) \sum_{O(a)=W} p(a) \Bigg) \\ &+ \prod_{a \in \overline{A}_{d^*}} p(a) \Bigg( \sum_{a \in A_d \,:\, O(a)=W} p(a) + \sum_{a \in A_d \,:\, O(a)=C} p(a) P\big(G, r, n(a, d^*)\big) \Bigg) . \end{aligned}$$

Then, letting $a^r \in A_{d^*}$ be the move played in the $r$th repetition,[9]

$$\begin{aligned} P(G,r+1|a^r) = &\sum_{d \in D \setminus S_{d^*}} \Bigg( \prod_{a \in \overline{A}_d} p(a) \sum_{O(a)=W} p(a) \Bigg) \\ &+ \prod_{a \in \overline{A}_{d^*}} p(a) \Bigg( \sum_{a \in A_d \,:\, O(a)=W} p'(a) + \sum_{a \in A_d \,:\, O(a)=C} p'(a) P\big(G, r, n(a, d^*)\big) \Bigg) . \end{aligned}$$

as the probabilities to win in decision nodes not in the subgame remain unchanged. We now want to show that $P(G, r+1|a^r) - P(G,r) > 0$. Note that both $P(G, r+1|a^r)$ and $P(G,r)$ share the same first term which can be canceled out, leaving the inequality

$$\begin{aligned} &P(G,r+1|a^r) - P(G,r) \\ = &\prod_{a \in \overline{A}_{d^*}} p(a) \Bigg( \sum_{a \in A_d \,:\, O(a)=W} p'(a) + \sum_{a \in A_d \,:\, O(a)=C} p'(a) P\big(G, r, n(a, d^*)\big) \Bigg) \\ - &\prod_{a \in \overline{A}_{d^*}} p(a) \Bigg( \sum_{a \in A_d \,:\, O(a)=W} p(a) + \sum_{a \in A_d \,:\, O(a)=C} p(a) P\big(G, r, n(a, d^*)\big) \Bigg) > 0 . \end{aligned}$$

Based on the initial conditions, $p(a) \neq 0$ for all $a$; this means that $\prod_{a \in \overline{A}_{d^*}} p(a) \neq 0$, so dividing that factor from both sides yields

$$\begin{aligned} &\sum_{a \in A_d \,:\, O(a)=W} p'(a) + \sum_{a \in A_d \,:\, O(a)=C} p'(a) P\big(G, r, n(a, d^*)\big) \\ - &\Bigg( \sum_{a \in A_d \,:\, O(a)=W} p(a) + \sum_{a \in A_d \,:\, O(a)=C} \big(p(a) P\big(G, r, n(a, d^*)\big)\big) \Bigg) > 0 . \end{aligned} \tag{3}$$

Now, we go into each of the two cases based on the outcome of the game from playing $a^r$ and substitute in the values of $p'(a)$ based on Definition 3.1.

[9] This is a general case before even considering what $O(a^r)$ is; that will be done shortly.

**Case One:** $O(a^r) = W$

$$
\begin{aligned}
&\sum_{a\in A_d\,:\,O(a)=W} p'(a) + \sum_{a\in A_d\,:\,O(a)=C} p'(a)P\big(G,r,n(a,d^*)\big) \\
&= \sum_{a\in A_d\setminus\{a^r\}\,:\,O(a)=W} \big(p(a)-(1-n)p(a)\big) + p(a^r) + \sum_{a\in A_{d^*}\setminus\{a^r\}} (1-n)p(a) \\
&+ \sum_{a\in A_d\,:\,O(a)=C} np(a)P\big(G,r,n(a,d^*)\big) \\
&= \sum_{a\in A_d\setminus\{a^r\}\,:\,O(a)=W} p(a) - \sum_{a\in A_d\setminus\{a^r\}\,:\,O(a)=W} (1-n)p(a) + p(a^r) + \sum_{a\in A_d\setminus\{a^r\}\,:\,O(a)=W} (1-n)p(a) \\
&+ \sum_{a\in A_d\,:\,O(a)=L} (1-n)p(a) + \sum_{a\in A_d\,:\,O(a)=C} (1-n)p(a) + \sum_{a\in A_d\,:\,O(a)=C} np(a)P\big(G,r,n(a,d^*)\big) \\
&= \sum_{a\in A_d\,:\,O(a)=W} p(a) + \sum_{a\in A_d\,:\,O(a)=L} (1-n)p(a) + \sum_{a\in A_d\,:\,O(a)=C} p(a)\big(1+n(P\big(G,r,n(a,d^*)\big)-1)\big)\,. \qquad (4)
\end{aligned}
$$

Substituting (4) into (3) and canceling terms results in the following expression for the left-hand side of (3):

$$
\begin{aligned}
&\sum_{a\in A_d\,:\,O(a)=L} (1-n)p(a) + \sum_{a\in A_d\,:\,O(a)=C} p(a)\big(1+n\big(P\big(G,r,n(a,d^*)\big)-1\big)\big) - \sum_{a\in A_d\,:\,O(a)=C} p(a)P\big(G,r,n(a,d^*)\big) \\
&= (1-n)\Bigg(\sum_{a\in A_d\,:\,O(a)=L} p(a) + \sum_{a\in A_d\,:\,O(a)=C} p(a)\Big(1-P\big(G,r,n(a,d^*)\big)\Big)\Bigg)
\end{aligned}
$$

which is clearly greater than 0, as $n < 1$, $p(a) > 0$, and $P\big(G,r,n(a,d^*)\big) < 1$.

**Case Two**: $O(a^r) = L$

$$
\begin{aligned}
&\sum_{a\in A_d\,:\,O(a)=W} p'(a) + \sum_{a\in A_d\,:\,O(a)=C} p'(a)P\big(G,r,n(a,d^*)\big) \\
&= \sum_{a\in A_d\,:\,O(a)=W} \Big(p(a) + \frac{p(a)}{1-p(a^r)}(1-n)p(a^r)\Big) \\
&+ \sum_{a\in A_d\,:\,O(a)=C} \Big(p(a) + \frac{p(a)}{1-p(a^r)}(1-n)p(a^r)\Big)P\big(G,r,n(a,d^*)\big) \\
&= \sum_{a\in A_d\,:\,O(a)=W} p(a) + \sum_{a\in A_d\,:\,O(a)=W} \frac{p(a)}{1-p(a^r)}(1-n)p(a^r) + \sum_{a\in A_d\,:\,O(a)=C} p(a)P\big(G,r,n(a,d^*)\big) \\
&+ \sum_{a\in A_d\,:\,O(a)=C} \frac{p(a)}{1-p(a^r)}(1-n)p(a^r)P\big(G,r,n(a,d^*)\big)\,. \qquad (5)
\end{aligned}
$$

Similarly, we now substitute (5) into (3) and cancel terms to find the left-hand side of (3):

$$\begin{aligned}
&\sum_{a\in A_d\,:\,O(a)=W}\frac{p(a)}{1-p(a^r)}(1-n)p(a^r)+\sum_{a\in A_d\,:\,O(a)=C}\frac{p(a)}{1-p(a^r)}(1-n)p(a^r)P\big(G,r,n(a,d^*)\big)\\
&=(1-n)\Bigg(\sum_{a\in A_d\,:\,O(a)=W}\frac{p(a)}{1-p(a^r)}p(a^r)+\sum_{a\in A_d\,:\,O(a)=C}\frac{p(a)}{1-p(a^r)}p(a^r)P\big(G,r,n(a,d^*)\big)\Bigg)
\end{aligned}$$

which is positive.

Since $d^*$ is arbitrary, it is shown that no matter what happens in the $r$th repetition of the game, $P(G,r+1)>P(G,r)$. □

Now, we use logic similar to that of Theorem 2.1 to prove that the probability to win converges to $1$.

*Proof.* Unlike in Section 2, the probability to play each individual winning move is not strictly increasing, as playing a winning move decreases the probability to play all other moves, even if some other moves are winning. Instead, we look at the overall probability to win, $\sum_{a\in A_d\,:\,O(a)=W}P(a)$.

First, we show that the probability to play a winning move on turn one is strictly increasing. It is easy to see how if the move played on turn one results in a loss, the overall probability to win in turn one increases (as the probability of every individual winning move increases) while if the game continues onto move two, the probability to win remains unchanged. Now, consider if the move played $a^*$ in the first move results in a win:

$$\begin{aligned}
\sum_{a\in A_d\,:\,O(a)=W}p'(a)&=p'(a^*)+\sum_{a\in A_d\setminus\{a^*\}\,:\,O(a)=W}p'(a)\\
&=p(a^*)+\sum_{a\in A_d\setminus\{a^*\}}(1-n)p(a)+\sum_{a\in A_d\setminus\{a^*\}\,:\,O(a)=W}\big(p(a)-(1-n)p(a)\big)\\
&=p(a^*)+\sum_{a\in A_d\setminus\{a^*\}\,:\,O(a)=W}p(a)+\sum_{a\in A_d\,:\,O(a)\neq W}(1-n)p(a)\\
&=\sum_{a\in A_d\,:\,O(a)=W}p(a)+\sum_{a\in A_d\,:\,O(a)\neq W}(1-n)p(a)\\
&>\sum_{a\in A_d\,:\,O(a)=W}p(a)
\end{aligned}$$

as $1-n$ and $p(a)$ are both strictly positive. Once again, let a type $I$ move be one that results in a win in round $1$ and $J_r$ be the set of repetitions in which a type $I$ move is played. If there are $j$ moves of type $I$ in the first $r$ repetitions of the game, denote that to be event $E^r_j$.

Then, define two random variables on the events $E^r_0,E^r_1,E^r_2,\ldots,E^r_r$:

- $S_1^{r+1}(E^r_j)$ is the probability that the player plays a winning move in the first move of round $r+1$ after $r$ game moves, of which $j$ were of type $I$.
- $\Upsilon^r$ is the number of moves of type $I$ in the first $r$ rounds. Note that $\Upsilon^r(E^r_j)=j$.

Finally, let $\Phi^r$ be the binomial random variable with $r$ trials and $w := \sum_{a\in A^1} p(a)$ probability of success in each trial where $A^i$ is the set of all actions $a$ in the first move of the $i$th round of a given game with outcome $O(a) = W$. Then,

$$E(\Upsilon^r) = \sum_{j=1}^{r} jP(j = |J_r|) \geq \sum_{j=1}^{r} j \binom{r}{j} \sum_{a\in W^1} p(a) = E(\Phi^r) = rw$$

as $\displaystyle\sum_{a\in W^1} p(a)$ is strictly non-decreasing. Also,

$$\begin{aligned} S^{r+1} = \sum_{a\in W^r} p(a) &\geq w + \sum_{i\in J_r} \left( (1-n)\Big(1 - \sum_{a\in W^i} p(a)\Big)\right) \\ &\geq w + \sum_{i\in J_r} (1-n)\Big(1 - \sum_{a\in W^r} p(a)\Big) \\ &= w + (1-n)\Big(1 - \sum_{a\in W^r} p(a)\Big)|J_r| \end{aligned}$$

as $1 - \displaystyle\sum_{a\in W^r} p(a) \geq 1 - \sum_{a\in W^r} p(a)$ for all $i$ and since $(1-n)\Big(1 - \displaystyle\sum_{a\in W^r} p(a)\Big)$ is independent of $i$.

Taking the expected value of both sides yields

$$\begin{aligned} E(S^{r+1}) &\geq E\left( w + (1-n)\Big(1 - \sum_{a\in W^r} p(a)\Big)|J_r|\right) \\ &= w + (1-n)\left(1 - E\Big(\sum_{a\in W^r} p(a)\Big)\right)E(\Upsilon^r) \\ &\geq w + (1-n)\big(1 - E(S^r)\big)rw\,. \end{aligned}$$

Since $E(S^{r+1})$ is a non-decreasing sequence bounded above, $\lim\limits_{r\to\infty} E(S^{r+1}) = \lim\limits_{r\to\infty} E(S^r)$ exists. Call it $\mathfrak{L}$. We then have

$$\mathfrak{L} \geq w + (1-n)(1-\mathfrak{L})rw$$

and so

$$\mathfrak{L} \geq \frac{w + (1-n)rW_1^1}{1 + (1-n)rw}\,.$$

As the right hand side approaches 1 as $r \to \infty$, the expected value of $S^{r+1}$ approaches 1 as well. Recalling that $S^{r+1}$ is the probability to win on the first move alone, the overall probability to win must be at least $S^{r+1}$ and thus also converges to 1. □

We now also generalize Theorem 3 to the class of sequential games and brute force learning mechanism introduced in this section.

**Theorem 10** (Generalized Introspection)**.** *Given any Brute Force Learning Mechanism $M$ and Simple Sequential Game $G$, learning happens faster when $n \to 0$.*

*Proof.* Maximizing the learning rate is the essentially the same as maximising the difference between the probability of winning in successive iterations of the game. The problem can be expressed as the optimisation problem

$$\max_n \big(P(G, r+1) - P(G, r)\big) \tag{6}$$

Following the logic used to reach (3), solving (6) is the same as solving the problem

$$\max_n \Bigg(\Big[\sum_{a \in A_d \,:\, O(a)=W} p'(a) + \sum_{a \in A_d \,:\, O(a)=C} p'(a)P\big(G, r, n(a, d^*)\big)\Big] \\ -\Big[\sum_{a \in A_d \,:\, O(a)=W} p(a) + \sum_{a \in A_d \,:\, O(a)=C} p(a)P\big(G, r, n(a, d^*)\big)\Big]\Bigg). \tag{7}$$

Now, notice in both of the cases where $O(a) = W$ and $O(a) = L$, the interior expression of (7) can be expressed in the form

$$(1-n)f(G, r, d^*, a^r)$$

where $f(G, r, d^*, a^r)$ is a strictly positive function. Now, taking the partial derivative of the difference with respect to $n$ yields

$$\frac{\partial}{\partial n}\big((1-n)f(G, r, d^*, a^r)\big) = -f(G, r, d^*, a^r)$$

which is strictly negative, as $f(G, r, d^*, a^r)$ was originally strictly positive. Thus, as $n$ decreases and approaches $0$, the increase in winning probabilities between each iteration grows larger, which contributes to a larger learning effect. □

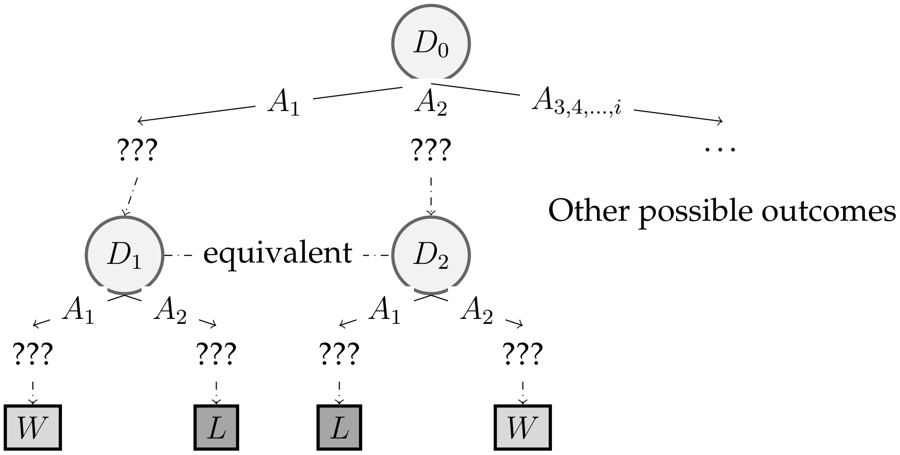
## 3.3 Conditions on a Simple Sequential Game

We will now take a moment to talk about why a brute force learning mechanism only works on Simple Sequential Games defined in Definition 4 instead of all sequential games. Specifically, there are three differences between a Simple Sequential Game and the general extensive-form game (once again, using the definition in Brihaye et al. [1]).

First is information. In a Simple Sequential Game, the player has perfect information on the structure of the game, knowing which decision node they are at but are unaware of the payoff function. In perfect-information extensive-form games, the player knows everything in a Simple Sequential Game, but also knows the payoff function. However, in an imperfect-information extensive-form game, the player does not know what decision node they may be at. To see how this would affect a Brute Force Learning Mechanism, consider the following (sub-)game:

It is easy to see how in a situation such as this, where move $A_1$ could lead to a win in some situations while resulting in a loss in others, a brute force learning mechanism

Figure 3: Case of Imperfect Information

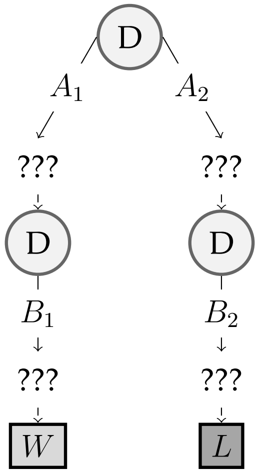


would both increase and then decrease the probability of playing $A_1$ and $A_2$, leading to no determinate outcome.

The next difference is the payoff function. For this game, we required that the outcome was discretely win or loss, while in the general extensive-form game, the outcome of the game is assigned a value in "utils" to each player. This difference can easily be overcome by simply imposing a threshold value such that any payoff below the threshold is counted as a loss and every payoff above the threshold is counted as a win. However, this would mean that a brute force learning mechanism only results in "decent" play instead of "perfect" play, as anything above the threshold is seen as good enough by the learning mechanism.

Finally, it is possible for some decision nodes of a general extensive-form game to not have any winning outcomes. For a simple example, consider the game in Figure 4 where the decision node reached after playing move $A_2$ has no winning outcomes:

Figure 4: Case of Unfavorable Outcomes



Given such a game, it is easy to see that the the probabilities of playing moves $B_1$ and $B_2$ are unchanged due to them being the only possible move, while the probability of playing moves $A_1$ and $A_2$ also do not change through iterations of the game due to a brute force learning mechanism not changing the probability of playing moves that result in a continuation of the game. Thus, play does not improve (or even change at all) through multiple iterations of the game.

## 4 Further Research

Further investigation could be done along several routes. Based on the conditions on a Simple Sequential Game, different learning algorithms can be designed to allow for non-deterministic results and for "mean" games where there isn't a way to win at every decision node. If generalized to all sequential games, this could provide another method to utilize an algorithm to brute force solve complicated games.

Given the iterative nature of a brute force learning, another possible application would be to test a brute force learning mechanism against a machine learning algorithm to see which converges to perfect play faster given the same game and initial conditions.[10]

## Acknowledgements

The author would like to thank Dr. Daniel C. Biles of Belmont University in Nashville, Tennessee for his assistance ranging from answering questions about mathematical writing to comments about the paper itself.

## References

[1] T. Brihaye, G. Geeraerts, M. Hallet and S. Le Roux, Dynamics and coalitions in sequential games, in *Proc. 8th Internat. Symp. on Games, Automata, Logics and Formal Verification*, pages 136–150, Electron. Proc. Theor. Comput. Sci. (EPTCS) 256, EPTCS, 2017.

[2] D. Fudenberg and D.K. Levine, Whither game theory? Towards a theory of learning in games, *J. Econ. Perspect.* **30** (2016), 151–170.

[3] D. Fudenberg and A. Liang, Predicting and understanding initial play, *Am. Econ. Rev.* **109** (2019), 4112–4141.

[4] J.B. van Huyck, R.C. Battalio and R.O. Beil, Tacit coordination games, strategic uncertainty, and coordination failure, *Am. Econ. Rev.* **80** (1990), 234–248.

[10] Although applications of machine learning to game theory are relatively new, for an interesting example, see Fudenberg and Liang [3].

[5] M.L. Littman, Friend-or-foe Q-learning in general-sum games, pages 322–328, in *Proc. 18th Internat. Conf. Machine Learning*, June 2001.

[6] A. Mahajan and D. Teneketzis. *Multi-Armed Bandit Problems*, Springer, Boston, MA, 2008.

[7] A. Now, P. Vrancx and Y.-M. De Hauwere, *Game Theory and Multi-agent Reinforcement Learning*, Springer, Berlin, Heidelberg, 2012.

[8] Fedor Sandomirskiy, On repeated zero-sum games with incomplete information and asymptotically bounded values, *Dyn. Games Appl.* **8** (2018), 180–198.

[9] A. Valluri, Learning and cooperation in sequential games, *Adapt. Behav.* **14** (2006), 195–209.