OR 762 Final Project
Spring 2017
Computer Simulation Techniques
Professor Yunan Liu
April 30, 2017

NBA Games Simulator

Shuai Yang and Yang Zhao

1 Introduction

National Basketball Association (NBA), one of world's largest sports leagues, has countless fans and amazing international influence. NBA players are the world's best paid athletes by average annual salary per player. As its market value keeps growing and a large amount of money is involved in the NBA circle, not surprisingly, a lot of studies have been developed recently to predict results, to analyze players' performances or to assist coaches with corresponding data and statistics.

Despite randomness of sports games and lasting doubt about the importance of data analysis, NBA is nevertheless undergoing its analytics revolution. Unlike the old-fashioned way of measuring how well a player or a team would perform which relies on professionals' suggestions, now coaches and staff pay more attention to statistics, figures and formulas to find the most effective methods for winning. Recently, a lot of strategic changes in NBA such as three-point obsession and obsolescence of traditional big center are encouraged by data foundation. The detailed NBA statistics provides us a clear player-by-player or team-by-team comparison and a visual method to track previously intangible offensive or defensive effects of certain low-key players. That is why players like Draymond Green, Rajon Rondo and Shane Battier, who have obvious drawbacks, have received praise for their adaptability and versatility. Their indispensabilities are hard to gauge before the new analytics' era of NBA.

Fortunately, meticulous data related to players and teams are recorded and available to download. With the assistance of such statistics, we design a game simulator to predict 16-17 NBA Play-off season results. The underlying formulas and parameters in our simulator can be adjusted so that it agrees with the characteristics of two simulated teams. We hope that simulator not only helps predict the game result but also offers a method to investigate the validity of given tactics. A game is split up into two sides' back-and-forth possessions so that each possession is simulated based on the regular season statistics this year.

2 Simulation Process

2.1 Data Preparation

The necessary statistics in our simulator can be obtained from on-line databases. This website(http://www.basketball-reference.com/) provides various and comprehensive statistics of NBA games, not just for per game, but also for per 100 possession, per 36 minutes, etc.

The simulation statistics are all based on the 2016-2017 NBA regular season records. Such statistics are divided into two categories: team statistics and player statistics. In terms of the team statistics, pace and turnover are adopted.

- *Pace*— estimate for the number of possessions per 48 minutes by one team.
- *Turnover*—average number of turnovers per game for one team in the regular season.

It is worth noting that there are another two team statistics offensive rebounds and defensive rebounds(ORB and DRB). However, these two statistics that represent total number of offensive/defensive rebounds of all players in one team per 100 possessions, are not obtained directly from the website. Instead, We add up five players offensive/defensive rebounds per 100 team possessions in the field to help decide the owner of a rebound later. For player statistics, we collected 8 different statistics for each player:

- *Field goal attempts per 100 team possessions*

  This statistic is used to characterize the role of a player in his team. The star player does more attempts than a role player. Therefore, the ratio of field goal attempts of one player to the total field goal attempts of all players in the field is applied into estimating the chance that this player will shoot in one possession.

- *2-point attempts per 100 team possessions*

  This statistic is considered to be the preference for a player to shoot 2-point.

- *3-point attempts per 100 team possessions*

  This statistic is considered together with 2-point attempts per 100 team possessions. The proportion of 3-point attempts per 100 team possessions to 2-point attempts per 100 team possessions is used to decide whether a player has a 3-point or 2-point shot in one possession.

- *2-point field goal percentage*

  To predict whether one players 2-point attempt scores or not, the regular season average 2-point field goal percentage gives an basic measurement.

- *3-point field goal percentage*

  Similar to the 2-point field goal percentage, this probability decides if a 3-point attempt scores in one possession.

- *Defensive rating*

  It is an estimate of points allowed per 100 possessions, which means the points got by the opponent that the player individually faces per 100 possessions. Therefore, the lower defensive rating, the better defensive ability that player has.

- *Offensive rebounds per 100 team possessions*

  This is the number of offensive rebounds got by the player per 100 team possessions. This statistics for all players in the field will be summed up to measure the ability for the team to fight for an offensive rebound. Offensive rebound usually leads to second-chance opportunities.

- *Defensive rebounds per 100 team possessions*

  This statistic refers to number of defensive rebounds got by the player per 100 team possessions. Like last statistic, it will be added up for all the players in the field.

2.2 Event Variables

- $\{a, b\} = \{1, 2\}$, indices for offensive/defensive side. When a possession ends, a and b are exchanged along with the beginning of another possession.

- $\mathbf{t}$ denotes the time point of the simulated game.

- $\mathbf{D}$ refers to cumulative goal difference. It, to some extent, describes the momentum of a team's scoring results. If a team scores much more points than its opponent in a period, it is reasonable to assume the goal percentage becomes higher for the next shots as the team may find effective solutions of opponent's defense.

- The offensive time $\mathbf{t^i}$ every possession for $\mathbf{i}^{th}$ team can be estimated by $t^i_{min} + Unif(0, 24 - t^i_{min})$, where $t^i_{min} = \left\lceil (\frac{48\times60}{PACE_i} - 24) \times 2.5 \right\rceil$ and $PACE_i$ is the number of paces for $\mathbf{i}^{th}$ team.

- $\mathbf{Fast}$(=0 or 1) denotes if next attempt is a fast break that certainly leads to a 2-point score.

- $\mathbf{score^i}$ refers the score of $\mathbf{i}^{th}$ team.

2.3 Input Variables

- $\mathbf{Q_{i,l,j,k}}$ denotes the $\mathbf{k}^{th}$ statistic for $\mathbf{j}^{th}$ player in the field of $\mathbf{l}^{th}$ line-up in $\mathbf{i}^{th}$ team, $i = 1, 2$, $l = 1, 2$, $j = 1, 2, \ldots, 5$, $k = 1, 2, \ldots, 8$. Table 1 shows all categories of $\mathbf{k}$.

- $\mathbf{ORB_i}, \mathbf{DRB_i}$ denote the total offensive/defensive rebounds of all players in the field per 100 team possessions of $\mathbf{i}^{th}$ team. $ORB_i = \sum_{j=1}^{5} Q_{i,l,j,7}, DRB_i = \sum_{j=1}^{5} Q_{i,l,j,8}$.

- $\mathbf{t_{change}} = [10; 17; 24; 28; 36; 40]$ denotes the time points to change line-ups. After exchanging line-ups, Q, ORB, DRB need to be updated at the same time.

k	Season Average Statistics
1	field goal attempts per 100 team possessions
2	2-point attempts per 100 team possessions
3	3-point attempts per 100 team possessions
4	2-point field goal percentage
5	3-point field goal percentage
6	defensive rating
7	offensive rebound per 100 team possessions
8	defensive rebound per 100 team possessions

Table 1: statistics of **k**

- Total turnover number of team a per game is $\mathbf{turnover_a}$. The probability that a turnover occurs in an possession is $Pr\{\mathbf{turnover(a)}\} = \frac{\mathbf{turnover_a}}{\mathbf{PACE_a}}$. (Only the offensive side is likely to have a turnover).
- The probability that $\mathbf{j}^{th}$ player will shoot at that possession of team **a** is $\mathbf{P_{a,j}} = \frac{Q_{a,l,j,1}}{\sum_{j=1}^{5} Q_{a,l,j,1}}$.
- For a chosen play who will shoot, the probability that he shoots for **M** points is $\mathbf{P_{a,j,M}} = \frac{Q_{a,j,M}}{Q_{a,l,j,2}+Q_{a,l,j,3}} \quad M = 2, 3$.
- Given $\mathbf{M}$(2 or 3), the probability one shot scores is $Pr\{\mathbf{score(a)}\} = Q_{a,l,j,M+2} * \frac{Q_{b,l,j,6}}{100} + \frac{2D*(-1)^a}{100}$, in which the scoring momentum **D** is taken into consideration.

2.4 Decision Process

- If a shot misses, the offensive side gets the rebound with $\frac{ORB_a}{ORB_a+DRB_b}$.
  - An offensive rebound gives the 2^{nd} possession, which occupies a half time of the usual possession.
  - A defensive rebound means exchange of possession. If previous shot is 3-pt, there is a low probability that fast break happens as the ball may bound far away.
- If a shot scores, the offensive side gets corresponding **M** points and then possession is exchanged.
- If $|\mathbf{D}|$ reaches 4, a time out is called and **D** is then reset to 0.
- If time variable reaches any exchange time point after one possession ends(score, defensive rebound or turnover), then the line-ups will be changed.

- If a turnover occurs, the opponent will grasp the possession that leads to a fast break.
- If time variable is beyond $48 * 60$ seconds, the game simulation is terminated.

Our simulation process is reflected in Figure 1 as below.

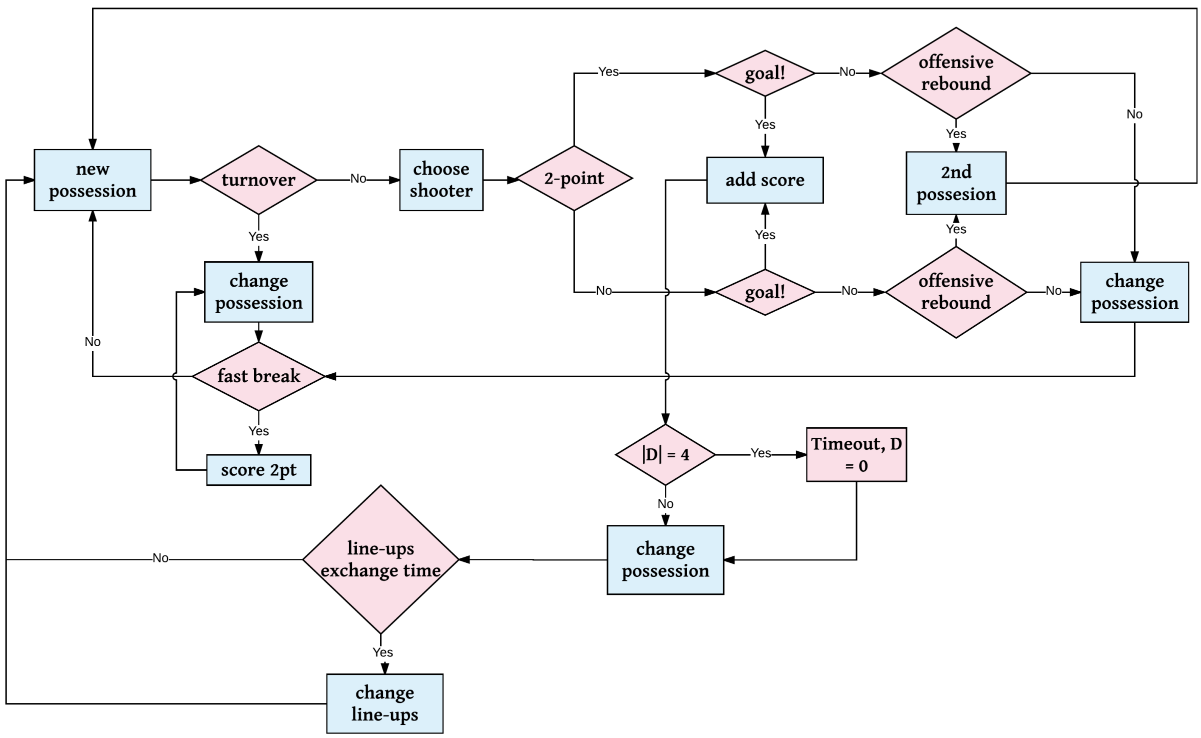

Figure 1: simulation flow chart

2.5 Simulation Method

The NBA playoff is the post-season tournament. It includes in total 4 rounds. Each round is a best-of-seven series, with the first team which wins four games advancing into the next round, while the other team will be eliminated from the playoff. Firstly, all the teams compete for their conference champion. The winner of the conference champion should pass 3 rounds: first round, conference semi-final, conference final. Then, the two conference champions will represent their conference to fight for the final NBA champion.

There are in total 16 qualified teams from Western Conference and Eastern Conference to fight for the championship. Therefore, statistics of regular season this year for these 16 teams are collected.

Player	FGA	2PA	3PA	2P%	3P%	DRTG	ORB	DRB
Stephen Curry	26.3	11.9	14.4	53.7	41.1	105	1.1	5.3
Klay Thompson	25	13.2	11.8	51.6	41.4	108	0.9	4.3
Kevin Durant	23.8	16.5	7.3	60.8	37.5	101	0.9	11
Draymond Green	12.7	7.6	5.1	49.4	30.8	99	1.9	9.8
Zara Pacchulia	11.6	11.5	0.1	53.8	0	101	5.3	10.2

Table 2: example of starting line-up for *Golden State Warriors*

Two line-ups are allowed for each qualified team in our simulation— starters and substitutes. Hence, each team has 10 players, we collected the 8 statistics($\mathbf{k}$ in table 1) for each player and sorted them by positions. These data will be imported into MATLAB and stored as a matrix. Table 2 a sample of organized data for one line-up:

In MATLAB, we used a 4-dimensional matrix $Q(i, j, k, l)$, where $i = 1, 2, l = 1, 2, j = 1, 2, \ldots, 5, k = 1, 2, \ldots, 8$ to represent all players statistics. For the subscripts, i refers to the index of teams; l is the index of line-ups;j is the index of players; k denotes the index of statistics.

We simulated the Play-off games this year and predict the final champion.

- Given two teams, in each simulation, at least four games would be simulated until one team get four victories.
- Such simulation process is repeated 10^3 times to determine which team is more likely to win the series. Then that team advance into next round.

3 Simulation Results

Our simulated results are shown in Figure 2, 3, 4. Surprisingly, Houston Rockets was the champion for the western conference. It beats two strong opponents: San Antonio Spurs and Golden State Warriors. San Antonio Spurs is considered as one of the best teams in the whole league and Golden State Warriors held the best regular season record. Therefore, it is interesting to see whether Houston Rockets can perform as well as it does in our simulation. In contrast with the western conference, the simulation results for eastern conference are in line with most people's expectation: last year's NBA champion — Cleveland Cavaliers wins the eastern conference champion.

The finals are between Houston Rockets and Cleveland Cavaliers. Rockets does not retain their good momentum, making Cavaliers achieve their second NBA championship.

We can wait until this June to see in the real world whether Cavaliers can be the final winner. Its worth to mention that the NBA playoff are in progress. According to the latest results, there are already 5 teams advancing to the next round. 4 out of them match our simulation results.

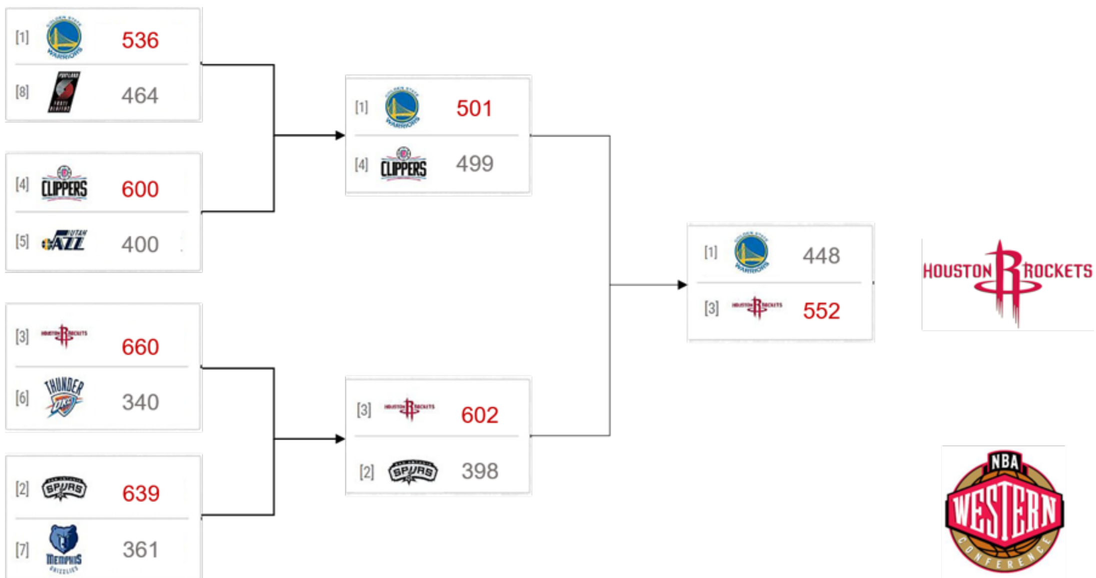

Figure 2: western conference results

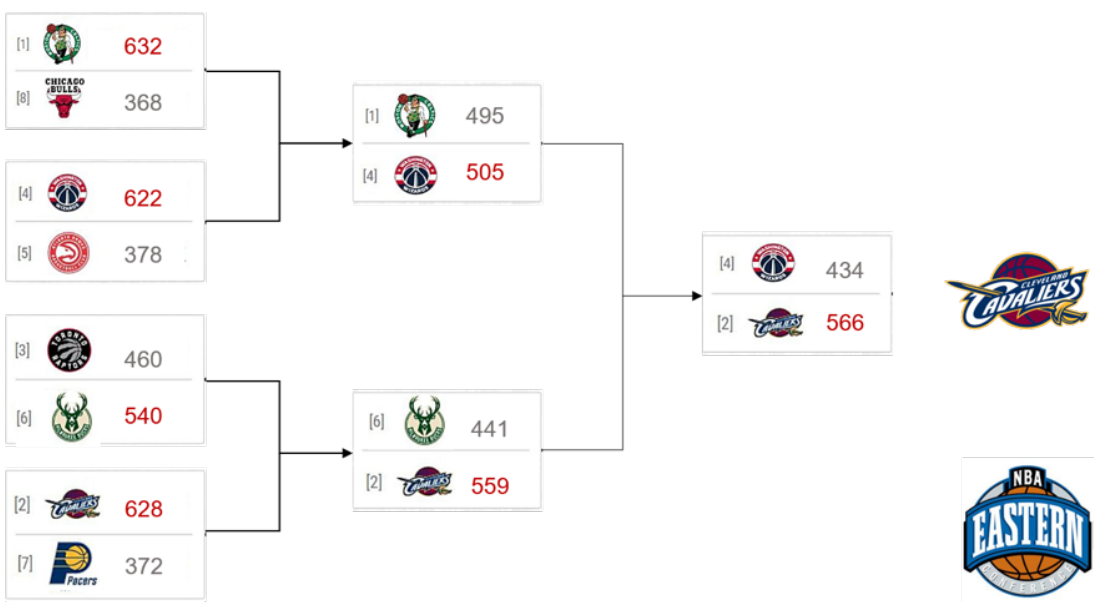

Figure 3: eastern conference results

Figure 4: final conference result

4 Future Work

The formulas in our simulation method could be optimized to become closer to the real game. Applying machine learning tools, a large sample set can be divided into training set, test set and validation set. In addition, many other factors need to be taken into accounts. For instance, ‘energy’ should be considered so that player’s field goal percentage would decay as the number of his attempts grows. Allowing three or more line-ups for each team may further make our simulator closer to the real game.

Given an opponent, we can compare different tactics and line-ups so that a maximum probability of winning is reached for one team in our simulator. For example, according to simulation results, the weight of goal attempts for each player and the ratio of 3-point shots can be adjusted for different opponents.

Admittedly, no matter how many factors are covered or how delicate formulas are produced in our simulator, it is still far away from the real games. However, through relevant adjustments against features of opponents, we hope the simulator can give some suggestion to find the best tactics.

5 Appendix

5.1 Algorithms

Algorithm 1: NBA Simulator Algorithms

Input : Q, ORB, DRB, Offensive/ Defensive Indices {a,b}, t_{change}
Initialization: $D = 0$, $t = 0$, $score^a = score^b = 0$, a = 1, b=2, $P = P_{starters}$, $ORB = ORB_{starters}$, $DRB = DRB_{substitutes}$, fast = 0 ;
while $t \leq 48 \times 60$ **do**
  read current;
  if $!isempty(t_{change})$&& $t > t_{change}(1)$ && $fast \,!= 1$ **then**
    exchange starter and substitute, determine l, update Q;
    delete $t_{change}(1)$ from t_{change};
  end
  if $fast == 1$ **then**
    $t_a = 0.5 * t^a_{min}$, $Score^a$+= 2, fast = 0, $D = D + (-1)^a$; change offensive side, $c = a, a = b, b = c$, $t = t + t_a$;
    if $|D| == 4$ **then**
      time out, $D = 0$;
    end
    continue;
  end
  Generate $t^a \sim t^a_{min} + Unif(0, 24 - t^i_{min})$, $u \sim Unif(0,1)$;
  if $u < P(turnover^i)$ **then**
    turnover happens, change offensive side, $t = t + 0.5 * t_a$, $c = a, a = b, b = c$;
    Generate $u \sim Unif(0,1)$, If $u <= 0.5$, then fast = 1; continue;
  end
  Generate $u \sim Unif(0,1)$, choose player j to shoot based on $P_{a,j}$;
  Generate $u \sim Unif(0,1)$, determine 2pt or 3pt based on $P_{a,j,2}$;
  If shoot for 2pt, L =2, else L =3;
  Generate $u \sim Unif(0,1)$ to determine if the shot scores;
  if $u \leq Q_{a,l,j,M+2} * \frac{Q_{b,l,j,6}}{100} + \frac{2D*(-1)^a}{100}$ **then**
    $t = t + t_a, score^a += M, D = D + (-1)^a$, change offensive side, $c = a, a = b, b = c$;
    if $|D| == 4$ **then**
      time out, $D = 0$;
    end
  else
    Generate $u \sim Unif(0,1)$, determine which side get the rebound;
    if $u \leq \frac{ORB_a}{ORB_a + DRB_b}$ **then**
      $t = t + 0.5 * t_a$, offensive side remains the same
    else
      change offensive side, $c = a, a = b, b = c$, $t = t + t_a$;
      if $L == 3$, fast = 1 with 10%
    end
  end
end

5.2 Codes

The codes for one simulated game between Cavaliers and Rockets are listed as below.

```
Q = zeros(2,2,5,8); ORB = zeros(2,2); DRB = zeros(2,2); Player_Name = cell
    (2,2,5); Team_Name = cell(1,2);
%Q(i,j,k,l), i = 1,2, j = 1,2, k = 1,2,...,5, l = 1,2,...,8
%i is the index of teams, j is the index of line-ups, k is the index of
%players, l is the index of statistics.
%k = 1: field goal attempts per 100 team possessions
%k = 2: 2-point attempts per 100 team possessions
%k = 3: 3-point attempts per 100 team possessions
%k = 4: 2-point field goal percentage
%k = 5: 3-point field goal percentage
%k = 6: defensive rating
%k = 7: offensive rebound per 100 team possessions
%k = 8: defensive rebound per 100 team possessions
%k = 9: name of the player
Team_Name = {'Cleveland Cavaliers', 'Houston Rockets'};
%starters for team 1
Q(1,1,:,:) = [28.1,19.3,8.7,0.505,0.401,112,1,3.5
15,3.6,11.4,0.329,0.351,112,0.7,4
24,17.9,6.1,0.611,0.363,108,1.7,9.7
23,12.7,10.3,0.471,0.373,107,3.9,13.7
9.3,9.3,0.1,0.604,0,109,6.1,9.2];

Player_Name(1,1,:) = { 'Kyrie Irving' 'J.R. Smith' 'LeBron James' ...
    'Kevin Love' 'Tristan Thompson'};

%substitutes for team 1
Q(1,2,:,:) = [15.1,9.7,5.4,0.489,0.415,114,0.1,4.5
12.6,5.9,6.7,0.469,0.36,112,1,4.6
15.6,4,11.6,0.493,0.485,114,0.4,5.2
10.6,4.9,5.8,0.58,0.333,113,0.9,5.4
18.6,6.6,12,0.546,0.409,109,1.3,9.1];

Player_Name(1,2,:) = { 'Deron Williams'  'Iman Shumpert' 'Kyle Korver'...
    'Richard Jefferson' 'Channing Frye'};
%starters for team 2
Q(2,1,:,:) = [25,12.7,12.3,0.53,0.347,107,1.5,9.2
12.7,6,6.7,0.461,0.382,108,2.2,7
13.8,4.2,9.6,0.556,0.343,108,0.9,7
17.6,6.1,11.5,0.446,0.404,113,2.5,4.9
17.4,17.4,0,0.643,0,106,5.5,10.8 ];

Player_Name(2,1,:) = { 'James Harden'  'Patrick Beverley' 'Trevor Ariza'...
    'Ryan Anderson' 'Clint Capela'};
%substitutes for team 2
Q(2,2,:,:) = [23.4,9.8,13.7,0.36,0.4,116,0,2.3
21,7.3,13.7,0.469,0.372,113,0.6,3.6
14.5,8.2,6.3,0.591,0.321,110,3.2,6.4
```

```
15.6,15.2,0.3,0.663,0.143,110,3.7,6.2
15.9,15.5,0.4,0.624,0.333,106,3.8,7.5];

Player_Name(2,2,:) = {'Bobby Brown' 'Eric Gordon' 'Sam Dekker' ...
    'Montrezl Harrell' 'Nene Hilario'};
%ORB(i,j), DRB(i,j)  i = 1,2, j = 1,2
%i is the index of teams, j is the index of line-ups,
ORB(1,1) = sum(Q(1,1,:,7)); %total ORB of starters of 1st team,100 poss
ORB(1,2) = sum(Q(1,2,:,7)); %total ORB of substitutes of 1st team 100 poss
ORB(2,1) = sum(Q(2,1,:,7)); %total ORB of starters of 2nd team 100 poss
ORB(2,2) = sum(Q(2,2,:,7)); %total ORB of substitutes of 2nd team 100 poss

DRB(1,1) = sum(Q(1,1,:,8)); %total DRB of starters,1st team 100 poss
DRB(1,2) = sum(Q(1,2,:,8)); %total DRB of substitutes,1st team 100 poss
DRB(2,1) = sum(Q(2,1,:,8)); %total DRB of starters,2nd team 100 poss
DRB(2,2) = sum(Q(2,2,:,8)); %total DRB of substitutes,2nd team 100 poss
PACE = [96.2, 100]; %the number of possessions PACE(1)-team 1, PACE(2)-team
    2
turnover = [1122/82, 1241/82]; %the total turnover number, turnover(1) is
    for team 1, turnover(2) is for team 2
t_min = ceil((48 * 60 ./ PACE - 24) * 2.5);
%the minimum time for every possession, t_min(1) is for team 1, t_min(2) is
    for team 2
Team1Wins = 0; Team2Wins = 0;
run = 1;
while run <= 1 %set the total number of runs
    D = 0; %cumulative goal difference
    t = 0; %game time
    Score = [0, 0]; %Score(1) is for team 1, Score(2) is for team 2
    a = 1;  b = 2; %team 1 offends
    line_ups = 1;
    %line_ups = 1 means starters on the field, line_ups = 2 means
        substitutes on the field
    fast = 0; %fast is 1 means current attempt is fast break
    t_change = [10, 17, 24, 28, 36, 40] * 60; %time point to change line-
        ups

    while t <= 48 * 60 - 24
       pause(0.25);
        if ~isempty(t_change) && t > t_change(1) && fast ~= 1
            line_ups = mod(line_ups, 2) + 1;  %line_ups change between 1
                and 2
            t_change(1) = [];  %delete the passed time point of changing
                line-ups
            S = sprintf('Line-ups exchange');
            disp(S)
        end
        if fast == 1
            t_a = 0.5 * t_min(a);   %t_a is the time for this possession
            Score(a) = Score(a) + 2;
            fast = 0;
```

```
            t = t + t_a;
            S = sprintf('%d min %d s %s fast break Scores, %d - %d', ...
                floor(t / 60), floor(mod(t, 60)), char(Team_Name(a)), Score(1), Score(2));
            disp(S)
            D = D + (-1) ^ a;
            %goal difference since last time out.Negative value means team 1 score more than team 2
            c = a; a = b; b = c; %change offensive side
            if abs(D) == 4
                D = 0; %time out
                S = sprintf('%s asks for time out', char(Team_Name(a)));
                disp(S)
            end
            continue;
        end
        t_a = t_min(a) + rand * (24 - t_min(a));  %t_a is the time for this possession
        if rand < turnover(a) / PACE(a)
            %turnover happens, change offensive side
            t = t + 0.5 * t_a;
            S = sprintf('%s turnover', char(Team_Name(a)));
            disp(S)
            c = a; a = b; b = c;

            if rand <= 0.5  %turnover leads to fast break
                fast = 1;
                continue;
            end
        end
        FGA_total =sum(Q(a, line_ups,:,1));
        %the total field goal attempt for all player on the field
        ShootProb = Q(a, line_ups, :, 1) / FGA_total; %the chance for each player to shoot
        u = rand;
        for j = 1:5   %choose the player to shoot
            if u <= ShootProb(j)
                break;
            end
            u = u - ShootProb(j);
        end
        Shoot2ptProb = Q(a, line_ups, j, 2) / (Q(a, line_ups, j, 2) +...
            Q(a, line_ups, j, 3)); %the probability for selected player to shoot 2pt
        if rand <= Shoot2ptProb
            L = 2;
        else
            L = 3;
        end
        ScoreProb = Q(a, line_ups, j, L + 2) *  Q(b, line_ups, j, 6)/100 + 2 * D * (-1)^a / 100;
```

```
        if rand <= ScoreProb
            t = t + t_a;
            Score(a) = Score(a) + L;
            S = sprintf('%d min %d s %s %s shoot %d pt, Scored! %d - %d', ...
                floor(t / 60), floor(mod(t, 60)), char(Team_Name(a)), ...
                char(Player_Name(a,line_ups,j)), L, Score(1), Score(2));
            disp(S)
            D = D + (-1) ^ a;
            %goal difference since last time out. Negative value means team 1 score more than team 2
            c = a; a = b; b = c; %change offensive side
            if abs(D) == 4
                D = 0; %time out
            end
        else
            t = t + t_a; %update time first to facilitate output in the next line
            S = sprintf('%d min %d s %s %s shoot %d pt, Missed!', floor(t / 60),...
                floor(mod(t, 60)), char(Team_Name(a)), char(Player_Name(a,line_ups,j)), L);
            disp(S)
            OffRebProb = ORB(a, line_ups) / (ORB(a, line_ups) + DRB(b, line_ups));
            %Probability for offsensive side get the rebound
            if rand <= OffRebProb
                S = sprintf('%s gets offensive rebound', char(Team_Name(a)));
                disp(S)
                t = t - 0.5 * t_a; %offensive side remains the same, time is half of normal possession

            else
                S = sprintf('%s gets defensive rebound', char(Team_Name(b)));
                disp(S)
                c = a; a = b; b = c;

                if L==3 && rand <= 0.1  %there is 10% chance to be fast break
                    fast = 1;
                end
            end
        end
    end
    S = sprintf('Final Score is %d - %d', Score(1), Score(2));
    disp(S)
    if Score(1) > Score(2)
        Team1Wins = Team1Wins + 1;
    elseif Score(1) < Score(2)
```

```
            Team2Wins = Team2Wins + 1;
        else %end in a tie, then repeat the simulation
            run = run - 1;
        end
        run = run + 1;
end
```