Winning the Trick as Defender in Contract Bridge Card Game Using Greedy Algorithm

Vincent Endrahadi - 13515117
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13515117@std.stei.itb.ac.id ; vincentendrahadi01@gmail.com

***Abstract*—Card game is one of the most played game in the world. Even after the globalization, card games are still being played by people all over the world. One of the card game variants is the trick-taking game. A trick in trick-taking game is a group of card played by players in turn in a round. Then, Trick-taking card game also consists of several variants of game. One of the most popular trick-taking game variant is Contract Bridge. Contract Bridge game is divided into two main part which is auction and playing. In the playing section, players and his partner have to win tricks according to the number of bid they made in auction or prevent other player to win respective amount of tricks. Greedy algorithm can be used to provide optimal solution in each trick to help the defender win the trick.**

***Keywords*—greedy, contract bridge, optimal, trick.**

I. Introduction

Playing is one of the most people usually do in their free time especially by younger generation. Although some of the older generation still playing games. Games consists of many variation of rules and styles that it is impossible for us to play it all. One of games that are played during free times is card games.

It is said that card games was first invented in Imperial China, appeared first in 9th century Tang China. The card games then spread to India and Persia where spread again into Egypt. From Egypt, the card games spread to Europe through Italy and Iberian Peninsulas in the 14th century.

Nowadays, there are many suits format of card in the world. French suits is the most popular one and can be found on most of the countries in the world. French suits consist of spade, heart, club, and diamond. Each suit consists of 10 number card and 3 face cards which is jack, queen and king.

There is a number of card games style of play all across the world. One of the most popular is trick-taking game. A trick is group of card that are played by players in turn. Trick-taking game essential goal is winning tricks according to the number of contract the players made in auction or bidding section. Of course, the number of tricks needed will be different according to the rules of the game. The style of card suits used in trick-taking game is French's suit.

Contract bridge or simply Bridge is the most popular style of trick-taking game. Contract bridge is played by millions of people in club, tournaments, online or even with

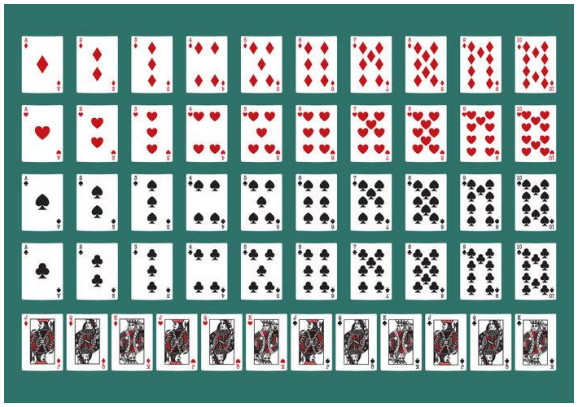

Figure 1 French suit cards.
Source:
https://static.vecteezy.com/system/resources/previews/000/099/080/non_2x/vector-poker-cards.jpg

friends at home. The governing body for international competition is World Bridge Federation.

In Contract Bridge, defender is players who lose the auction. The goal of the defender is to prevent winner and his partner to win particular number of tricks. Therefore, to win tricks as defender, greedy algorithm can be used to provide good solution.

II. Basic Theories

1. Bridge (Contract Bridge)
   - Overview

     Bridge is played by four players sitting together a table using 52 standard deck (French suit). Each player will sit opposite to his partner. Each player will be given 13 cards in their hand. Game consists of 2

Figure 2 In Hand Cards
source :
https://www.bridgehands.com/A/Arranging_Suit_Sort.jpg

stages which is auction and play.

A trick will be started when a player lead the trick (plays the first card). Leader of the first trick is determined by the auction (player who win the auction). Then, for the following trick, the leader will be players who win the preceding trick.

Each player only allowed to play one card each trick in a clockwise order. Player must played card with the same suit as the lead played. If they have none, they can play any card. The situation where the player do not have card that have the same suit as the lead card is called void. The player who played the highest ranked card win the trick.

In a game, there can be a trump suit which is decided in the auction section. Trump suit is the highest rank suit in that round. However, there can be situation where there is no trump suit which is called NT (No Trump).

- Auction

  The dealer will start the auction. Players in the auction able to bid or pass. The auction proceeds in clockwise order. If a player pass in his first bid, he still can join the bidding later. A bid is consist of level of contract and the trump suit (include NT). Each bid must be higher than the preceding bid. A bid is higher than the preceding bid if the level is higher or the denomination is higher. The denomination in ascending order is ♣, ♦, ♥, ♠, and NT (no trump).

  A player is said to win the auction if his bid is the highest and the other player did not bid anymore (pass). The trump suit for the game is the one that won the auction. Player have to win a number of tricks according to his bids in excess of six.

  For example, if A won the auction with 2 Spades, A and his partner have to win 2+6 = 8 trick with spades as trump suit.

- Play

  The player from the winning side that first bid the suit in the final bid becomes declarer. Player left to the declarer become the lead in the first trick. Partner of the declarer is called dummy.

  After the lead play his card, dummy will lay his card face up in the table. Plays proceed in clockwise order. Player have to follow the led suit if possible. Dummy player will not able to play as his card will be played by declarer. However, usually the declarer request his partner to play particular card.

  The highest ranked card will win the trick. Next trick will be led by player who won the previous trick. The defenders is players who lost the auction. Their goal is to prevent the declarer and his partner to win particular number of tricks.

2. Greedy Algorithm

Greedy algorithm is one of the most popular method in solving optimization problem. Solving optimization problem can be done by searching optimum problem for the given problem.

Greedy algorithm is an algorithm paradigm that choosing the locally optimal choice step by step. The primary concept of greedy algorithm is choosing the most optimal solution in each step hoping to find globally optimal solution.

Greedy algorithm choose the most optimal solution in each step. However, this way of choosing may not lead to the most optimum solution.

There are many type of problem that can be solved using greedy algorithm. Here is a list of few that use greedy approach:

- Travelling Salesman Problem
- Prim' Minimal Spanning Tree Algorithm
- Kruskal's Minimal Spanning Tree Algorithm
- Dijkstra's Minimal Spanning Tree Algorithm
- Graph – Map Coloring
- Knapsack Problem
- Counting Coins
- Etc.

Greedy algorithm consist of several elements, which is:

1. List of Candidate

   List of candidate solution that we can choose from in order to find optimal solution.
2. List of Solution

   Empty list at first. List of solution that are taken from list of candidate in order to find optimal solution.
3. Selection Function

   Function to select the most optimal choice from list of candidate in that step. Function to take the most locally optimal solution.
4. Checking Function

   Function that are used to make sure that the solution selected from selection function are inside the given boundary.
5. Objective Function

   Function to check the list of solution is the optimal solution for the problem.

From the elements above, the step of general greedy algorithm is:

1. Initialize S (list of solution).
2. Use selection function to select from list of candidate.

3. Check if the solution given by selection function union with list of solution are still acceptable.
4. If yes, delete selected element from list of candidate and add it to list of solution.
5. If no, iterate through next element of list of candidate.
6. After finished iterating, check if the solution can be accepted, if no, then there is no solution for this problem using greedy algorithm.

For example, in counting coins problems, the problem given is to count to a given problem with least possible coins. Using the greedy approach, take the largest possible coins. Given coins of 1, 3, 4, 10. Find the least number of coins that sum up to 18.

From the problem above, we know that:

1. List of candidate consists of 1,3,4,10
2. Selection function is to find the largest possible coin from list of candidate.
3. Checking function is to check that the sum of given solution are not greater than 18.
4. Objective function is check that the sum of given solution is 18.

Then from the points above, the greedy procedure will be:

1. Take the 10 coin, Remaining count = 8.
2. Take the 4 coin, remaining count = 3.
3. Take the 3 coin, remaining count = 1.
4. Take the 1 coin, remaining count = 0.

For this problem, the results that we get using greedy algorithm is 4 coins. This result is the most optimal result for this problem. However, if we change the problem a little bit, the result that we get using greedy algorithm may not be the most optimal.

If we change the provided coins into 1, 9, and 10. With greedy approach will result in 10 + 1 + 1 +1 +1 +1 +1 +1 +1, which is 9 coins. While the optimal solution is 9 + 9.

Thus, using greedy algorithm for optimization problems, will not always produce the most optimal solution. However, the solution will be enough to be used as approximation of the most optimal solution as usually algorithm that can give optimal problem needed a lot of time or memory space.

III. PROBLEM ANALYSIS

3.1. Defender in Bridge

Defenders consists of two player that are partner. These players lose the auction, therefore, they have to prevent the winning side of auction in completing their goal. In order to prevent opposing side to reach particular number of tricks, defenders have to win some number of tricks.

3.2. Using Greedy Algorithm Concept to Choose Card to Play

Greedy Algorithm can be used in order to win tricks. Concept of "take what you can get now" can help decision making in winning tricks. To choose the best card to play, there several conditions that make a card is the best card to play at the moment:

1. If there is ace (A) in your hand that had the same suit with let suit, choose ace.
2. If there is no ace (A) in your hand that had the same suit with led suit, check if the remaining card is the highest (card higher than it had been played in the rounds before).
3. If no, select the smallest ranked card with the same suit with led suit.
4. If partner are winning the trick, choose the smallest possible.
5. If you are not the lead and in void situation, select the smallest number of trump to play.
6. Else, choose the smallest number possible to play.

3.3. Greedy Algorithm Usage in Bridge

From the points above, we can see that using greedy algorithm can give solution to win tricks. Using the concept above, we have greedy elements which is:

1. List of Candidate
   List of candidate will consists card available to play which is the 13 cards that are in hand. The number and the suit will be saved for further usage. The number will decrease after each turn.
2. List Of Solution
   List of Solution will consists of card that will be played. But this list will not be effective as we cannot predict what kind of card be played next. Therefore, this list will saved the number of tricks we win. So that we can checked if the number of enemy tricks did not fulfill their promises.
3. Selection Function
   This function will be the main function to select which card are the best to play at the moment. This function will pick card according to situation, for example, if list of candidate have ace that can be played, this function will select the ace.
4. Checking function.
   This function will check if there is still card to play and is it our turn to play card.
5. Objective Function
   This function will be use to check if the solution we use is the most optimal one whether we manage to prevent enemy fulfill their contract or not.

From these element above, we can generate the overall of greedy algorithm that will can used to win tricks:

1. Initialized the list of solution, because list of solution only contain number of tricks we win, it will only be an integer.
2. Check if there are still cards in our hand and it is our turn to play card.
3. Use selection function, to select the best card to play at the moment.
4. Update the solution.

Implementation in pseudo-code:

```
Procedure PlayCard()
{
    //I.S : Player have 13 cards in hand,
            and player is the defender.
    //F.S : Player have 0 cards in hand,
            and a number of tricks won.
    VARIABLE
    n = 0;
    //Number of tricks won

    ALGORITHM

    while(CheckCard())
    {
        while(not(Turn()))
        {

        }
        SelectCard();
        if(WinTricks)
        {
            n = n + 1;
        }
    }
}
```

Here are the explanation of function used in pseudo-code above:

1. CheckCard
   Function to check if there still cards in hand.
2. Turn
   Function to check if it is our turn.
3. SelectCard
   Function that will return card that will be played. This function will check condition that currently in then choose the best card.
4. WinTrick
   Function to check if we win the tricks. If yes, increase the number of tricks we won.

V. IMPELEMENTATION IN GAMES

We sit in south area.
S stands for spade, c stands for club, d stands for diamond, and h stands for heart.
Suppose we have cards in our hands:
Spade : J , 10 , 6, 4
Heart : 3
Diamond : 10, 8, 6, 4, 2
Club : 10, 6, 2

The winning bid is 1 Heart by West, it means that enemy have to won 7 tricks to fulfil their promises and we have to win 7 trick to prevent them winning 7 tricks.

Round 1:
North: K d
East: 5 d
South: 2 d
West: A d win

J s	Not suit
10 s	Not suit
6 s	Not suit
4 s	Not suit
3 h	Not suit
10 d	Smaller
8 d	Smaller
6 d	Smaller
4 d	Smaller
2 d	The smallest, choose
10 c	Not suit
6 c	Not suit
2 c	Not suit

Round 2:
West: 7 h win
North: 6 h
East: 2 h
South: 2 d

J s	Not suit
10 s	Not suit
6 s	Not suit
4 s	Not suit
3 h	The only one, choose
10 d	Not suit
8 d	Not suit
6 d	Not suit
4 d	Not suit
10 c	Not suit
6 c	Not suit
2 c	Not suit

Round 3:
West: A h win
North: 3 d
East: 4 h
South: 2 c

J s	Not suit
10 s	Not suit
6 s	Not suit
4 s	Not suit
10 d	Not suit
8 d	Not suit
6 d	Not suit
4 d	Not suit
10 c	Not suit
6 c	Not suit

2 c	Void, choose the smallest

Round 4:
West: K h win
North: 2 s
East: 5 h
South: 4 d

J s	Not suit
10 s	Not suit
6 s	Not suit
4 s	Not suit
10 d	Not suit
8 d	Not suit
6 d	Not suit
4 d	Void, the smallest
10 c	Not suit
6 c	Not suit

Round 5:
West: 9 h win
North: 7 d
East: 8 h
South: 4 s

J s	Not suit
10 s	Not suit
6 s	Not suit
4 s	Void, the smallest
10 d	Not suit
8 d	Not suit
6 d	Not suit
10 c	Not suit
6 c	Not suit

Round 6:
West: 10 c
North: Q c win
East: 5 c
South: 6 c

J s	Not suit
10 s	Not suit
6 s	Not suit
10 d	Not suit
8 d	Not suit
6 d	Not suit
10 c	Partner already winning
6 c	Partner already winning, choose

Round 7:
North: Q d win
East: j d
South: 6 d
West: 3 s

J s	Not suit
10 s	Not suit
6 s	Not suit
10 d	Partner winning
8 d	Partner winning
6 d	Partner winning, choose
10 c	Not suit

Round 8:
North: 9 d
East: Q h win
South: 8 d
West: 7 s

J s	Not suit
10 s	Not suit
6 s	Not suit
10 d	No higher
8 d	No higher, choose
10 c	Not suit

Round 9:
East: J h win
South: 6 s
West: 8 s
North: 7 c

J s	Not suit
10 s	Not suit
6 s	Void, choose smallest
10 d	Not suit
10 c	Not suit

Round 10:
East: 10 h win
South: 10 c
West: 4 c
North: 5 s

J s	Not suit
10 s	Not suit
10 d	Not suit
10 c	Void, choose smallest

Round 11:
East: A c win
South: 10 d
West: 8 c
North: 9 c

J s	Not suit
10 s	Not suit
10 d	Void, choose smallest

Round 12:
East: Q s
South: 10 s
West: 9 s
North: K s win

J s	Cannot win
10 s	Choose smallest

Round 13:
North: J c
East: A s

South: J s
West: K c win

J s	Choose

From the game above, we can see that the enemy got 10 tricks while we got 3 tricks, therefore, we failed to prevent enemy to fulfill their promise.

The result above is the most optimal solution given our hand cards. This show bridge game is not just about playing. If the bidding are not carry out the right way, no matter how optimal the play is, it is still difficult to prevent the enemy win.

V. Conclusion

It is clear that greedy algorithm can provide some optimal solution given cards in hand. However, there exists some variation that greedy cannot give optimal problems because it included heuristics where we have to save our cards in order to create streak win so that the enemy will not have opportunity to win the trick. This kind of situation cannot be solved by greedy algorithm. Other than that, the cards given was also a factor that affecting the game, if the given card was not good enough, even with the most optimal solution we cannot prevent the enemy to fulfill their contract. Hence, using greedy algorithm as a defender should be give some good results.

VII. Acknowledgment

The author would like to express his gratitude to the God for giving the author the power to finish this paper, to his parents for giving him many support mentally and physically. The author also wanted to thanks Dr. Ir. Rinaldi Munir as the lecturer of Algorithm Strategy who give me teachings and insight to complete this paper. Finally, the author wanted to thanks my friends at KMB ITB that give me knowledge and teaching me to play bridge which inspire me to write this paper.

References

[1] http://www.wopc.co.uk/history/.
Accessed at 17 May 2017.

[2] https://boardgamegeek.com/boardgamemechanic/2009/trick-taking
Accessed at 17 May 2017.

[3] Munir, Rinaldi. 2009. Strategi Algoritma Bandung: Penerbit Informatika, Palasari.

[4] https://en.wikipedia.org/wiki/Contract_bridge
Accessed at 16 May 2017

[5] https://en.wikipedia.org/wiki/Trick-taking_game
Accessed at 16 May 2017

[6] https://www.tutorialspoint.com/data_structures_algorithms/greedy_algorithms.htm
Accessed at 17 May 2017

[7] http://www.arkadium.com//games/bridge/
Accesed at 17 May 2017, for experiment usage.

Pernyataan

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 18 Mei 2016

Vincent Endrahadi - 13515117