1 Introduction

2 Background

2.1 Description of Pairs

2.2 Definitions

Throughout the remainder of this paper, let n be the rank of the highest card in the deck we are considering, and let m be the number of players. This allows us to analyze more general decks other than the standard triangular Pairs deck in which $n = 10$.

Definition 1. A **deck** is an n-dimensional vector $d \in \mathbb{Z}^n$. We denote the jth element of a deck d as $d[j]$. For a deck d and an integer $q \in \mathbb{Z}$, we use the notation $d + q\langle j \rangle$ to the refer to the deck in which the jth element of d is incremented by q, i.e. $(d + q\langle i \rangle)[j] = d[j] + q\delta_{ij}$.

We use the abstract deck d to represent a physical deck with $d[i]$ cards of rank i. Then, $d - \langle i \rangle$ represents removing a card of rank i from the deck. For example, if a player hits from a deck d and receives a card of rank i, the new deck is $d - \langle i \rangle$.

Definition 2. The **normalized deck** of a deck $d \neq 0$ is the vector $d/\left(\sum_j d[j]\right)$. We denote the normalized deck of a deck $d \neq 0$ as $\tilde{d}$.

We can think of a normalized deck $\tilde{d}$ as a probability distribution $p(j) = \tilde{d}[j]$, where $p(j)$ represents the probability of drawing a card of rank j from a deck d uniformly at random. It is easily seen from the definition of a normalized deck that $p(j)$ corresponds to a valid probability distribution.

Definition 3. A **stack** is an n-dimensional vector $s \in \{0, 1\}^n$. For a stack s and an index j for which $s[j] = 0$, we use the notation $s + \langle j \rangle$ to refer to the stack in which the jth element of s is incremented by 1.

We use a stack s to represent the faceup cards of a player, where $s[j] = 1$ if and only if the player a faceup card of rank j. Because whenever a faceup pair occurs, either the dealt card is discarded (in the case of dealing to determine who goes first) or the player loses (in the case hitting), a stack s is sufficient to characterize any player's hand at any point in the game.

Definition 4. Let $D = \mathbb{Z}^n$ be the set of possible decks. Let $S = \{0, 1\}^n$ be the set of possible stacks. Then, a **configuration** of the game is a tuple $(d, s_1, s_2, \ldots, s_m)$, where $d \in D$ and $s_i \in S$ for $i \in \{1, 2, \ldots, m\}$.

For a configuration $(d, s_1, s_2, \ldots, s_m)$, the deck d represents the possible cards that

3 Analysis of the Two-Player Case

We begin by considering a single round of Pairs with two players. In this single round game, the score is the rank of either the paired card or the penalty card taken for folding, and we take the payoffs to be negative of the score for the losing player, the one that folds or forms a pair, and positive of the score for the winning, other player. We can think of these payoffs corresponding to the losing player paying the winning player the score that the losing player accumulated in the round, so over many rounds, the overall winner has gained the difference in score from the losing player.

3.1 Analysis

For simplicity, we assume that whenever Fold and Hit at some configuration are expected to yield the same payoff, the players prefer Fold. This allows us to avoid bothersome edge cases in which a player faced with both choices yielding the same payoff might change his action depending on which history led him to the current configuration of the game. Under this assumption, we can prove the following theorem.

Theorem 1. In a subgame perfect equilibrium, a player's action at each decision node can depend only on the configuration of the game.

Proof. For any configuration c, let $k_c = \sum_i \sum_j s_i[j]$, where s_i are the stacks of c, denote the total number of cards among all of the players' faceup cards. Because $s_i[j] \in \{0, 1\}$ for each component, we have that $k_c \le mn$, which is bounded, and it follows that there are finitely many possible values for k_c.

Suppose for contradiction that we have some subgame perfect equilibrium in which there exist two decision nodes x and y which have the same configuration c, but the player's action at the two nodes differ, and assume without loss of generality that x and y are chosen so that the associated k_c is as large as possible. Then, the possible actions for the player to move are Hit or Fold. In this case of Fold, the player takes the lowest-rank faceup card, which is encoded in the stacks of the configuration, and the resulting payoffs at either x or y are the same.

In the case of Hit, the probabilities of drawing each card are encoded in the deck of the configuration, so they are the same at either x or y. After choosing Hit, suppose that the player receives a card of rank j. Then, either the player has formed a pair and received a payoff of $-j$, or the player has avoided forming a pair, moving the game to a new decision node x' or y'. Note that both x' and y' will have the same configuration c' with $k_{c'} = k_c + 1 > k_c$, and because the number of faceup cards is nondecreasing throughout a round, we know that the players' strategies from here on only depend on the configuration of the game at each node, so the payoffs from the subtrees rooted at x' or y' depend only on c' and are thus equal. Thus, the payoffs from receiving any card from Hit are the same at both x and y, so the expected payoffs from Hit at x and y are also the same.

But if the expected payoffs at x and y are the same for each action, then the player would have chosen the action resulting in the higher payoff, and if both actions result in the same payoff, the player would have chosen to Fold, so it cannot be that the action at x and y, which have the same configuration, differ, which is a contradiction. $\square$

Corollary 2. Let C be the set of possible configurations. Then, any subgame perfect equilibrium can be described by a function $a : C \to \{\text{Hit}, \text{Fold}\}$. From any decision node, the expected payoff of the moving player from following the subgame perfect equilibrium given by a can be described by a function $u : C \to \mathbb{R}$.

Proof. Follows immediately from Theorem 1. $\square$

Using Corollary 2, we can find a simple recursive formula for finding a subgame perfect equilibrium.

Theorem 3. Under a subgame perfect equilibrium defined by Corollary 2, we have
$$u(d, s_1, s_2) = \max\{u_{\text{Fold}}(d, s_1, s_2), u_{\text{Hit}}(d, s_1, s_2)\} \tag{1}$$
where the expected payoff from Fold is
$$u_{\text{Fold}}(d, s_1, s_2) = -\min\{j \text{ such that } s_i[j] = 1 \text{ for some } i\} \tag{2}$$
and the expected payoff from Hit is
$$u_{\text{Hit}}(d, s_1, s_2) = -\sum_{j=1}^{n} \left(j s_1[j]\tilde{d}[j] + u(d - \langle j \rangle, s_2, s_1 + \langle j \rangle)(1 - s_1[j])\tilde{d}[j] \right). \tag{3}$$

Here, we have assumed that we never need to reshuffle. If reshuffling is necessary, then we will need to replace $d - \langle j \rangle$ in the Equation 3 with whatever deck results from reshuffling.

To prove this theorem, we will first need to prove two lemmas showing that the expected payoffs from Fold and Hit are indeed given by 2 and 3.

Lemma 4. Under a subgame perfect equilibrium defined by Corollary 2, the expected payoff from Fold is given by Equation 2.

Proof. After folding, the folding player chooses a faceup card of rank j from the table and receives a payoff $-j$. Clearly, he chooses the lowest-rank faceup card, which is given by the min expression in 2, and it follows that his (expected) payoff is given by Equation 2. $\square$

Lemma 5. Under a subgame perfect equilibrium defined by Corollary 2, the expected payoff from Hit is given by Equation 3.

Proof. After hitting, the hitting player receives a card at random, and the probability of receiving a card of rank j is given by $\tilde{d}[j]$. Suppose that the hitting player receives a card of rank j. If he has a card of rank j among his faceup cards, i.e. $s_1[j] = 1$ and $1 - s_1[j] = 0$, then he has formed a pair and receives a payoff of $-j$. If instead he does not have a card of rank j among his faceup cards, i.e. $s_1[j] = 0$ and $1 - s_1[j] = 1$, then he does not form a pair, and the new configuration of the game is $c = (d - \langle j \rangle, s_2, s_1 + \langle j \rangle)$ as the deck has lost a card of rank j, the stack of the moving player has gained a card of rank j, the stack of the other player has not changed, and now it is the turn of the other player. Because this is a zero-sum game, the expected payoff of the player who just moved is then $-u(c)$. We see that the payoffs of both cases are encoded in the expression

$$- (j s_1[j] + u(d - \langle j \rangle, s_2, s_1 + \langle j \rangle)(1 - s_1[j]))$$

as exactly one of $s_1[j]$ and $1 - s_1[j]$ is one, while the other is zero, so only the appropriate term contributes to the payoff. Finally, it follows that the expected payoff from Hit is the above expression weighted appropriately by the probability $\tilde{d}[j]$ of receiving a card of rank j, which is the expression given in 3. $\square$

With these two lemmas, we are now in a position to prove Theorem 3.

Proof of Theorem 3. With Lemmas 4 and 5, we see that the expected payoffs from Fold and Hit are indeed given by Equations 2 and 3, and because at every node in a subgame perfect equilibrium, the player chooses the action that results in the highest expected payoff, his resulting payoff is then given by the maximum of the two expected payoffs, as in Equation 1, completing the proof. $\square$

Corollary 6. Under a subgame perfect equilibrium defined by Corollary 2, we have

$$a(d, s_1, s_2) = \text{Fold if } u(d, s_1, s_2) = u_{\text{Fold}}(d, s_1, s_2)$$
$$\text{Hit otherwise.} \tag{4}$$

Proof. Follows from players choosing the payoff-maximizing action, preferring Fold when expected payoffs are equal. $\square$

Corollary 7. The subgame perfect equilibrium given by Equation 4 is unique.

Proof. Consider the recurrence relation of Equation 1. Note that the only recursive term comes from Equation 3. Consider the expression $\sum_i \sum_j s_i[j]$, which is an integer between 0 and mn and increments by 1 with each step of the recurrence. Once $\sum_i \sum_j s_i[j] = mn$, the second term of Equation 3 is 0 as $1 - s_i[j] = 0 \ \forall \ i, j$, so the recursion has limited depth of mn. Furthermore, Equation 3 is still well-valued when this occurs, so the boundary conditions of the recurrence are "built in" with Equation 3. Thus, Equation 1 uniquely determines $u(d, s_1, s_2)$.

Corollary 2 shows that every subgame perfect equilibrium must have the same indicated form $a(d, s_1, s_2)$. Corollary 6 constructs a unique strategy of the indicated form $a(d, s_1, s_2)$ from $u(d, s_1, s_2)$. The claim immediately follows. $\square$

According to Corollary 7, Theorem 3 and Corollary 6 are sufficient to calculate the unique subgame perfect equilibrium, which we will do in the next section with the aid of a computer. But first, we prove an additional result that will help to reduce the runtime of our computation.

Corollary 8. Under a subgame perfect equilibrium defined by Corollary 2, if

$$-\sum_{j=1}^{n} \left(j s_1[j] \tilde{d}[j] \right) \leq \sum_{j=1}^{n} \left(u_{\text{Fold}}(d, s_2, s_1)(2 - s_1[j]) \tilde{d}[j] \right)$$

then $u(d, s_2, s_1) = u_{\text{Fold}}(d, s_2, s_1)$ and $a(d, s_2, s_1) = \text{Fold}$.

Proof. Note that

$$u(d - \langle j \rangle, s_2, s_1 + \langle j \rangle) \geq u_{\text{Fold}}(d - \langle j \rangle, s_2, s_1 + \langle j \rangle)$$
$$= -\min\{j' \text{ such that } s_2[j'] = 1 \text{ or } (s_1 + \langle j \rangle)[j'] = 1\}$$
$$\geq -\min\{j' \text{ such that } s_2[j'] = 1 \text{ or } s_1[j'] = 1\}$$
$$= u_{\text{Fold}}(d, s_1, s_2).$$

Then, we have

$$u_{\text{Hit}}(d, s_1, s_2) = -\sum_{j=1}^{n} \left(j s_1[j] \tilde{d}[j] + u(d - \langle j \rangle, s_2, s_1 + \langle j \rangle)(1 - s_1[j]) \tilde{d}[j] \right)$$
$$\leq -\sum_{j=1}^{n} \left(j s_1[j] \tilde{d}[j] + u_{\text{Fold}}(d, s_2, s_1)(1 - s_1[j]) \tilde{d}[j] \right)$$
$$\leq \sum_{j=1}^{n} \left(u_{\text{Fold}}(d, s_2, s_1)(2 - s_1[j]) \tilde{d}[j] - u_{\text{Fold}}(d, s_2, s_1)(1 - s_1[j]) \tilde{d}[j] \right)$$
$$= \sum_{j=1}^{n} \left(u_{\text{Fold}}(d, s_2, s_1) \tilde{d}[j] \right)$$
$$= u_{\text{Fold}}(d, s_2, s_1).$$

And the claim follows. $\square$

Intuitively, a player's expected payoff is at least the payoff he gets from Fold as he would otherwise not be playing a best response. Furthermore, the disutility of Fold is given by the smallest rank faceup card, which can only decrease as the round progresses, so we can bound the payoff a player can get from Hit and successfully force the other player to make a move. Corollary 8 states that if

the expected loss of pairing up from choosing Hit is too large, i.e. outweighed by the possible return from getting the other player to make a move discounted by the certain outcome from Fold, then the player should choose Fold. Note that Corollary 8 allows us to avoid having to recurse in some cases, reducing computation time.

3.2   Computational Analysis

4   Analysis of the m-Player Case

We now consider the more general case in which there are m players. As before, we first consider the single-round game, and we take the payoffs to be $-(m-1)$ times the score for the losing player and positive the score for the other players. Note that for $m = 2$, these payoffs reduce to the same payoffs we considered for the two-player case. For the general case, we can think of these payoffs corresponding to the losing player paying every other player the score that the losing player accumulated in the round, so over many rounds, the overall loser has paid each player the difference in their scores.

4.1   Analysis

As before, we assume for simplicity that whenver Fold and Hit at some configuration are expected to yield the same payoff, the players prefer Fold, and this allows us to restrict ourselves to strategies in which the action at each decision node depends only on the current configuration of the game. Note that Theorem 1 applies without modification, and the only change to the proof is that the payoff from pairing up from Hit is $-(m-1)j$ instead of j, but the same proof otherwise holds.

Although Corollary 2 is still true, we will modify it to facilitate the following analysis.

Corollary 9. Let C be the set of possible configurations. Then, any subgame perfect equilibrium can be described by a function $a : C \rightarrow \{\text{Hit}, \text{Fold}\}$. From any decision node, the expected payoffs from following the subgame perfect equilibrium given by a can be described by a function $u : C \rightarrow \mathbb{R}^m$, where $u(d, s_1, s_2, \ldots, s_m) = (p_1, p_2, \ldots, p_m)$ indicates an expected payoff of p_i for the player with the stack s_i.

Proof. Follows immediately from Theorem 1. $\qquad\square$

The only difference we made from Corollary 9 is that u is now an m-dimensional vector. In the two-player case, it was sufficient to know only the payoffs of one of the players because we could infer the payoff of the other player using the zero-sum property of the game. In the m-player case, however, we need to maintain each of the payoffs in order to get a full description of the payoffs.

With this new form of the payoffs, the analog to Theorem 3 is given by the following theorem.

Theorem 10. Under a subgame perfect equilibrium defined by Corollary 9, we have

$$u(d, s_1, s_2) = u_{\text{Fold}}(d, s_1, s_2) \text{ if } u_{\text{Fold}}(d, s_1, s_2)[1] \geq u_{\text{Hit}}(d, s_1, s_2)[1]$$
$$= u_{\text{Hit}}(d, s_1, s_2) \text{ otherwise} \tag{5}$$

where the expected payoff from Fold is

$$u_{\text{Fold}}(d, s_1, s_2) = (-(n-1)r, r, r, \ldots, r)$$
$$r = \min\{j \text{ such that } s_i[j] = 1 \text{ for some } i\} \tag{6}$$

and the expected payoff from Hit is

$$u_{\text{Hit}}(d, s_1, s_2) = \sum_{j=1}^{n} (-(n-1)j, j, j, \ldots, j) s_1[j] \tilde{d}[j]$$
$$+ u_+(d - \langle j \rangle, s_2, s_3, \ldots, s_m, s_1 + \langle j \rangle)(1 - s_1[j]) \tilde{d}[j]. \tag{7}$$

where u_+ refers to the vector cyclically shifted to the right by one element. Again, we have assumed that we never need to reshuffle. If reshuffling is necessary, then we will need to replace $d - \langle j \rangle$ in the Equation 7 with whatever deck results from reshuffling.