Using Graph Networks to Optimize NBA Games

Tiffany Chen
Cornell Tech
New York, New York
Email: twc75@cornell.edu

Max Dumas
Cornell Tech
New York, New York
Email: mfd64@cornell.edu

***Abstract*—Currently, the schedule of games for teams in the National Basketball League (NBA) is created based on a variety of factors by an executive committee. This schedule is largely dictated by the structure of the NBA's teams into divisions and conferences. Within that structure, matchups are decided qualitatively. We demonstrate how a graph network can assist in the scheduling of NBA games with the goal of decreasing travel time for each team and increasing attendance at games. Our results show how simple clustering to optimize conference and division assignment can reduce team travel time variance by 7.4% and decrease mean travel time. We also show that max-flow algorithms (Edmonds-Karp) used with historical attendance data can be used to determine which games will be most popular.**

I. Introduction

Currently, NBA teams have a mix between home games and away games as part of their regular season schedule. The teams play their peers in their own conference and division the most. However, this can result in long travel times for teams if conferences and divisions are not created with minimizing travel time in mind. Longer travel time means shorter practice and recovery time which can ultimately lead to more injuries for players. In addition, reducing travel time can reduce the carbon footprint that NBA teams generate. We use graph networks to determine the optimal assignment of teams to conferences and divisions for the NBA regular season which minimizes the distance those teams must travel.

In addition, certain teams and arenas tend to draw more attendance. In the 2021-2022 season, game attendance remained 5% lower compared to the 2018-2019 season (the last full season unaffected by COVID-19). Viewership has also been steadily decreasing. The NBA is becoming increasingly concerned that their popularity is waning. Therefore, we use graph networks to determine what is the configuration of games that should be played to maximize attendance in a given day, such as a holiday, where high attendance is critical to maximizing advertising and ticket revenue.

A. Problem Formulation

For the travel distance minimization problem, we formulate the network as an undirected weighted graph between the locations of NBA teams in North America. The goal is to minimize the miles traveled per team by recommending new conferences and divisions. To accomplish this, we use an open source implementation of the Node2Vec algorithm to vectorize the topography of the team locations, and then leverage K-Means and spectral clustering to generate clusters with centroids that minimize travel distance.

We perform two phases of clustering to match the structure of the NBA. First, we partition the full set of teams into two sets, to represent the two conferences of the NBA. Second, we cluster each conference into three divisions.

For the attendance problem, we model the network as a bipartite graph between teams, with average attendance as flow. The "left" set of nodes is NBA teams and the "right" set of nodes are the cities that games are played in. This describes the matchups, with the left node being the away team and the home team corresponding to the city in the "right" node. Using the average attendance at away games, this graph will tell us which team should play in which city on a given day to maximize possible attendance. This can be useful because currently, the NBA employs an individual whose entire job is to create the schedule. This can help automate the process and remove possible human-error. In addition, this can be used to plan games during days such as holidays or opening week which tend to draw more viewership and attendance.

B. Approach

1) Data Collection: Before beginning analysis, we must collect prerequisite data. We did the following:

1) Identify data that needs to be collected: This includes things like the teams, the locations of their home courts, and the historical attendance record of the teams in previous years.
   We did the following to collect team location data:
   a) Find the list of teams in each conference and division.
   b) Find their home cities.
   c) Find the latitude/longitude of the stadiums in their home cities.
   d) Project these points onto a map of North America.
   e) Use geopandas to get pairwise distances between all of the locations.
2) Archive the teams dataset for later usage. The final teams dataset can be found here. It is also attached at the end of this document.
3) We did the following to collect attendance data:
   a) Find and compile data on 2018-2019 NBA season attendance for all games (both home and away).

We obtained this data from the Basketball Reference site [1].
   b) Aggregate data across each team and game location to obtain the mean attendance across all games at each Away team-city pair.
4) Archive the attendance dataset for later usage. The final attendance dataset can be found here.

2) Travel Distance Minimization: Next, we used our team location data to identify new conferences and divisions that the NBA could use to minimize the travel distance for teams as they are playing within a division and within a conference. We did the following:

1) Create a complete graph of all teams with their pairwise distances used as edge weights.
2) Use Node2Vec to obtain topological embeddings of the graph. We used dimensions=8, walk_length=10, num_walks=1000 as our hyperparameters for generating the embedding. We obtained these values after performing a hyperparameter search through 2,500 different parameter combinations, and discovering these parameters as yielding the optimal embedding for our use case.
3) Perform Spectral clustering with 2 clusters to partition the teams into two conferences according to least distance traveled.
4) Within each conference, use K-Means clustering to find 3 clusters which minimize travel distance within each division.

The method above will minimize travel distance both within a given conference and within given a division, which will reduce overall travel distances.

3) Algorithm for Spectral Clustering: [2]

1) Calculate the Laplacian L (or the normalized Laplacian)
2) Calculate the first k eigenvectors (the eigenvectors corresponding to the k smallest eigenvalues of L)
3) Consider the matrix formed by the first k eigenvectors; the l-th row defines the features of graph node l
4) Cluster the graph nodes based on these features (e.g., using k-means clustering)

4) Algorithm for K-Means: [3]

1) Assign each observation to the cluster with the nearest mean: that with the least squared Euclidean distance where each x_p is assigned to exactly one $S^{(t)}$
$$S_i^{(t)} = \{x_p : \|x_p - m_i^{(t)}\|^2 \leq \|x_p - m_j^{(t)}\|^2 \forall j, 1 \leq j \leq k\}$$
2) Recalculate means (centroids) for observations assigned to each cluster.
$$m_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{x_j \in S_i^{(t)}} x_j$$

5) Attendance Maximization: To determine the best configuration of games to maximize attendance, we used our attendance data to do the following:

1) Create a bipartite graph with NBA teams on one side and home cities on the other.
2) Set average attendance for each game configuration as the capacity for each edge.
3) Add a source and target with the average attendance for each team as the source edge capacity and the average attendance for each city being the target edge capacity.
4) Run a max-flow algorithm (aka Edmonds-Karp which is a variation of the Ford-Fulkerson algorithm).
5) Walk backwards through the graph from the target node to identify the top N matchups in terms of expected attendance.

The method above will maximize attendance because it maximizes the flow of attendance for any given day. This provides the flow maximizing attendances for each pairing of team and city. For background, a bipartite graph a set of graph vertices which are split into two disjoint and independent sets in which no two graph vertices in the same set are adjacent and every edge will connect a vertex in set 1 to a vertex in set 2. With maximum bipartite matching, it will return a configuration that contains as many edges as possible.

6) Algorithm for Ford-Fulkerson: [4]

Constraints The flow along an edge cannot exceed its capacity.
$$\forall (u, v) \in E : f(u, v) \leq c(u, v)$$

Flow Conservation The net flow to a node is zero, except for the source, which ”produces” flow, and the sink, which ”consumes” flow.
$$\forall u \in V : u \neq s \text{ and } u \neq t \Rightarrow \sum_{w \in V} f(u, w) = 0$$

Value(f) The flow leaving from s must be equal to the flow arriving at t.
$$\sum_{(s,u) \in E} f(s, u) = \sum_{(v,t) \in E} f(v, t)$$

Inputs Given a Network $G = (V, E)$ with a flow capacity c, a source node s, and a sink node t

Output Compute a flow f from s to t of maximum value

1) $f(u, v) \leftarrow 0$ for all edges (u, v)
2) While there is a path p from s to t in G_f, such that $c_f(u, v) > 0$ for all edges $(u, v) \in p$:
   a) Find $c_f(p) = \min c_f(u, v) : (u, v) \in p$
   b) For each edge $(u, v) \in p$
      i) $f(u, v) \leftarrow f(u, v) + c_f(p)$ (*Send flow along the path*)
      ii) $f(v, u) \leftarrow f(v, u) - c_f(p)$ (*The flow might be ”returned” later*)

The Edmonds-Karp algorithm follows Ford-Fulkerson except that the search order when finding the augmenting path is defined and the path is found using breadth-first-search. [5]

C. Analysis and Numerical Results

1) Travel Distance Minimization: There are 30 NBA teams with 2 conferences (East and West). Within each conference, there are 3 divisions of 5 teams each. The divisions for the

Eastern Conference are: Atlantic, Central, and Southeast. The divisions for the Western Conference are: Northwest, Pacific, and Southwest. See Appendix I or here for the complete dataset of the teams. A map of the connections between teams and their relative edge weights is visualized in Figure 1.

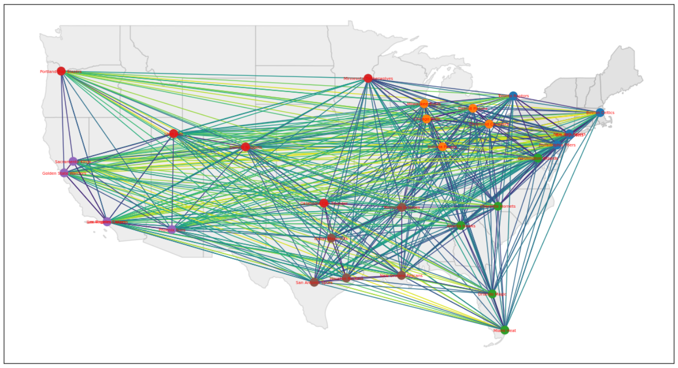


Fig. 1. Visualization of NBA teams and their locations. Edge weights are colored according to distance. Brighter colors indicate longer distances.

We first generate 2 clusters for the two conferences of the NBA: Eastern and Western. We generate these clusters using Spectral clustering, as it produces better results than K-Means for this phase. We then generate our divisions clusters within each conference. This ensures that we satisfy the constraint that divisions must be fully contained within a single conference. We generate 3 clusters within each conference using a constrained K-Means implementation [6] which allows us to specify that each cluster must contain at least 4 nodes and at most 6 nodes. This ensures that, while every division will not contain the same number of teams, each will at least contain a comparable number of teams, with no divisions having only a single team or many more than 5 teams.

The results of the conference clustering can be seen in Figure 2a. Our results largely reproduce the conferences as they currently exist, with two notable shifts. The Memphis Grizzlies and New Orleans Pelicans switch from the Western Conference to the Eastern Conference. This proves advantageous when generating divisions.

Generated divisions can be seen in Figure 2b. We see a number of small changes that generally result in more cohesive choices of divisions. The New Orleans Pelicans and Memphis Grizzlies, now part of the Eastern Conference, are placed into a division with the rest of the Deep South. The Oklahoma City Thunder joins the nearby Texas teams to form their own division. Toronto forms a division with the rest of the Great Lakes Cities instead of joining the Atlantic division, which now includes Charlotte. One questionable result is that the Sacramento Kings trade places with the Portland Trail Blazers in the Central division, but this is likely purely to avoid violating the constraint of a minimum of four teams per division.

The distribution of travel distances across matches within divisions can be seen in Figure 3. The new assignment of divisions represents a minor improvement in travel distances. Across all divisions we see a 4.4% reduction in travel distance, with a 7.4% reduction in travel distance variance.

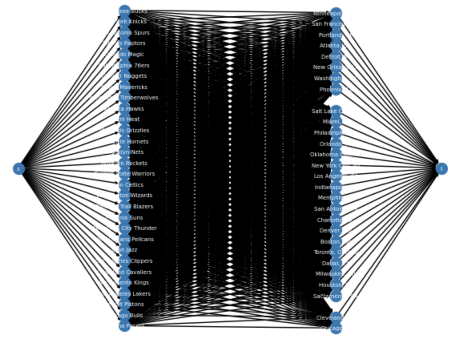


Fig. 4. Visualization of bipartite graph for NBA teams matched with possible game locations.

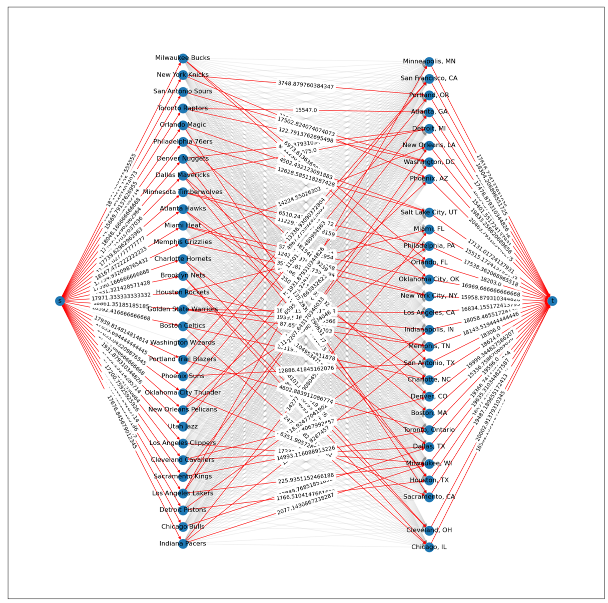
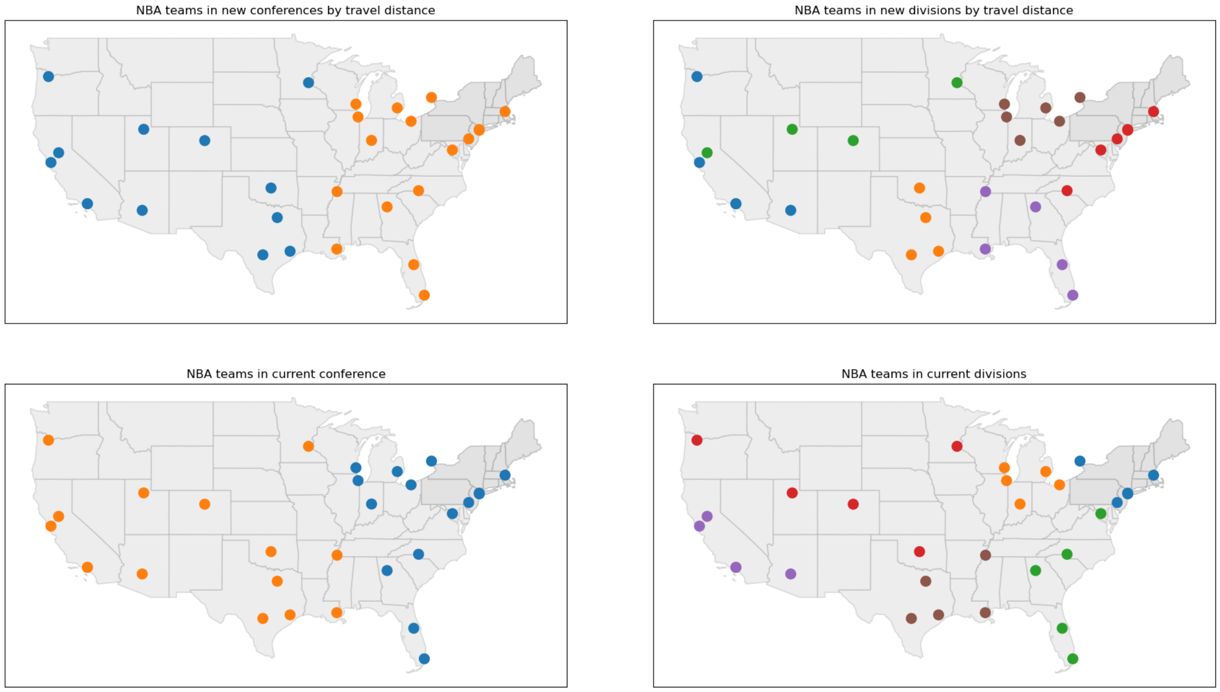


Fig. 5. Flow diagram showing which games should be played at which locations in order to maximize attendance.

2) Attendance Maximization: We also developed a method for the NBA to use historical attendance data to find the combination of games in which cities for all the NBA teams which will result in the highest attendance. To do so, we gathered historical game attendance data from the 2018-2019 season. We then created a bipartite graph with the average attendance for each team and location represented as the flow. Then, we used the Edmonds-Karp algorithm to determine which combinations of teams and locations would result in the highest attendance on a given day for in the form of Away games. The flow diagram can be seen in Figure 5.

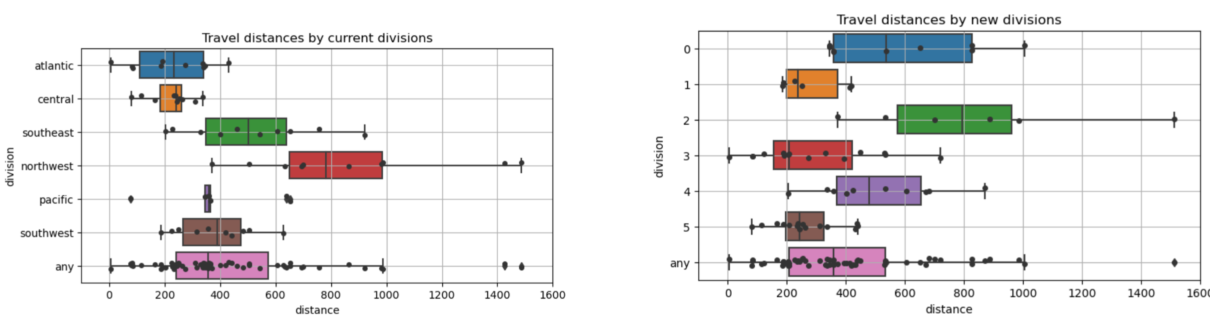
Thus, if the NBA wanted to have all the teams play in a time span where high attendance is preferred (such as opening week), they would be able to use this graph network to find the configuration of games that should be played that week to



(a) Conference clustering.

(b) Division clustering.

Fig. 2. Illustration of clustering results to minimize travel time.



(a) Old division assignments.

(b) New division assignments.

Fig. 3. Changes in travel times between old division assignments and new.

ensure every team plays at least once. In particular, they can also use this to pick the games that will have the maximum attendance to be played on the most important day (such as opening day). For this graph, the three games that should be played on opening day would be the Los Angeles Lakers in Minneapolis, MN (the Timberwolves) with a predicted attendance of 17616 people, the Cleveland Cavaliers playing in Denver, CO (the Nuggets) with a predicted attendance of 17733 people, and the Chicago Bulls playing in Dallas, TX (the Mavericks) with a predicted attendance of 17849 people.

II. Conclusion

We have proposed new classifications of teams in the NBA conferences and divisions which will minimize travel time through K-Means and Spectral Clustering. We have also provided recommendations on which teams should play in which cities to maximize attendance using the Edmonds-Karp algorithm. This will help the NBA as it faces problems regarding team exhaustion, sustainability, viewership, and attendance in the coming years. It will also help the NBA total revenue as the amount of money a family of four would need to spend to attend an NBA game, or the Fan Cost Index (FCI), for the 2019-2020 NBA season was $430.25 [7].

III. Future Work

The conference and division recommendations can be used for potential expansion teams that the NBA is rumoured to be considering for the 2024-2025 NBA season. Our work can be used for planning purposes for the expansion teams to decide which conferences and divisions they should be placed into. This can also be applied to the G-League, which is the official minor league for the NBA where many up and coming players start in. Currently, it's conprised of 15 teams. It can also be applied to the Women's National Basketball League (WNBA) and is comprised of 12 teams. Both the G-League and WNBA would be interesting expansions of the clustering as the players in those leagues, particularly the WNBA, have been vocal regarding the exhaustion and struggles that they face dealing with the travel required from them for the games as they do not have as many resources like team planes to get them to games [8].

Other future work that can be done are to use it to build out a full-fledged schedule for certain game weeks which will maximize attendance. They can also use the graph network set-up we have created towards the other leagues such as the G-League and WNBA.

References

[1] https://www.basketball-reference.com/leagues/NBA_2019_games-april.html

[2] https://en.wikipedia.org/wiki/Spectral_clustering

[3] https://en.wikipedia.org/wiki/K-means_clustering

[4] https://en.wikipedia.org/wiki/Ford%E2%80%93Fulkerson_algorithm

[5] https://www.cs.cornell.edu/courses/cs4820/2012sp/handouts/edmondskarp.pdf

[6] https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/tr-2000-65.pdf

[7] https://www.hoopsaddict.com/how-nba-makes-money/

[8] https://nba.nbcsports.com/2022/12/12/lakers-reportedly-among-several-teams-interested-in-cam-reddish-trade/

IV. Appendix I: NBA Team Dataset

team_name	city_name	conference	division	arena_lat	arena_long
Boston Celtics	Boston, MA	eastern	atlantic	-71.062228	42.366303
Brooklyn Nets	New York City, NY	eastern	atlantic	-73.974689	40.682650
New York Knicks	New York City, NY	eastern	atlantic	-73.993611	40.750556
Philadelphia 76ers	Philadelphia, PA	eastern	atlantic	-75.171944	39.901111
Toronto Raptors	Toronto, Ontario	eastern	atlantic	-79.379167	43.643333
Chicago Bulls	Chicago, IL	eastern	central	-87.674167	41.880556
Cleveland Cavaliers	Cleveland, OH	eastern	central	-81.688056	41.496389
Detroit Pistons	Detroit, MI	eastern	central	-83.245556	42.696944
Indiana Pacers	Indianapolis, IN	eastern	central	-86.155556	39.763889
Milwaukee Bucks	Milwaukee, WI	eastern	central	-87.916944	43.043611
Atlanta Hawks	Atlanta, GA	eastern	southeast	-84.396389	33.757222
Charlotte Hornets	Charlotte, NC	eastern	southeast	-80.839167	35.225000
Miami Heat	Miami, FL	eastern	southeast	-80.188056	25.781389
Orlando Magic	Orlando, FL	eastern	southeast	-81.383611	28.539167
Washington Wizards	Washington, DC	eastern	southeast	-77.020833	38.898056
Denver Nuggets	Denver, CO	western	northwest	-105.007500	39.748611
Minnesota Timberwolves	Minneapolis, MN	western	northwest	-93.276111	44.979444
Oklahoma City Thunder	Oklahoma City, OK	western	northwest	-97.515000	35.463333
Portland Trail Blazers	Portland, OR	western	northwest	-122.666667	45.531667
Utah Jazz	Salt Lake City, UT	western	northwest	-111.901111	40.768333
Golden State Warriors	San Francisco, CA	western	pacific	-122.387500	37.768056
Los Angeles Clippers	Los Angeles, CA	western	pacific	-118.267222	34.043056
Los Angeles Lakers	Los Angeles, CA	western	pacific	-118.267222	34.043056
Phoenix Suns	Phoenix, AZ	western	pacific	-112.071389	33.445833
Sacramento Kings	Sacramento, CA	western	pacific	-121.518056	38.649167
Dallas Mavericks	Dallas, TX	western	southwest	-96.810278	32.790556
Houston Rockets	Houston, TX	western	southwest	-95.362222	29.750833
Memphis Grizzlies	Memphis, TN	western	southwest	-90.050556	35.138333
New Orleans Pelicans	New Orleans, LA	western	southwest	-90.081944	29.948889
San Antonio Spurs	San Antonio, TX	western	southwest	-98.437500	29.426944