Markov Chains and Aeroplane Chess

Lingyu Tan

Contents

1 Background

Aeroplane Chess is a Chinese cross-and-circle board game which is based on the Western game of Ludo and the Indian game of Pachisi. It is referred to as *feixing qi* in Chinese.[1]

Developed in the 20th century, aeroplane chess quickly grew in popularity because of children's fascination with flight and its relative activities.[2] Until now, aeroplane chess has been quite popular among most young people especially those primary school students, even some adults will choose to play aeroplane chess in parties.

A key component of the game is imagination, as players must think like pilots to strategize their next move, and thus, most parents in China believe that it is an exciting puzzle game that can help to develop children's intelligence. Therefore, they would like to buy an aeroplane chess for their kids. According to one of the most famous Chinese shopping websites, Tmall, we find that the highest monthly sales of aeroplane chess among stores are over 30,000.[3]

2 Simplified Rules Description

Players try to get their own plane pieces from their hangars, located at the corners of the board, into the base of their own colour in the centre of the board. To simplify the procedure, each player has only one plane and takes a turn by rolling the die.[1] On a turn a player may do the following:

- Take the plane of the hangar onto the board (state 0). This can only be done by rolling a 6. You can do nothing if your plane has not taken off.
- There are totally 56 squares to go through for a plane.
- Move your plane that is on the board clockwise around the track, the number of spaces indicated by the die.
- A plane must fly into the centre base on an exact roll. Otherwise, the player should go forward until reaching the centre base and then go backwards for redundant steps. For example, a player rolls a 5 but there are only 3 steps remained to the centre base, so the player should go forward for 3 steps and then go backwards for 2 steps.
- The first player to get his/her plane to the centre of the board wins.

Additional rule (to be added in sensitivity analysis):

- There are additional *shortcut* squares (dotted air line in the chessboard). When a plane lands on one of these of its own colour, it may take the shortcut, that is, jumping from state 18 directly to state 30.

Figure 1 below is a chessboard sample:

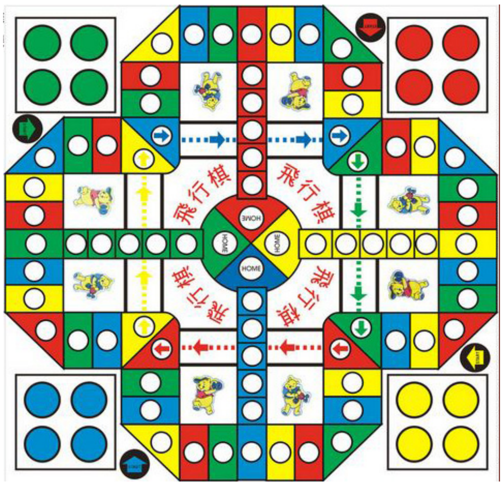

Figure 1: Chessboard Sample

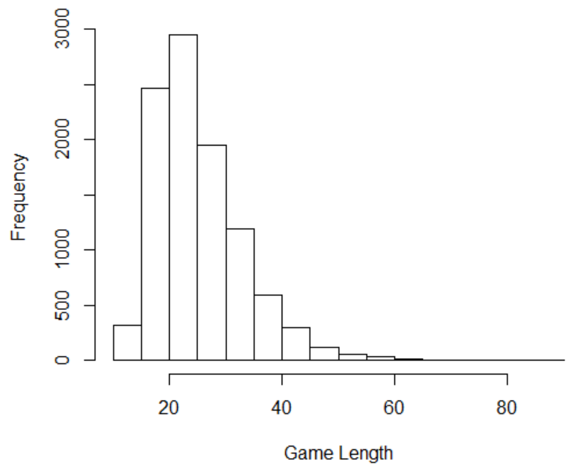

Figure 2: Histogram of game length for single player

3 Simulation for Single Player

Considering the case where only one player plays the game, after simulating 10,000 times, here are the statistical summary and histogram (See Figure 2 above):

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
12.00	20.00	24.00	25.75	30.00	86.00

Comments:

From the table, we know that a single player may need to take an average of 25.75 turns to complete the game. Among 10,000 simulations, the minimum steps is 12 and the maximum is 86. The minimum possible steps a player may take to complete the game is 10 steps, for example, roll nine successive 6 following a single 2 to exactly enter the centre. However, the probability for that is so small that we even did not manage to do that in 10,000 simulations.

From the figure, we can see that the distribution is basically right skewed with most of the lengths lying in [15, 35] and a peak just over 20.

Besides, at the very end of the game, if we had not had to get an exact roll to enter the centre, the excepted steps would be $57/3.5 = 16.3$. Combined with the fact that our mean steps is 25.75, we know that we have wasted a lot of steps keeping going forwards and backwards near the end.

4 Simulation for Multiple Players

Considering the case where there are multiple players playing the game, after simulating 10,000 times for 2, 3, 4 players respectively, the statistical summary and histograms are shown below:

Player no.	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
2	12.00	18.00	21.00	21.56	24.00	51.00
3	11.00	17.00	19.00	19.86	22.00	42.00
4	11.00	17.00	19.00	18.89	21.00	37.00

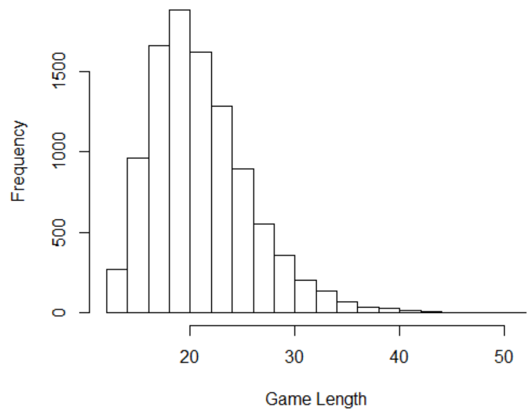

Figure 3: Game length for two players

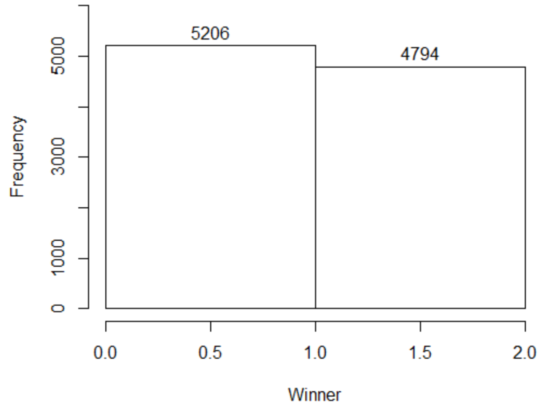

Figure 4: Winners for two players

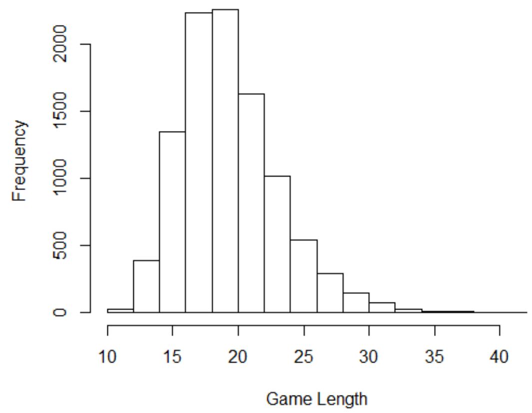

Figure 5: Game length for three players

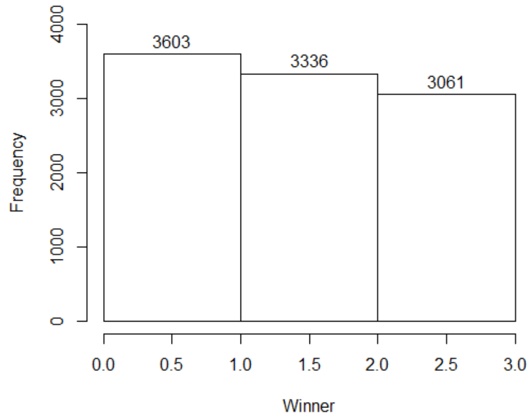

Figure 6: Winners for three players

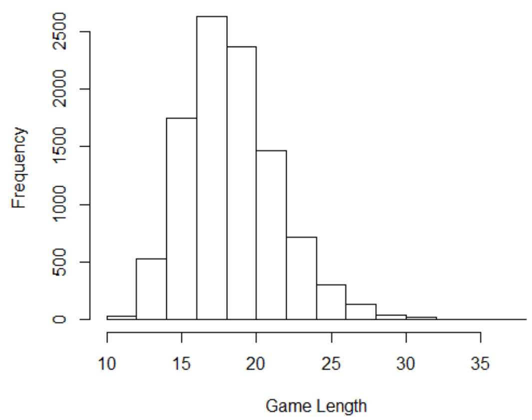

Figure 7: Game length for four players

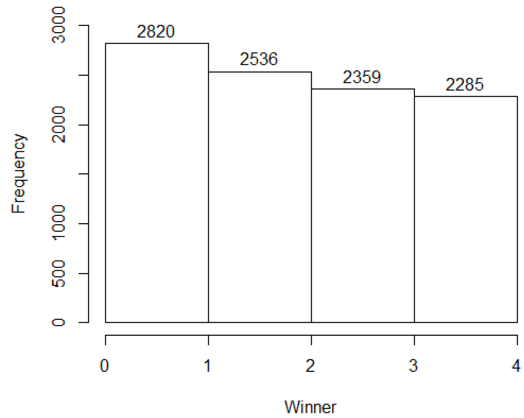

Figure 8: Winners for four players

Comments:

From the table, generally all the summary statistical data decrease when the number of the players increase, which means the more players there are, the less number of turns required for a game.

From Figure 3, 5 & 7, which represent the cases where there are 2, 3 or 4 players respectively, we can see that the general shape of the distribution does not change; however, the mode of the distribution moves towards left gradually, showing the same trend as the table.

Besides, there seems to be a go-first advantage. From Figure 4, 6 & 8 indicating the winners when there are 2, 3 or 4 players respectively, we can see that the player who plays first is more likely to win the game. Also, the more behind a player ranks, the less likely he will be to win.

5 Sensitivity Analysis

Case 1: We add the additional rule, then redo simulations for 4 players. We have the following statistical summary table:

Shortcut	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
FALSE	11.00	17.00	19.00	18.88	21.00	37.00
TRUE	9.00	15.00	17.00	17.73	20.00	35.00

Case 2: We change the total squares number from 56 to 60 to see what will happen. Here is the statistical summary table:

Square Num	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
56	11.00	17.00	19.00	18.88	21.00	37.00
60	12.00	18.00	20.00	19.94	22.00	37.00
64	13.00	19.00	21.00	21.08	23.00	42.00

Case 3: We use a 4-sided die instead of 6-sided, and here is the summary table:

Die sides	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
6	11.00	17.00	19.00	18.88	21.00	37.00
4	16.00	22.00	24.00	24.32	26.00	41.00

Comments:

- Case 1: After adding a shortcut, the length of the game generally decreases because the plane can get to the centre more quickly using the shortcut.
- Case 2: After adding additional squares to the board, the length of the game basically increases because the plane have more squares to go through.
- Case 3: After changing the sides of die from 6 to 4, the length of the game obviously increases since the plane moves "more slowly". And the length of the game is quite sensitive to the die sides.

References

[1] Wikipedia contributors. (2019, November 9). Aeroplane chess. In Wikipedia, The Free Encyclopedia. Retrieved 18:38, November 17, 2019, from https://en.wikipedia.org/w/index.php?title=Aeroplane_chess&oldid=925304457

[2] Asiapac Editorial. (2012). *Gateway to old school games*. Singapore: Asiapac Books. pp. 65-67. Available at https://books.google.com/books?id=0_kAgAAQBAJ&pg=PA65

[3] Tmall. https://list.tmall.com/search_product.htm?q=%B7%C9%D0%D0%C6%E5&type=p&spm=a220o.0.a2227oh.d100&from=.detail.pc_1_searchbutton

A Appendix - Code Used for Single Player

```
#Aeroplane Chess Single Player

N = 57 #total steps

shortcut = F #if additional rule is applied

#Set up P matrix
P=matrix(0, nrow=N, ncol=N, byrow=T)

for (i in 1 : (N-6))
{
  for (j in 1 : 6)
  {
    P[i, i+j] = 1/6
  }
}

for (i in c(N-5, N-2))
{
  P[i, N-4] = 1/6
  P[i, N-3] = 1/6
  P[i, N-2] = 1/6
  P[i, N-1] = 1/3
  P[i, N] = 1/6
}

for (i in c(N-4, N-3))
{
  P[i, N-3] = 1/6
  P[i, N-2] = 1/3
  P[i, N-1] = 1/3
  P[i, N] = 1/6
}

for (j in (52:N))
{
  P[N-1, j] = 1/6
}
P[N, N] = 1

if (shortcut == T)
{
  P[13:18, 19] = 0/6
  P[13:18, 31] = 1/6
}

s=c(1:N) #possible states
gamelength=vector()

for (i in 1:10000)
{
  takeoff = 0
  current = vector()
  gameon = 1
```

```
  turns = 0

  #if not taken off, do nothing
  while (takeoff == 0)
  {
    turns = turns + 1
    if(sample(1:6, 1) == 6)
    {
      takeoff = 1
      current[turns] = 1
    }
  }

  #start moving
  while (gameon == 1 && takeoff == 1)
  {
    current[turns+1]=sample(s,1,replace=TRUE,prob=P[current[turns],])
    if (current[turns+1] == N)
    {
      gameon = 0
    }
    turns = turns+1
  }
  gamelength[i] = turns - 1
}

summary(gamelength)
hist(gamelength, xlab = "Game Length", main = "")
```

B Appendix - Code Used for Multiple Players

```
#Aeroplane Chess Multiplayer
#Would be much easier by directly simulation than using Markov chain

N = 57 #total steps - Sensitivity Analysis

num = 4 #number of players

shortcut = F #if additional rule is applied - Sensitivity Analysis

#----------Set up P matrix----------#
P=matrix(0, nrow=N, ncol=N, byrow=T)

for (i in 1 : (N-6))
{
  for (j in 1 : 6)
  {
    P[i, i+j] = 1/6
  }
}

for (i in c(N-5, N-2))
{
  P[i, N-4] = 1/6
  P[i, N-3] = 1/6
  P[i, N-2] = 1/6
  P[i, N-1] = 1/3
  P[i, N] = 1/6
}

for (i in c(N-4, N-3))
{
  P[i, N-3] = 1/6
  P[i, N-2] = 1/3
  P[i, N-1] = 1/3
  P[i, N] = 1/6
}

for (j in (N-5):N)
{
  P[N-1, j] = 1/6
}
P[N, N] = 1

if (shortcut == T)
{
  P[13:18, 19] = 0/6
  P[13:18, 31] = 1/6
}
#----------------Done----------------#

s=c(1:N) #possible states
gamelength=matrix(0, nrow = num, ncol = 10000, byrow = T)
takeoff = vector("numeric", 4)
gameon = vector("numeric", 4)
```

```r
turns = vector("numeric", 4)

for (i in 1:10000)
{
  current = matrix(0, nrow = num, ncol = 100, byrow = T)
  for (no in 1:num)
  {
    takeoff[no] = 0
    gameon[no] = 1
    turns[no] = 0

    #if not taken off, do nothing
    while (takeoff[no] == 0)
    {
      turns[no] = turns[no] + 1
      if(sample(1:6, 1) == 6)
      {
        takeoff[no] = 1
        current[no, (turns[no])] = 1
      }
    }

    #start moving
    while (gameon[no] == 1 && takeoff[no] == 1)
    {
      current[no, turns[no]+1]=sample(s,1,replace=TRUE,prob=P[current[
    no, turns[no]],])
      if (current[no, turns[no]+1] == N)
      {
        gameon[no] = 0
      }
      turns[no] = turns[no] + 1
    }
    gamelength[no, i] = turns[no] - 1
  }
}

#Find the minimum step among players
Gamelength = apply(gamelength, 2, min)

winner = vector()
for (i in 1 : 10000)
{
  winner[i] = match(Gamelength[i], gamelength[, i])
}

#histogram of length
summary(Gamelength)
hist(Gamelength, xlab = "Game Length", main = "")

#histogram of winners
h <- hist(winner, breaks = 0:num, xlab = "Winner", main = "", ylim = c
    (0, 12000/num))
text(h$mids,h$counts,labels=h$counts, adj=c(0.5, -0.5))
```

C Appendix - Transition Matrix Using 4-sided Die

```r
#Aeroplane Chess Multiplayer
#Transition Matrix when using 4-sided die

#--------How to use-------#
#1. Run this whole file to replace P matrix in Project_Multi.R
#2. Run Project_Multi.R from Line 50 (except P matrix part)
#-----------END-----------#

N = 57 #total steps - Sensitivity Analysis

num = 4 #number of players

shortcut = F #if additional rule is applied - Sensitivity Analysis

#-----------Set up P matrix-----------#
P=matrix(0, nrow=N, ncol=N, byrow=T)

for (i in 1 : (N-4))
{
  for (j in 1 : 4)
  {
    P[i, i+j] = 1/4
  }
}


for (i in c(N-3, N-2))
{
  P[i, N-2] = 1/4
  P[i, N-1] = 1/2
  P[i, N] = 1/4
}

P[N-1, N-3] = 1/4
P[N-1, N-2] = 1/4
P[N-1, N-1] = 1/4
P[N-1, N] = 1/4

P[N, N] = 1

if (shortcut == T)
{
  P[13:18, 19] = 0/6
  P[13:18, 31] = 1/6
}
#------------------Done------------------#
```