Player Centrality on NBA Teams

Sohum Misra (sohummm@stanford.edu)
Stanford University SCPD Student

1. Introduction

The NBA is entering its golden age of statistical analysis with teams all over the league trying to employ novel techniques to gain an edge on their competitors. Although it is not as statistically motivated as baseball, the league has shifted its focus to advanced statistics to be able to determine the value of a player that may not be obviously tangible.

The goal of this project and paper is to build on work done in this domain by applying some concepts of graph theory to the problem. I will be constructing many network models of NBA line-ups and will try to optimize different metrics to determine node centrality. The goal of the project is to (a) determine who the best players in the NBA are according to these network models, (b) determine who the most important players are on each team and (c) compare these findings to some of the most popular advanced statistics that are used by teams today.

The data-set for this project is NBA line-up data for the last five completed seasons (beginning from the 2012-13 season up to the 2016-17 season).

2. Related Work

2.1 Five-man line-up analysis

This kind of network analysis to evaluate player performance and centrality in the NBA is based on *Evaluating Basketball Player Performance via Statistical Network Modeling* by Piette et al. [1] In that paper, the authors designed this methodology for evaluating player performance and used it to determine the most important players in four NBA seasons (2006-07 to 2009-10).

The authors used three variations of efficiency as their weighting function: offensive efficiency, defensive efficiency and total efficiency[1] and computed the weighted eigenvector scores for each player to determine the most central nodes in the graph. One of the findings was that the results of node centrality heavily favored those players who had many neighbors (i.e. players who were part of many line-ups). To provide context for this, the authors constructed a bootstrap distribution using the actual data to determine p-values for their scores.

2.2 Outcome-based analysis

Two other papers investigated similar usages of networks in the space of NBA player performances, but did not construct graphs using five-man line-ups. Instead, in *Basketball Teams as Strategic Networks*, Fewell et al. defined player centrality as a measure of how frequently players participated in a path in a bipartite graph of players and outcomes

[1] Efficiency is defined as the number of points scored (or allowed, for defensive efficiency) per possession.

(baskets made, missed, etc.). [2] This data was used to determine optimal path lengths to optimize for certain outcomes.

Similarly, in *PageRank Model for Player Performance Assessment in Basketball, Soccer and Hockey*, Brown took this model further by introducing edges for outcomes other than just a basket being scored or missed, such as steals and blocks. [3] Brown's model defined a goal node as the scoring of a basket and used the PageRank algorithm to determine the players who optimized the path to the goal.

Unfortunately, though I had initially planned to, I was unable to expand on these models as granular pass-by-pass data is not available from the NBA. As a result, I would have only be able to create edges for players making or missing shots, which would result in a very trivial graph.

3. Data Collection

The official statistical website for the NBA, stats.nba.com, offers a variety of statistics for consumption via publicly callable JSON APIs. My initial plan was to use existing Python libraries that do this[2] but upon beginning the project, it became evident that they did not provide an API for the line-ups data that I needed. As a result, the first step was to write my own Python module that hit the NBA stats service to pull the line-ups data for the last five seasons.

The raw data was a set of serialized JSON objects that stored the players involved in a given line-up as well as many relevant statistics such as minutes played, win/loss, points scored, field goals made and attempted, etc. Of these, I pulled the fields that were relevant to my weighting functions and stored them in a Python dictionary and then serialized it to disk.

The final data set comprised of 812 players and 36045 line-ups.

4. Method

4.1 Model

From the data-set, a bipartite graph is constructed where nodes are either players or five-man units that play together. Weighted edges are drawn between player nodes and line-up nodes where the weight is one of the six metrics defined in 4.2 as providing insight into how effective a player is. As a result, each of these weighting functions results in a different bipartite graph, all with the same edge-set but with different weights.

Next, a weighted adjacency matrix, W is constructed where $W_{i,j}$ represents the weight between player i and line-up j. The transpose of this matrix is multiplied by itself ($W^T W$) to form a new adjacency matrix that contains edges between player nodes, where the weights represent the sum of all the weights for line-ups containing both those players. For example, if players A and B belong to line-ups $L1$ and $L2$, the weight between A and B is the sum of the weights of *(A, L1), (A, L2), (B, L1), (B, L2)*.

[2] https://github.com/bradleyfay/py-goldsberry and https://github.com/ethanluoyc/statsnba-playbyplay were two such libraries.

Finally, I will compute a variety of node centrality scores for the players against each graph. These scores were used to rank players, both overall and within their team.

4.2 Edge weighting functions

The following weight functions will be used to consruct the bipartite graph. The initial edge weight between players and five-mine line-ups will be the calculation of each weight function for the line-up.

- Assist to turnover ratio[3]
- Field-goal percentage[4]
- Offensive rebound percentage[5]
- Steals/possessions ratio[6]
- Opponent field-goal miss percentage[7]
- Defensive rebound percentage[8]

4.3 Algorithms

Two node centrality algorithms will be used to calculate the importance of each player within the networks: *PageRank* and *eigenvector centrality*. The mathematical definitions are provided here for completeness, but I will use the SNAP.PY and NetworkX[9] libraries to define my networks and compute my scores.

4.3.1 PageRank

PageRank (*PR*) is a measure of the most important nodes in a network using the concepts of in-links and out-links. A node's *PR* is the sum of *PR* of all the source nodes of in-links and a node's *PR* is divided equally between all the destinations of out-links of a node. Since this definition is recursive, the solution is computed by following these steps from [4]:

- Initially, assign all n nodes the same initial PageRank, set to $1/n$.
- Choose a number of steps k.
- Perform a sequence of k updates to the PageRank values. Each node divides its current PageRank equally across its out-links and passes these equals shares to the nodes it points to. Each node updates its new PageRank to be the sum of the shares it receives.

---

[3] An assist is a pass that directly leads to a basket being made. A turnover is when the ball changes possessions without a shot being attempted.
[4] A field goal means a shot. Field goal percentage is the percentage of shots made given shot attempts.
[5] An offensive rebound is a ball rebounded by the team on offense. Offensive rebounding percentage is a measure of how many of their own missed shot attempts the team on offense rebounded.
[6] A steal is when a defensive player dispossesses the offensive player.
[7] Opponent FG miss percentage is the percentage of shots missed given shot attempts by the opponent.
[8] A defensive rebound is a ball rebounded by the team on defense, after the offense misses a shot. Defensive rebounding percentage is a measure of how many of the opponent's missed shots the team on defense rebounded.
[9] NetworkX library can be found at https://networkx.github.io/.

For some value of k, this algorithm will reach an equilibrium where the shares for each node do not change. These final shares are the *PR* for each node.

4.3.2 Eigenvector centrality (eigencentrality)

Eigenvector centrality is a more generalized form of the PageRank algorithm and is a measure of both how well-connected a node is (i.e. its degree) as well as the quality of those connections. [5] Given an adjacency matrix A, a node's centrality can be defined as the average of the centralities of the node's neighbors:

$$x_i = \frac{1}{\lambda}\sum_{j=1}^{n} A_{ij}x_j \Rightarrow \lambda x = A \cdot x$$

The solution to the above equation, x, is an eigenvector of A with eigenvalue λ. As described in *The mathematics of networks* by Newman, the eigenvector corresponding to the largest eigenvalue is the one containing the node centrality scores. [5]

5. Results & Findings

5.1 Evaluation basis

The results of computing the algorithms on the networks I defined will be compared against the following advanced statistics that are in use in the league today:

Rank	PER[10]	WS/48[11]	VORP[12]
1	Kevin Durant	Kevin Durant	LeBron James
2	LeBron James	Chris Paul	Russell Westbrook
3	Russell Westbrook	LeBron James	Stephen Curry
4	Anthony Davis	Stephen Curry	James Harden
5	Chris Paul	James Harden	Kevin Durant
6	Stephen Curry	Kawhi Leonard	Chris Paul
7	James Harden	Russell Westbrook	Kyle Lowry
8	Hassan Whiteside	Hassan Whiteside	Kawhi Leonard
9	Karl-Anthony Towns	Anthony Davis	Jimmy Butler
10	Demarcus Cousins	Rudy Gobert	Damian Lillard

Table 1: PER, WS/48 and VORP of all NBA players with >= 160 games played in 2012-13 to 2016-17 seasons

5.2 Per team results

The first approach I is to calculate a graph per team for each edge weight function and run analysis on the network. **Tables 2** and **3** summarize the results. There is significant overlap compared to the evaluation bases in **Table 1**, however there are also some notable omissions: Kevin Durant and LeBron James. These two players appear in the top 5 of all our evaluation tables, but do not appear in the calculated rankings.

The reason for this is that these two players played on two different teams during the evaluation period. As a result, only the better of their two teams makes the cut to the final

[10] PER leaders for 2012-13 to 2016-17 seasons can be found at http://bkref.com/tiny/8Ig9G.
[11] WS/48 leaders for 2012-13 to 2016-17 seasons can be found at http://bkref.com/tiny/VentI.
[12] VORP leaders for 2012-13 to 2016-17 seasons can be found at http://bkref.com/tiny/MTE40.

table. Thus, these computations favor players who have long tenures with teams, which is why nearly every player in these tables has been on the same team for the last five seasons.

Unfortunately, there is no reasonable way to combine this information because a player's importance to one team is correlated to the other players on that team. For example, Deron Williams has played on three teams in the last 5 seasons: Brooklyn, Dallas and Cleveland. In Brooklyn he was the second-best player on the team for three seasons whereas in Dallas he was much worse. However, his teammates in Brooklyn were not elite players, thus his centrality score was very high. Since each team has its own network, the centrality scores are measures of the player's importance to that specific network. As they shift teams, that level of importance might change.

#	AST/TO	FG%	OReb %	Stl/Poss	Opp Miss FG%	DReb %
1	D Lillard	J Crawford	J Crawford	J Harden	J Crawford	J Harden
2	J Crawford	D Lillard	J Harden	D Green	D DeRozan	D Lillard
3	S Curry	J Harden	D Lillard	D Lillard	J Harden	J Crawford
4	K Walker	D DeRozan	T Thompson	D DeRozan	D Lillard	G Antetokounmpo
5	R Westbrook	K Walker	K Walker	J Crawford	K Walker	C Anthony
6	D DeRozan	S Curry	A Drummond	K Walker	J Butler	K Walker
7	J Harden	D Green	PJ Tucker	PJ Tucker	D Green	T Thompson
8	K Lowry	J Wall	D DeRozan	J Butler	S Curry	PJ Tucker
9	T Ross	J Butler	C Anthony	G Hayward	PJ Tucker	K Lowry
10	D Green	K Thompson	R Westbrook	A Drummond	K Lowry	J Wall

Table 2: PageRank scores of each network

#	AST/TO	FG%	OReb %	Stl/Poss	Opp Miss FG%	DReb %
1	D Lillard	J Harden	C Anthony	J Harden	J Harden	J Crawford
2	K Walker	D Lillard	J Harden	D Lillard	K Walker	D Lillard
3	C Anthony	J Crawford	T Thompson	PJ Tucker	J Crawford	D DeRozan
4	A Davis	G Ant'nmpo	J Crawford	K Walker	D Lillard	J Harden
5	K Lowry	C Anthony	K Walker	D DeRozan	G Ant'nmpo	J Butler
6	D Nowitzki	K Walker	PJ Tucker	G Hayward	C Anthony	K Walker
7	M Conley	J Wall	G Ant'nmpo	G Ant'nmpo	A Davis	G Hayward
8	R Westbrook	J Johnson	D Lillard	D Green	T Thompson	K Lowry
9	J Harden	A Davis	A Drummond	A Drummond	J Wall	D Green
10	J Wall	D Nowitzki	R Westbrook	H Thompson	PJ Tucker	K Thompson

Table 3: Eigencentrality scores of each network

#	Average PageRank	Average Eigencentrality
1	Jamal Crawford	James Harden
2	Damian Lillard	Damian Lillard
3	James Harden	Kemba Walker
4	DeMar DeRozan	Giannis Antetokounmpo
5	Kemba Walker	Carmelo Anthony
6	Draymond Green	Jamal Crawford
7	Jimmy Butler	John Wall
8	Stephen Curry	Tristan Thompson
9	Kyle Lowry	Kyle Lowry
10	Gordon Hayward	Anthony Davis

Table 4: Average page rank and eigencentrality scores

5.3 Normalized Edge Weights

In this general model, "[a node's] centrality score [is] artificially inflated by having many neighbors." [1] Thus, a player who has been in the team a long time and thus has had lots of

neighbors (i.e. players who have come and gone) will have an inflated centrality score. One potential way to normalize this is by boosting players who have been at a team for a lesser time and penalize players who have been in a team a long time.

In this exploration, I will build another set of networks similar to 5.2 except instead of using the raw calculated edge weights, I will increase or decrease the weight depending on the average tenure in that team. For example, if the average tenure in a team is two seasons, then everyone below 2 seasons will get a slight boost and everyone greater than 2 seasons will get a slight penalty. It will be important to dampen the boost effectively such that poor players who were with the team a very short time do not get an artificially high boost.

The general function for boost will normalize each player's contribution to the average, capping the boost by a percent α. So, if a player has played one season for a team and the average tenure is 2, instead of doubling that player's edge weight, we will multiply it by $1 + \frac{\alpha}{100}$.

#	$\alpha = 10$	$\alpha = 20$	$\alpha = 30$	$\alpha = 40$	$\alpha = 45$
1	J Crawford	J Crawford	J Crawford	D Lillard	D Lillard
2	D Lillard	D Lillard	D Lillard	J Crawford	J Crawford
3	J Harden	J Harden	J Harden	D DeRozan	D DeRozan
4	D DeRozan	D DeRozan	D DeRozan	J Harden	J Harden
5	K Walker	K Walker	D Green	D Green	D Green
6	D Green	D Green	K Walker	K Walker	K Walker
7	J Butler	J Butler	J Butler	S Curry	S Curry
8	S Curry	S Curry	S Curry	J Butler	J Butler
9	K Lowry	G Hayward	G Hayward	R Westbrook	G Hayward
10	G Hayward	K Lowry	R Westbrook	G Hayward	R Westbrook

Table 5: Average page ranks with changing α

#	$\alpha = 10$	$\alpha = 20$	$\alpha = 30$	$\alpha = 40$	$\alpha = 45$
1	J Harden	J Harden	J Harden	J Harden	J Harden
2	K Walker	K Walker	K Walker	J Crawford	J Crawford
3	D Lillard	D Lillard	D Lillard	D Lillard	D Lillard
4	G Antetokounmpo	J Crawford	J Crawford	K Walker	K Walker
5	J Crawford	T Thompson	T Thompson	J Wall	D DeRozan
6	C Anthony	G Antetokounmpo	J Wall	T Thompson	J Wall
7	T Thompson	J Wall	A Davis	D DeRozan	T Thompson
8	J Wall	A Davis	D DeRozan	A Davis	A Davis
9	A Davis	C Anthony	R Westbrook	D Green	D Green
10	K Lowry	D DeRozan	K Lowry	R Westbrook	S Curry

Table 6: Average eigencentralities with changing α

From the results in **Tables 5** and **6**, it does not look like normalizing the nodes in this way gets us much more gain in terms of boosting players who have fewer neighbors. One can conclude thus that this model will suffer from this constraint.

5.4 Best players by team

In addition to overall numbers, I also looked at the best players by total page rank and eigencentrality by team, and compared it to the expected best players according to PER, WS/48 and VORP. For each team, I counted the model's value as correct if the player appeared in any of the advanced statistics.

The full data is summarized in **Table 7** below, but the models performed mediocrely. The PageRank scoring system was correct 57% of the time while the eigencentrality scoring was 67% accurate.

Team	PER	WS/48	VORP	PageRank	Eigencentrality
ATL	A Horford	A Horford	P Millsap	D Schroder	D Schroder
BKN	B Lopez	B Lopez	B Lopez	**B Lopez**	**B Lopez**
BOS	I Thomas	I Thomas	I Thomas	A Bradley	A Bradley
CHA	A Jefferson	C Zeller	K Walker	**K Walker**	**K Walker**
CHI	J Butler	J Butler	J Butler	**J Butler**	**J Butler**
CLE	L James	L James	L James	T Thompson	T Thompson
DAL	D Nowitzki	D Powell	D Nowitzki	**D Nowitzki**	**D Nowitzki**
DEN	K Faried	K Faried	K Faried	**K Faried**	**K Faried**
DET	A Drummond	A Drummond	A Drummond	**A Drummond**	**A Drummond**
GSW	S Curry	S Curry	S Curry	D Green	**S Curry**
HOU	J Harden	J Harden	J Harden	**J Harden**	**J Harden**
IND	P George	G Hill	P George	**P George**	**P George**
LAC	C Paul	C Paul	C Paul	J Crawford	J Crawford
LAL	K Bryant	J Hill	K Bryant	J Clarkson	J Clarkson
MEM	M Conley	M Conley	M Gasol	Z Randolph	**M Conley**
MIA	H Whiteside	C Bosh	D Wade	**C Bosh**	**C Bosh**
MIL	G Monroe	G Monroe	G Ant'nmpo	**G Ant'nmpo**	**G Ant'nmpo**
MIN	K Towns	K Towns	K Towns	R Rubio	R Rubio
NOP	A Davis	A Davis	A Davis	**A Davis**	**A Davis**
NYK	C Anthony	C Anthony	C Anthony	**C Anthony**	**C Anthony**
OKC	K Durant	K Durant	R Westbrook	**R Westbrook**	**R Westbrook**
ORL	N Vucevic	N Vucevic	N Vucevic	**N Vucevic**	**N Vucevic**
PHI	N Noel	N Noel	N Noel	H Thompson	H Thompson
PHX	E Bledsoe	G Dragic	E Bledsoe	PJ Tucker	A Len
POR	L Aldridge	D Lillard	D Lillard	**D Lillard**	**D Lillard**
SAC	D Cousins	D Cousins	D Cousins	B McLemore	B McLemore
SAS	K Leonard	K Leonard	K Leonard	P Mills	P Mills
TOR	K Lowry	K Lowry	K Lowry	D DeRozan	**K Lowry**
UTA	R Gobert	R Gobert	G Hayward	**G Hayward**	**G Hayward**
WAS	J Wall	M Gortat	J Wall	**J Wall**	**J Wall**
Total	-	-	-	**17/30 (57%)**	**20/30 (67%)**

Table 7: Top players by team according to PER, WS/48, VORP and models

6. Conclusion

This project explored the modeling of NBA player statistics via a directed network to determine the best players in the league, based on the work done by Piette et al in [1]. While the model does provide some overlap with existing advanced metrics of player importance, there are several ways that it is an inaccurate representation of the problem space.

First, my initial network stuck every single line-up and player into a single bipartite graph and attempted to reduce it. Unfortunately, this did not yield very accurate results, since it promoted players who had been a part of many teams to the top, since these players were essentially the central nodes between the clusters representing the actual players on a team.

Thus, I instead elected to model each team as a separate network. The result of this was that the rankings I got were decent per team (57% and 67% accurate). Empirically looking at defensive metrics, the better defenders in a team were ranked higher while the same was true of offensive metrics. However, since these numbers were in context to each specific team,

comparing them overall did not result in a good comparison. A player on a more balanced team would have a dampened overall score whereas a player on a less balanced team would have an inflated score, because their centrality in the team would be more or less distributed.

Another problem with the model is that nodes with high degrees bubble to the top. This was specifically true of Jamal Crawford, who played significant minutes for the Los Angeles Clippers. By most statistical and empirical metrics, Jamal Crawford would be rated the third best player on the team after Chris Paul and Blake Griffin. Instead, within this model, Jamal Crawford rated as not only the best player in his own team, but an overall top 10 player in the league. This could be due to injuries (both Paul and Griffin have missed significant time in the last few seasons), thus Crawford spent more time out there with more lineups, thus his node degree was higher.

That said, this was a good model of the most important players on a team by their actual availability. While talent and skill is important, being really talented is of no use if you are always injured.

7. Further Work

Although it is out of the scope of this project, it would be interesting to continue building on this model to try and fix the problems outlined above. One way to counter the "good player on a bad team" modeling would be to look at the average scores on the team and provide a boost for players on teams that had a high average score.

To normalize nodes with high degrees, I could project probabilistic edges onto the graph for players that had missed significant time based on their performance in line-ups that they were present. Alternatively, I could filter out games that did not involve the top n players on a team based on some predicate (for example the top 5 in terms of efficiency), so that only games where most of the premier players were available are counted.

8. References

1. Piette J, Anand S, Pham L. Evaluating Basketball Player Performance via Statistical Network Modeling. In: MIT Sloan Sports Analytics Conference; 2011. Available from: http://www.sloansportsconference.com/?p=2840.
2. Fewell JH, Armbruster D, Ingraham J, Petersen A, Waters JS. Basketball Teams as Strategic Networks. PLoS ONE. 2012 11;7(11):e47445. Available from: http://dx.doi.org/10.1371/journal.pone.0047445.
3. Brown S. A PageRank Model for Player Performance Assessment in Basketball, Soccer and Hockey. In: MIT Sloan Sports Analytics Conference; 2017. Available from: http://www.sloansportsconference.com/wp-content/uploads/2017/02/1494.pdf.
4. Easley, D., & Kleinberg, J. (2010). Chapter 19: Cascading behavior in networks. Networks, crowds, and markets: Reasoning about a highly connected world (pp. 79—83). Cambridge: Cambridge University Press.
5. Newman M (2008) Mathematics of networks. The New Palgrave Encyclopedia of Economics.