Network Representations Of Real And Fictional Basketball Games

Trace Arbuckle
Computer Science and Engineering
University of Nevada, Reno
Reno, NV
tarbuckle@unr.edu

***Abstract*—In all NBA and most college basketball games, two types of gameplay data are recorded. The first is 48 minutes of “Player p did s at time t,” where every s increments statistical totals in categories including points, shot attempts, rebounds, assists, and turnovers. The other type is player tracking data, structured as “player p was at (x, y) on the court at time t,” captured by clusters of state-of-the-art cameras that form a composite of over 25 points on the body in three-dimensional space. College tracking data is instead enabled by chip-installed jerseys and balls. However, chip outputs are diluted severely before these numbers are shared with team personnel, and the raw data isn’t available upon request like is the case for NBA teams. This paper proposes two new avenues of network theory-inspired data analysis to make up for this deficit. The first is the creation of network representations of passes, or all ball movements. The second is the simulation of games between fictional teams of real players who aren’t actually teammates, informed by their performance in real-life.**

I. Introduction

The narrative of baseball’s “Moneyball” revolution– as popularized by the film adaptation of a 2003 Michael Lewis book– was that traditional baseball statistics were misleading when used as indicators of player performance and value, and dependence on them from the league’s decision makers had created inefficiencies ripe for exploitation in the market for players. The uptick in analytics-driven decision making made its way into basketball in the 21st century too, as people progressively realized the limitations of the traditional counting stats. Jaren Jackson Jr., a friend and former teammate of mine who was just crowned the NBA’s ’22-’23 Defensive Player of the Year, records a defensive stat only about 15 percent of the time. He makes all sorts of great plays during the other 85 percent, but there’s no way to know from the box score unless it ends in a block, steal, or defensive rebound. Much remains to be desired in ascertaining player value during that “other 85 percent” on the offensive end of the court. If a pass leads directly to a made basket in a game, it gets recorded as an assist, a statistic with which all basketball fans are familiar. But assists represent well less than 10 percent of all passes; team network can keep track of the rest.

A team network is a network of intra-team ball movement, with no information beyond what can be conveyed by the directional links. All 7-10 players in a given team’s active rotation can have their own node, or it can be condensed to five nodes, as only five teammates share the floor at a time. In the NBA, the five positions are referred to as the point guard (PG), the shooting guard (SG), the small forward (SF), the power forward (PF), and the center (C). In college, this is usually simplified numerically to 1 through 5, roughly in order of player height. Note that in the condensed version, one individual may contribute to multiple position nodes over the course of a game or season. Two additional “basket” nodes can be added, one receiving links when players end possessions (via a shot / foul drawn / turnover), with the other sending out links to the players who start them (via a rebound / inbound pass). These links don’t distinguish between these methods, however.

A theoretical full-game network could contain interactions between opposing players and play results by indexing every link by type, on top of all of the information conveyed by a team network. Each link type corresponds to game events such that the entire traditional box score can be recreated from the network, whether or not the links have event timestamp data included. A game network requires three non-player nodes: the two basket nodes mentioned above, though now with both incoming and outgoing links from each, and another “out-of-bounds” node for all inbounds pass situations that do not follow a made basket or free throw. (Those links come from the basket node, and do not count as a pass for the inbounder.) It also requires the existence of links between players that do not represent an exchange of the ball. Between opponents, this allows for the representation of fouls and blocked shots, among other things; between teammates, it optionally allows for the inclusion of events like ball screens, where an offensive player attempts to block the path of the ball handler’s defender to create an advantage.

This project begins by constructing team networks for contests from the 2023 NCAA Men’s Basketball Tournament, colloquially known as “March Madness.” The underlying data for the networks is processed and stored in a program that parses both manually recorded passing data and preexisting play-by-play PDFs.

The remainder experiments with algorithmic methods of creating fictional game networks. Imagine taking one player of interest from 20 different March Madness games and splitting them into two teams of 10. Regardless of who comprises the 20 or how they are split up, it is feasible to create a fictional network amongst them such that the distribution of each player's incoming and outgoing links approximates that of their node in real-life games. The fictional network can be decomposed into a fictional play-by-play, box score, and final score.

Altogether, this method is proposed as a preliminary version of a new type of fantasy basketball game to be played by fans. Current iterations are enjoyed worldwide, but market penetration is dwarfed by fantasy football and growth is stagnant, leaving the door open for a new type of product.

II. Related Work

Three areas of related work, with the first two pertaining only to the NBA, are explored to guide the process of network creation for college games. The first is a paper from researchers at Arizona State University, which preceded the NBA's player tracking era entirely and produced various visualizations and metrics. The second is the current state of the NBA's player tracking apparatus, the publicly available data derived from it, and the possibilities it creates. The third is how a company called Synergy Sports processes basketball film for both the NBA and college, providing a sort of tracking data on a simpler level than Second Spectrum.

In "Basketball Teams as Strategic Networks," the ASU researchers took two games apiece from all eight first round matchups in the NBA playoffs in 2010 and recorded every exchange of the ball themselves. This allowed for the construction of team networks in the 'condensed' state as described earlier. The consolidation to only five position nodes per team enabled a network representation of all 16 teams and games combined, shown in Figure A.

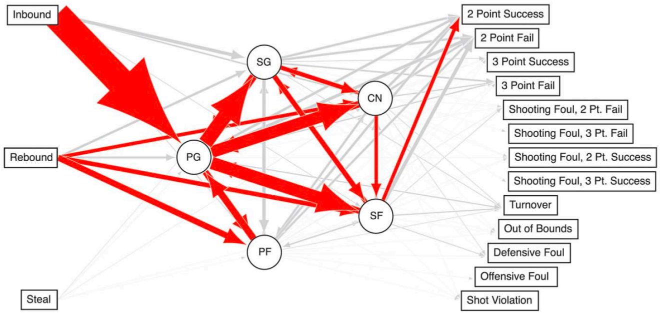


Figure 1. Weighted graph of ball transitions across all teams and all games. Edge width is proportional to probability of transition between nodes. Red edges represent transition probabilities summing to the 60th percentile.
doi:10.1371/journal.pone.0047445.g001

Fig. 1. (from [1])

The frequencies of various ball movements relative to others are signified by the thickness of the arrows. In the networks for individual teams, those 13 types of events that end possessions are simplified into 'success' or 'failure' for ease of comprehension and follow-up calculations.

All statistics produced from the networks were calculated by team, in an attempt to see if any characteristic stood out as being especially predictive of success. The simplest statistic was the mean and variance of 'path length,' or the number of passes plus two (to include how the possession started and ended.) Degree centrality, flow centrality, and team entropy were also calculated in an attempt to show numerical evidence of which teams had a centralized attack through a star player or two, and which teams spread opportunities more evenly.

The authors were vindicated in their approach when the player tracking era began soon after to automate the collection of passing data. Since 2017, a company called Second Spectrum has equipped NBA arenas with cameras above the court that capture the precise locations of 10 players and the ball many times per second. SportVU was the tracking provider for the four seasons prior, making this technology a decade old in total. Second Spectrum, which does the same sort of thing for soccer matches, was acquired by Genius Sports in 2021 for a massive 200M dollars, demonstrating the growth that investors envision for this technology.

Current cameras track more than 25 points on a player's body in three dimensions at every time step, but this number will grow to 7,000 or more in the next generation. The company is hard at work on proprietary software to derive benefits from this upgrade, in the hopes that it results in more monetizable value for both teams and fans. An example of an existing application is real-time shot distances on TV broadcasts; if Stephen Curry makes a shot from way behind the three-point line, fans can know exactly how many feet away from the basket he was.

The "NBA Advanced Stats" database on NBA.com has a 'Tracking' tab with 14 categories of numbers calculated from Second Spectrum's data. It includes viewable options like total passes made, total passes received, touches per game, seconds per touch, and dribbles per touch. It also distinguishes between things like whether a shot came after a dribble or a catch, or whether a rebound was contested or uncontested. It even displays every player's average distance traveled per game in miles.

The construction of team networks as described in the introduction would be a trivial extension of Second Spectrum's location data if the passing sequences derived from it were all publicly available. But only NBA front offices, scouting departments, and coaching staffs have access to all of the data in its raw form, allowing for more complex experimentation. For example, they may try to calculate the 'gravity' of offensive players in certain situations, which is the ability to draw attention and keep your defender close by so they can't play help defense on other players. But breakthroughs in this endeavor are industry secrets, as the edge disappears if other teams become aware of new insights and agree on the value they provide. Therefore, the people most fit to produce new basketball knowledge from player tracking are keeping their findings to themselves; hopefully, this eventually spills over into the public domain.

The film from all NBA and college games are available the next day on a subscription website called Synergy Sports, which breaks up the video into clips for ease of navigation.

Each clip has play type, sequence, and result information associated with it. The result information includes the shot distance, and whether it was guarded or unguarded. These two data points and some others are combined to create a metric called Synergy Shot Quality (SSQ), which is defined as the expected point total if an average player took that shot. For example, if it is a two-point shot that an average player would make 50 percent of the time, or a three-point shot that an average player would make 33.3 percent of the time, the SSQ is 1.

Synergy does not use Second Spectrum data to create SSQ and other types of advanced insight. Instead, they pay a company called ShotTracker for the information that enables this. Before games, team personnel place player-specific chips in a slot near the back shoulder of each jersey. Another chip sits on the inside edge of the ball. (This chip is either light enough to not influence the projectile motion of the ball, or it is counterbalanced on the opposite side. Or maybe all balls have been marginally lopsided since ShotTracker began.) The way in which ShotTracker parses real-time chip location data into human-readable basketball information is a black box– same as for Second Spectrum method of processing tracking video– but its results show up in things like SSQ and shot charts. All college teams have access to a ShotTracker tablet app that shows various data immediately upon the conclusion of games, like the relative performances of each distinct 5-man lineup that is employed throughout.

This app also shows the relative percentage of possessions that had 0-2 passes, 3-5, or 6+. This indicates that underneath the hood, every single pass is recorded with both the passer and recipient noted. If this data was available in its entirety, creating network representations of ball movement would be trivially simple. But this information is hidden even from team personnel, and so a portion of the remainder of this paper relies on charting passing sequences manually.

III. Methodology

This project's results are the output of two separate programs written in Java, both processing data taken from '23 March Madness games. The first takes in a game's passing sequences as a .csv file and derives the resultant team network information. The second takes in two fictional teams of players and all of their real-life stats across multiple March Madness games. It then simulates one or many games between those teams.

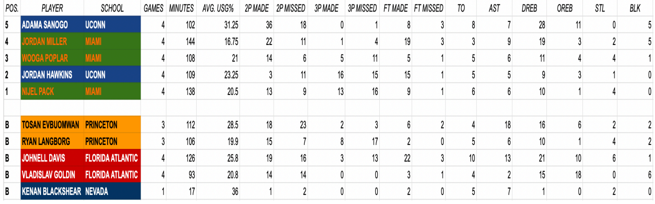
While the ASU paper was prescient in its goal given the player tracking era that would follow and the lack of statistical sophistication at the time, its results are mostly uninformative due to the elimination of specific player nodes in favor of a generic five-position model. These positions are really just approximations of how a team will match up to an opponent defensively, and have little bearing on offensive roles. For example, some NBA centers shoot no three-pointers and spend most of their time near the basket, while some hang out on the perimeter and shoot mostly threes. If these two players split the playing time at center for a given team, the center node is a diluted version of both, resulting in the team's true play style being corrupted in the network representation. As such, the first program does not simplify passing data into the five-position model, instead keeping all individual players as their own node. On top of this, many of the network theory statistical algorithms that the ASU researchers used were a poor fit for a network with only five nodes. And while an increase to 7-12 nodes solves the last paragraph's problem, it makes a negligible difference in this sense. Their result calculations like clustering coefficients and team entropy are nebulous at best in terms of how they map to basketball insights. And even though they used a small sample size of two games per team, which should create more variance, the results are clustered together and not much jumps out; some outliers are interesting on their face, but almost all of them can be quickly explained by "Team A has star player B at position C," and that's as far as the potential analysis goes. This likely extends beyond basketball to all networks with such a small number of nodes, no matter how complex the edge patterns get. This might not be the cases if the edges are weighted. While edges in a basketball network can be directional and labeled, there's no intuitive way to employ weights.

A simple adjacency matrix, with nodes for every individual instead of every position, allows for straightforward analysis without relying on a bunch of calculations performed seemingly just because. The data lives in spreadsheets, with a given player's name appearing once for every time they touch the ball. The information moves to the next row when possession alternates, so a 70 possession per-team game would have 140 rows in total. A simple java implementation parses the information and stores directional links. Note that players have no incoming link if they start a possession via a rebound or inbound catch, and they have no outgoing link if they end a possession via a shot, a turnover, or a shooting foul drawn. A player's row in the adjacency matrix represents the passes he threw, and his column represents the passes he caught. An example of this is produced in the results section.

For all of the talk in the introduction about the limitations of traditional stats, the second program relies primarily on box score information, not passing sequences. The creation of fictional game networks from fictional teams– but with real players and real stats– is a novel task. There is no point in starting from passing information when there are no events yet that collectively resemble a game; ball movement networks are already an afterthought to the basketball community for real games for now. A skeleton of a game must be created first; ball movement information and other ideas can be entertained afterward to iterate on the algorithm and make the simulations progressively more realistic.

Like in the first program, spreadsheet data serves as the starting point. There are two .csv files for each team. The first type is shown in Figure B. From here on, this group of 10 players from five schools is 'Team A.'

The stats are summed from each players' '23 March Madness games between Round 1 and the 'Elite 8'; national semifinal and championship games are excluded. Almost all of

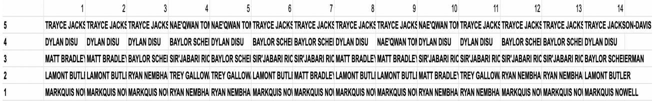


POS.	PLAYER	SCHOOL	GAMES	MINUTES	AVG. USG%	2P MADE	2P MISSED	3P MADE	3P MISSED	FT MADE	FT MISSED	TO	AST	DREB	OREB	STL	BLK
5	ADAMA SANOGO	UCONN	4	102	31.25	36	18	0	1	8	3	8	7	28	11	0	5
4	JORDAN MILLER	MIAMI	4	144	16.75	22	11	1	4	19	3	3	9	19	3	2	5
3	WOOGA POPLAR	MIAMI	4	108	21	14	6	5	11	5	1	5	6	11	4	4	1
2	JORDAN HAWKINS	UCONN	4	109	23.25	3	11	16	15	15	1	5	5	9	3	1	0
1	NIJEL PACK	MIAMI	4	138	20.5	13	9	13	16	9	1	6	6	10	1	4	0
B	TOSAN EVBUOMWAN	PRINCETON	3	112	28.5	18	23	2	3	6	2	4	18	16	6	2	2
B	RYAN LANGBORG	PRINCETON	3	106	19.9	15	7	8	17	2	0	5	6	10	1	4	2
B	JOHNELL DAVIS	FLORIDA ATLANTIC	4	126	25.8	19	16	3	13	22	3	10	13	21	10	6	1
B	VLADISLAV GOLDIN	FLORIDA ATLANTIC	4	93	20.8	14	14	0	0	3	1	4	2	15	18	0	6
B	KENAN BLACKSHEAR	NEVADA	1	17	36	1	2	0	0	2	0	5	7	1	0	2	0

Fig. 2. Team A '23 March Madness Statistics

these categories are common knowledge to basketball novices. The exception is usage percentage, though it has become more prevalent over time. A given player's usage rate is the percent of offensive possessions that end with him while he is on the floor, either via a shot attempt, a turnover, or a shooting foul drawn. A perfectly balanced team would have five players each with a usage rate of 20 percent. Because Team A is an all-star team of sorts, the collective usage of any grouping of five players will exceed 100. In this context, the usage rate for each player is averaged across all of their tournament games, weighted by minutes played in each.

The second type of .csv file used is shown in Figure C, this time with the players from Team B.



	1	2	3	4	5	6	7	8	9	10	11	12	13	14
5	TRAYCE JACKS	TRAYCE JACKS	TRAYCE JACKS	NAE'QWAN TO	NAE'QWAN TO	TRAYCE JACKS	TRAYCE JACKS	TRAYCE JACKS	TRAYCE JACKS	NAE'QWAN TO	TRAYCE JACKS	TRAYCE JACKS	TRAYCE JACKS	TRAYCE JACKSON-DAVIS
4	DYLAN DISU	DYLAN DISU	DYLAN DISU	BAYLOR SCHEI	DYLAN DISU	BAYLOR SCHEI	BAYLOR SCHEI	DYLAN DISU	NAE'QWAN TO	DYLAN DISU	DYLAN DISU	BAYLOR SCHEI	BAYLOR SCHEI	DYLAN DISU
3	MATT BRADLEY	MATT BRADLEY	BAYLOR SCHEI	SIR'JABARI RIC	BAYLOR SCHEI	SIR'JABARI RIC	SIR'JABARI RIC	MATT BRADLEY	MATT BRADLEY	SIR'JABARI RIC	SIR'JABARI RIC	SIR'JABARI RIC	SIR'JABARI RIC	BAYLOR SCHEIERMAN
2	LAMONT BUTLI	LAMONT BUTLI	RYAN NEMBHA	TREY GALLOW	TREY GALLOW	LAMONT BUTLI	MATT BRADLEY	LAMONT BUTLI	LAMONT BUTLI	MATT BRADLEY	TREY GALLOW	RYAN NEMBHA	RYAN NEMBHA	LAMONT BUTLER
1	MARKQUIS NO	MARKQUIS NO	MARKQUIS NO	RYAN NEMBHA	RYAN NEMBHA	MARKQUIS NO	MARKQUIS NO	MARKQUIS NO	MARKQUIS NO	RYAN NEMBHA	RYAN NEMBHA	MARKQUIS NO	MARKQUIS NO	MARKQUIS NOWELL

Fig. 3. Team B Hypothetical Playing Time Rotation

The columns represent the five-man lineup on the floor in each of 14 game segments (7 per half). In this project's simplified, clock-agnostic model, each segment has five possessions per team, so the entire game has 70 possessions. A team that averages 70 possessions per game would be in the 88th percentile of the country in tempo; the average is around 67.

Under the hood of the program, AST, REB, STL, and BLK are converted into 'rate stats' for use later on; that is, a private member variable stores the value of each of these when divided by minutes played. Rate stats for NBA players are commonly formatted as per 36 minutes played, or per 100 possessions. This turns the counting stats into something that can be normalized across all players.

The program in its entirety contains Main, Player, Team, and Game classes. The Main class parses the first type of .csv and fills 10 Player objects accordingly, which can collectively be passed to the Team constructor. Game object constructors are passed two Team objects, and the second type of .csv for each. In the Game class, an overarching runGame() method has 14 runSegment() calls nested within it. Ten runPlay() calls lay within runSegment().

Each runSegment() call begins with a normalize() call that proportionally decreases everyone's usage percentages, resulting in the five rates adding to 100. The normalize() method also updates arrays called USGA and USGB that both store four variables of type double. The doubles split the 0-to-1 range into 5 parts; each part corresponds to a given player, and their sizes are determined by the normalized usage rates.

Each runPlay() call begins with the creation of a random double between 0 and 1. This will fall into the range for one of the five players, making that possession his to be used. A getRandomPoss() method in Player returns either a 2P make, a 2P miss, a 3P make, a 3P miss, a turnover, or a trip to the FT line, based on that player's relative proportions of each in the stats .csv file. Whatever possession is chosen is removed from the pool for the rest of the game, so the proportions change slightly for next time after a given event is used.

The method of choosing a player by using an array of four doubles and a random number between 0 and 1 is also used to determine who gets assigned assists, rebounds, and steals at times. There are separate arrays for this, proportionalized according to the rate stats mentioned a bit ago.

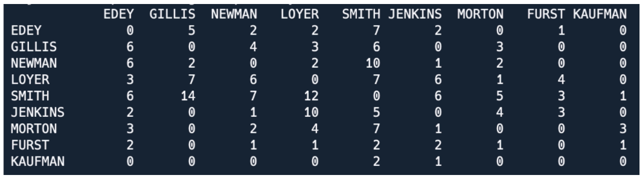
Altogether, when runGame() finishes, a complete play-by-play, box score, and final score become available. This is demonstrated in the Results section.

IV. Results

The game chosen to demonstrate and test the first program is arguably the most notable of the 2023 tournament; 16 seed Fairleigh Dickinson, with a middling win-loss record of 19-15 in the worst conference in all of college basketball, defeated 1 seed Purdue, who won 29 of their 34 games while competing in the 2nd best conference. Prior to the game, the well-respected analytics website kenpom.com had FDU was ranked as the 299th best out of 363 Division I teams; Purdue was ranked 6th, and predicted to have a 98 percent chance of winning. This was only the 2nd time a team seeded 16th has won their first round matchup versus a team seeded 1st in about 150 chances since March Madness expanded to 64 teams in 1985. (I'm thankful it has since expanded to 68, as the UNR team that employs me was the very last team to receive an at-large berth this year.)

The noteworthy storyline heading into the game was how FDU planned to defend Zach Edey, Purdue's best player. Edey stands at a whopping 7'4" and weighs over 300 pounds; his dominance as a scorer, rebounder, and shot blocker earned him National Player of the Year honors from the Associated Press and other outlets. On the other side, FDU was ranked dead last (363rd) in the country in average height weighted by playing time (6'1.4"). That average was dragged down by two diminutive starting guards at 5'8" and 5'9", but the situation on the front line wasn't much different; FDU's two tallest players were 6'6"– a 10-inch disadvantage to Edey– and one of them didn't even scratch 200 pounds.

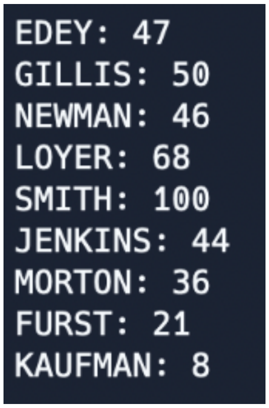
So what happened on the floor to enable FDU's 63-58 victory? One factor is visible rightaway in the box score. While Edey was efficient as always, he ended only 25 percent of Purdue's offensive possessions versus his season average of 32 percent; given how many minutes he played and the tempo of the game, that difference is worth four offensive plays. The adjacency matrix described earlier, shown for Purdue in this game in Figure D, seeks to provide additional context to that statistical anomaly and others.



```
          EDEY  GILLIS  NEWMAN   LOYER   SMITH JENKINS  MORTON   FURST KAUFMAN
EDEY         0       5       2       2       7       2       0       1       0
GILLIS       6       0       4       3       6       0       3       0       0
NEWMAN       6       2       0       2      10       1       2       0       0
LOYER        3       7       6       0       7       6       1       4       0
SMITH        6      14       7      12       0       6       5       3       1
JENKINS      2       0       1      10       5       0       4       3       0
MORTON       3       0       2       4       7       1       0       0       3
FURST        2       0       1       1       2       2       1       0       1
KAUFMAN      0       0       0       0       2       1       0       0       0
```

Fig. 4. Purdue Passing Adjacency Matrix Versus FDU

The indegrees and outdegrees for each player are also summed to show the degree centrality of each player, shown in Figure E for Purdue.



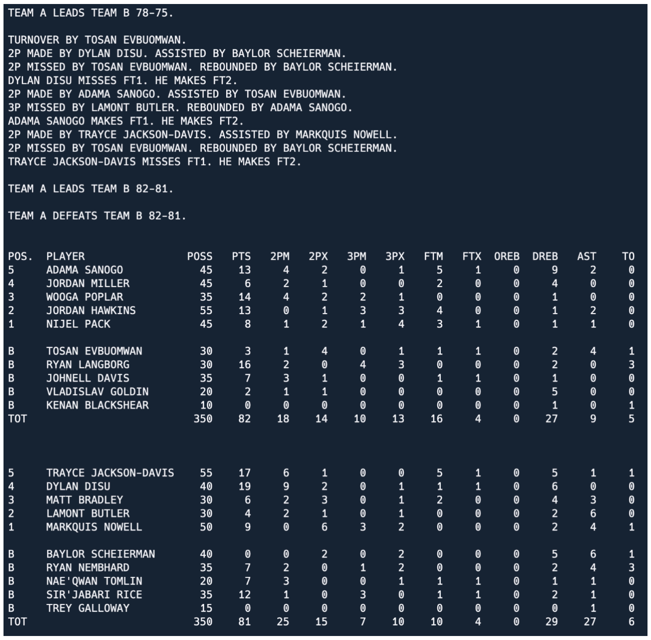
```
EDEY: 47
GILLIS: 50
NEWMAN: 46
LOYER: 68
SMITH: 100
JENKINS: 44
MORTON: 36
FURST: 21
KAUFMAN: 8
```

Fig. 5. Purdue Player Node Degree Centralities Versus FDU

The degree centralities instantly show that Edey was not the focal point of the offense like usual. It isn't quite as severe as it looks because neither his defensive rebounds nor his shots add to his degree, but nonetheless, he was clearly underfed the ball. Observing FDU's defensive strategy as a spectator shows why this was the case. While most opponents guard Edey with two defenders once he catches the ball– which has forced him to develop an arsenal of lightning quick moves immediately upon the catch– FDU double teamed him nearly 100 percent of the time to make up for their size disadvantage. One player would be in a "front," positioned directly between Edey and the ball, while the other would shade his backside to discourage lob passes over the top. The backside shader abandoning his man led Purdue to shoot well more three-pointers than usual, at almost 50 percent of shot attempts against a season average of 38 percent; they made only 19 percent of them versus a season average of 32. Purdue also turned the ball over 16 times– seven more than FDU – and many of those came from ignoring the double team and trying to throw the ball inside anyway.

FDU's passing network and degree centralities are not reproduced here, as it was their defense that won them the game, but an interesting result stood out. Role player Sean Moore caught a total of 15 passes from Grant Singleton, but only 17 from his other 7 teammates combined. That is three times the number he received from point guard Demetre Roberts, who is known both as more ball-dominant and a better passer than Singleton. Further analysis would be needed to figure out if this was happenstance or if it reveals an aspect of offensive scheme, but either way, it shows how the adjacency matrix alone can provoke new types of team analysis.

As for the fictional game networks, the previous section was incomplete in its explanation of how an element of randomness enables variance from one simulation to the next in both team and player performance. The best way to do this is not to explain every last implementation detail, but to show the output and work backwards from there. The end of Simulation 1– admittedly cherry-picked to include some late-game madness– is shown in Figure 6.



```
TEAM A LEADS TEAM B 78-75.

TURNOVER BY TOSAN EVBUOMWAN.
2P MADE BY DYLAN DISU. ASSISTED BY BAYLOR SCHEIERMAN.
2P MISSED BY TOSAN EVBUOMWAN. REBOUNDED BY BAYLOR SCHEIERMAN.
DYLAN DISU MISSES FT1. HE MAKES FT2.
2P MADE BY ADAMA SANOGO. ASSISTED BY TOSAN EVBUOMWAN.
3P MISSED BY LAMONT BUTLER. REBOUNDED BY ADAMA SANOGO.
ADAMA SANOGO MAKES FT1. HE MAKES FT2.
2P MADE BY TRAYCE JACKSON-DAVIS. ASSISTED BY MARKQUIS NOWELL.
2P MISSED BY TOSAN EVBUOMWAN. REBOUNDED BY BAYLOR SCHEIERMAN.
TRAYCE JACKSON-DAVIS MISSES FT1. HE MAKES FT2.

TEAM A LEADS TEAM B 82-81.

TEAM A DEFEATS TEAM B 82-81.


POS.  PLAYER                POSS   PTS   2PM   2PX   3PM   3PX   FTM   FTX  OREB  DREB   AST    TO
5     ADAMA SANOGO            45    13     4     2     0     1     5     1     0     9     2     0
4     JORDAN MILLER           45     6     2     1     0     0     2     0     0     4     0     0
3     WOOGA POPLAR            35    14     4     2     2     1     0     0     0     1     0     0
2     JORDAN HAWKINS          55    13     0     1     3     3     4     0     0     1     2     0
1     NIJEL PACK              45     8     1     2     1     4     3     1     0     1     1     0

B     TOSAN EVBUOMWAN         30     3     1     4     0     1     1     1     0     2     4     1
B     RYAN LANGBORG           30    16     2     0     4     3     0     0     0     2     0     3
B     JOHNELL DAVIS           35     7     3     1     0     0     1     1     0     1     0     0
B     VLADISLAV GOLDIN        20     2     1     1     0     0     0     0     0     5     0     0
B     KENAN BLACKSHEAR        10     0     0     0     0     0     0     0     0     1     0     1
TOT                          350    82    18    14    10    13    16     4     0    27     9     5



5     TRAYCE JACKSON-DAVIS    55    17     6     1     0     0     5     1     0     5     1     1
4     DYLAN DISU              40    19     9     2     0     1     1     1     0     6     0     0
3     MATT BRADLEY            30     6     2     3     0     1     2     0     0     4     3     0
2     LAMONT BUTLER           30     4     2     1     0     1     0     0     0     2     6     0
1     MARKQUIS NOWELL         50     9     0     6     3     2     0     0     0     2     4     1

B     BAYLOR SCHEIERMAN       40     0     0     2     0     2     0     0     0     5     6     1
B     RYAN NEMBHARD           35     7     2     0     1     2     0     0     0     2     4     3
B     NAE'QWAN TOMLIN         20     7     3     0     0     1     1     1     0     1     1     0
B     SIR'JABARI RICE         35    12     1     0     3     0     1     1     0     2     1     0
B     TREY GALLOWAY           15     0     0     0     0     0     0     0     0     0     1     0
TOT                          350    81    25    15     7    10    10     4     0    29    27     6
```

Fig. 6. Segment 14 PBP and Final Box Score for Simulation 1

The POSS column is a stand-in for minutes played (out of 40) in real box scores. With possessions batched into segments, and no substitutions within them for simplicity's sake, all players will have a value that is a multiple of five. POSS can be divided by 70 and multiplied by 40 to provide a minutes approximation, functionally assuming unvarying possession lengths of 17.1 seconds. A given player's POSS, or playing time, does not vary from simulation to simulation unless the rotation .csv file is altered. This constant playing time distribution is optimal for running large amounts of simulations and analyzing the results, as only the random element can be responsible for the variance. To demonstrate this, consider Princeton guard Ryan Langborg's 16 points scored in Simulation 1. In most simulations with the same specifications, he totals somewhere in the mid-single digits. What luck needs to go his way to get to 16?

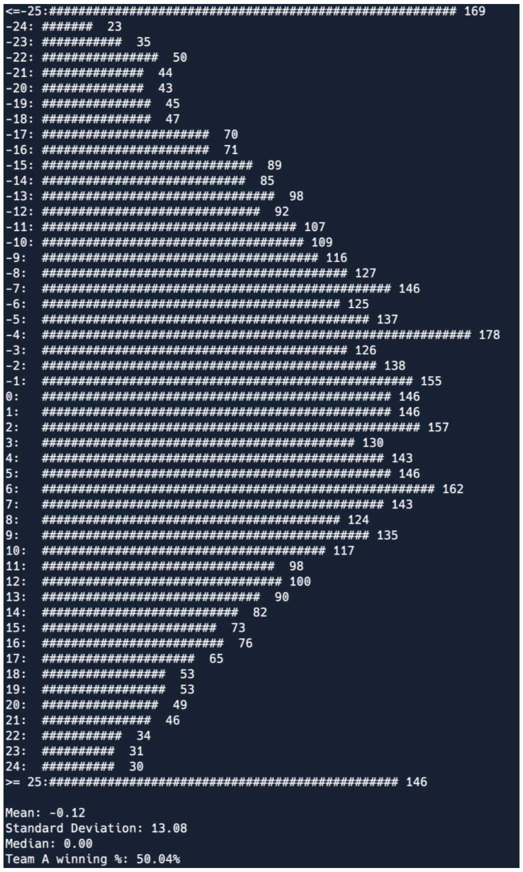
Langborg's average usage rate in the input statistics was 19.9 percent, the second lowest on Team A. This will be scaled down to about 17.5 percent in normalize() no matter who he is on the floor with. He plays 30 possessions total, so his expected value of possessions in a simulation is 30 * .175 = 5.25. But in this 16 point outing, he uses 9 possessions. Just by chance, nextDouble() landed in his region 30 percent of the time against a probabilistic value of 17.5. Now that the possessions are his, he also needs random luck to avoid misses and turnovers. He does so, with 4 out of 9 becoming made three-pointers even though these only represent 8 out of his 53 possessions in the true data. That is good for 15 percent, meaning he only should've expected 1.36 three-point makes over nine possessions.

This potential for variance in any given simulation applies to every player except for Kenan Blackshear, the only player involved to have only one game of stats. With a much smaller basket of possessions to choose from, a high usage rate, and the non-repetition of possessions– used ones are removed from the player's available pool until the pool is emptied and it resets– Kenan's stat line was more or less deterministically plugging turnovers into the box score during every simulation. This was a good way to learn that with the current program implementation, a small sample size can corrupt the simulation, or at least make a given ßplayer's game production much more predictable than others. This works in the positive direction too; Dylan Disu played remarkably, but only in two games; he didn't have a 3rd and 4th to cool off and regress to the mean. He had 30 possessions that ended in two-point shot attempt possessions, and only two that didn't. Shooting 22 for 30 on those two-pointers, Dylan is expected to be worth about 1.4 points per possessions on average; this is a terrific number to begin with, and he even exceeded it in Simulation 1.

In the initial playing time rotation, Kenan was playing 5 segments, or 25 possessions total. Over the course of 5000 simulations, his Team A averaged a two point deficit to Team B and won around 40 percent of the time. His time was reduced to 2 segments, or 10 possessions, for the next 5000-game simulation. This change brought the average point deficit back to zero, essentially locking the two teams in a dead heat. The results of this simulation are shown as a histogram in Figure 7. The positive numbers in the bottom half represent Team A wins, as Team B's score was subtracted from Team A's.

It is noted that 69.8 percent of game outcomes were within one standard deviation (13 points) on either side of the mean (0). This means the score differences essentially follow a normal distribution, which is an interesting result. Game logic would have to change if a different type of distribution was desired.

While this preliminary fictional game simulation algorithm produces acceptable box scores, there is plenty of room to incrementally make it more realistic. Most importantly, the scoring efficiency performance of each player should not exist completely independently of his teammates. Creative methods of hypothesizing how and why specific pairs or groups of players would or wouldn't mesh together are needed, and these results should carry over into the algorithm accordingly.



```
<=-25:######################################################## 169
-24: #######  23
-23: ###########  35
-22: ################  50
-21: ##############  44
-20: ##############  43
-19: ###############  45
-18: ###############  47
-17: #######################  70
-16: #######################  71
-15: #############################  89
-14: ############################  85
-13: ################################  98
-12: ##############################  92
-11: ################################### 107
-10: #################################### 109
-9:  ###################################### 116
-8:  ########################################## 127
-7:  ################################################ 146
-6:  ######################################### 125
-5:  ############################################# 137
-4:  ########################################################### 178
-3:  ########################################## 126
-2:  ############################################## 138
-1:  ################################################### 155
0:   ################################################ 146
1:   ################################################ 146
2:   #################################################### 157
3:   ########################################### 130
4:   ############################################### 143
5:   ################################################ 146
6:   ###################################################### 162
7:   ############################################### 143
8:   ######################################### 124
9:   ############################################# 135
10:  ####################################### 117
11:  ################################  98
12:  ################################# 100
13:  ##############################  90
14:  ###########################  82
15:  ########################  73
16:  #########################  76
17:  #####################  65
18:  #################  53
19:  #################  53
20:  ################  49
21:  ###############  46
22:  ###########  34
23:  ##########  31
24:  ##########  30
>= 25:################################################ 146

Mean: -0.12
Standard Deviation: 13.08
Median: 0.00
Team A winning %: 50.04%
```

Fig. 7. Point Differential Distribution From 5000 Simulations After Rotation Change

References

[1] J. H. Fewell, D. Armbruster, J. Ingraham, A. Petersen, and J. S. Waters, "Basketball teams as strategic networks," PLoS ONE, vol. 7, no. 11, Art. no. e47445, 2012. https://doi.org/10.1371/journal.pone.0047445