CS 5180 - Final Project Notes

Roger Fowler

MONTH DAY YEAR

BIG CHANGES:
switched to pettingzoo because gymnasium is not well set up to handle multiple agents switched to deep w leaning because state space proved to be infeasibly large

1 State Definitions

At the beginning of a hand, a player is dealt a number of cards. It then bets a certain number of tricks between 0 and the number of available tricks, inclusive. When a player is presented with a choice of card to play, there are a number of pieces of information they have:

1. Cards in their hand
2. Trump suit
3. Led suit
4. Cards which have been played
5. Who led the hand
6. Bet amount
7. Tricks won
8. Tricks remaining

Some of these are simply redundant; who led the hand is obvious by the number of cards in the current trick pile, and is important for the unpredictability of later players. The led suit is also obvious because it is the suit of the first card in the trick pile, and is not defined if there is no pile. Tricks remaining is equal to the number of cards in hand.

Bet amount and tricks won can be combined into the number of tricks that must still be won to make the bet. In this case the agent can only decrease this number and will aim for zero. But decreasing it past zero will incur a penalty at scoring time.

A naive treatment of knowledge about specific cards would produce a massive state space. The 60 card deck is distributed among several groups; with some number of known cards in hand, a known card flipped to reveal trump suit, known cards that have been played, and the rest of the unknown cards distributed between the remaining undealt deck and the other players' hands. In fact, there is additional information here; a player may not know which specific cards are in another player's hand, but because a player can be forced to play the led suit, if they do not do so implies that they have no cards of that suit.

Ultimately the game allows for much simplification. Suits are fungible, except that one is the trump suit for the hand and one (possibly the same) is the led suit for the trick. Additionally, cards themselves are fungible. If a player has a set of cards in their hand of a single suit, and knows that there are no cards in other players' hands of the same suit and between the ranks of the set, then they are all functionally the same card because they have the same chance of

winning in a particular situation. Once card values have merged in this way they will never unmerge.

For any game state, since the player knows much about which cards are where or are likely to be where, a card-counting agent can calculate exactly how many cards could appear that would beat a given card. It can then abstract the large amount of information it has into a value for each card, and to combine those cards into groups of cards with the same value. Ultimately, this leads to a representation of the state which is much simplified compared to all of the information the agent actually has. An agent's hidden state is composed of all of the information about exact cards and where they can be. But the agent only needs to know, in the moment, a simplified representation of its hand and how many tricks it must win in the remaining hand, and it will use this simplified state to evaluate its value function. TODO THIS IS UNCLEAR

State transitions become simplified in this representation. A state is composed of a hand of N cards and a goal of T remaining tricks to win. After an action is taken, the rest of the agents take their actions, and the agent arrives at a new state. This state will have exactly N-1 cards, and the agent is in control of which card was removed. However, it is generally not in control of whether T has remained the same or decreased by exactly one. Eventually this leads to a state where there are no cards in hand. In these states, any action (which can be restricted to a single null action) gives a reward and leads to a final terminal state. This reward is only positive when the bet was achieved (T=0) and increasingly negative otherwise. Also, since the agent has no control over the length of the game it does not make sense to discount rewards.

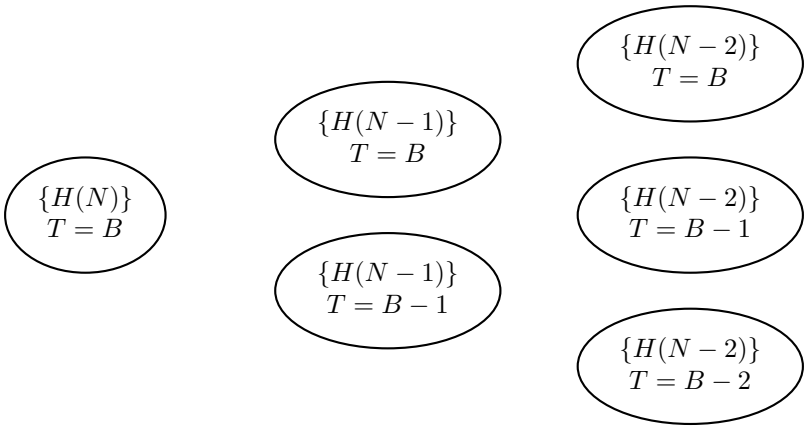

Figure 1: State Diagram for an Agent during a Hand

1)a final state representation

T, sum(T), hand, pile, whether there is a trump suit (because first suit in hand is always trump suit), whether the led suit is the trump suit (as otherwise it would be the second suit, and the case where there is no led suit should be obvious from the pile)

padding this vector with -1 so it is always the same length is good for consistent indexing and convolution

possible hand representations:

- list of card indices - no consistent convolution-friendly stuff, but this is the format that the env should give as an observation

- binary list of whether card is in hand
- binary list of whether card is in hand, sorted by wizard-trump-led-other1-other2-jester suit
- binary list of whether card is in hand, sorted, with -1 for cards that have appeared - includes info about game state beyond your own hand
- list of card values evaluated by full card counting

Therefore:

- T
- sum(T) across players
- number of players
- trump suit flag
- led suit = trump suit flag
- card state (padded, sorted) = $\begin{cases} 2 & \text{in pile} \\ 1 & \text{in hand} \\ 0 & \text{unseen} \\ -1 & \text{in previous piles} \end{cases}$

2 Action Definitions

The action space is large, but the permissible actions in any given state are small. However many groups of cards are in an agent's hand are the number of actions available to it.

actions:

Index	Action
[0, 59]	place card at index i
[60, 80]	bet $i - 60$
[81, 84]	choose suit for trump
[85]	null action used for receiving reward

Table 1: Action Space Definition

3 State Transition

Starting with the trick leader we proceed around the table comparing cards. If a card is better than the current winner, that card becomes the winner. The possibilities are encoded in Table 2.

Legend	
Y	New Card Wins
X	Current Winner Remains
>	Compare Card Values
N/A	Impossible

New Card \ Current Winner	Wizard	Trump	Led	Other	Jester
Wizard	>	>	>	N/A	>
Trump	>	>	Y	N/A	>
Led	>	X	>	N/A	>
Other	>	X	X	N/A	>
Jester	>	>	>	N/A	>

Table 2: Trick Winner Logic

4 Training Procedure

Hands are independent of one another. The deck is shuffled between hands, so there is no connection in state, and total score is simply a sum of score from each hand. This allows a hand to be played in isolation without having to consider the rest of the game.

Because rewards only occur at the end of a hand, it is reasonable to start training closest to the reward. That is, starting with a hand with a low number of tricks is easier to learn. This also reduces the number of actions an agent can take in each trick, because it has fewer cards in hand. The last tricks of a large hand are similar but not exactly the same as the tricks in a small hand, because the possible number of bet tricks may be very high, and also because the agent has much more knowledge of what has and has not been played. Still, the knowledge should be generalizable, so short hands should be trained before long hands.

Once the agent is able to play for an arbitrary choice of hand and bet, it should be possible to train the agent to choose its own bet as well. These special actions are reserved for only this phase of play. In reality since this stage of play is completely divorced this choice could be handed off to a different specialized agent, except that this agent would need access to the expected rewards of starting each hand with a given bet. Actions are already restricted by play, so it is not a serious complication to have special actions reserved for only this situation. This also does not require any additional state information.

There is one more special state; the dealer turns over the top card of the deck to set the trump suit after dealing the hand. If this card is a Wizard, the dealer is able to choose the trump suit. Again, this creates four actions that only occur in this rare situation. And again, access to the state value function is useful to train this, and no additional state information is necessary, so it can be rolled into the larger agent. Since this choice appears before the choice of bet and before play, it should be trained last. During training this choice should be taken randomly.