Basketball Elimination problem

Alon Lapid, Spring 2020

Elimination problem: How can you tell in the middle of a season which team could not finish first ?

Simple case: Is team 4 eliminated ?

				Games to play			
Team	**Wins**	**Losses**	**Left**	**team1**	**team2**	**team3**	**team4**
team1	83	71	8	-	1	6	1
team2	80	79	3	1	-	0	2
team3	78	78	6	6	0	-	0
team4	77	82	3	1	2	0	-

Answer: **Yes**. Even if team 4 win all its remaining game it will have only 80 points, far behind team 1

Elimination problem: Cont.

Not so simple case: Is team 5 eliminated ?

Games to play

Team	Wins	Losses	Left	team1	team2	team3	team4	team5
team1	75	59	21	-	3	8	7	3
team2	71	63	14	3	-	2	7	4
team3	69	66	10	8	2	-	-	-
team4	63	72	14	7	7	-	-	-
team5	49	86	7	3	4	-	-	-

Answer: **Yes(!!)** Even if it wins all its 27 games have 76 points, other teams will be awarded points as well and there is no scenario in which it will finish first

Elimination problem: Cont.

Can we try to solve the previous problem using brute force namely trying all possible outcomes of the remaining games ?

We can but the run time will be exponential with the number of games. In the previous example we will have to evaluate **2^((27+27+27+28+28)/2)** = **4.1740217e+20** possibilities.... not so practical.

The network flow solution

What is a network flow ?

It is a directed, weighted graph, with 2 special nodes the sync "s" and the sync "t". The weights on the edges are called capacity. We would like to flow the maximum input flow into s so that :

- The total amount of incoming flow to each node is the total among flowing out from the node
- The total flow on each edge is less or equal to the capacity of the edge.
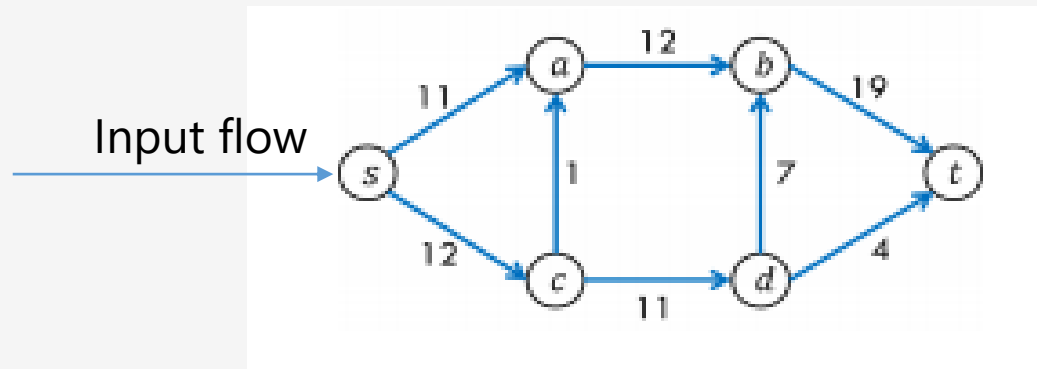
Example : Using Maximum flow for testing if team 5 is eliminated

| Team	Wins	Losses	Left	team1	team2	team3	team4	team5
team1	75	59	21	-	3	8	7	3
team2	71	63	14	3	-	2	7	4
team3	69	66	10	8	2	-	-	-
team4	63	72	14	7	7	-	-	-
team5	49	86	7	3	4	-	-	-
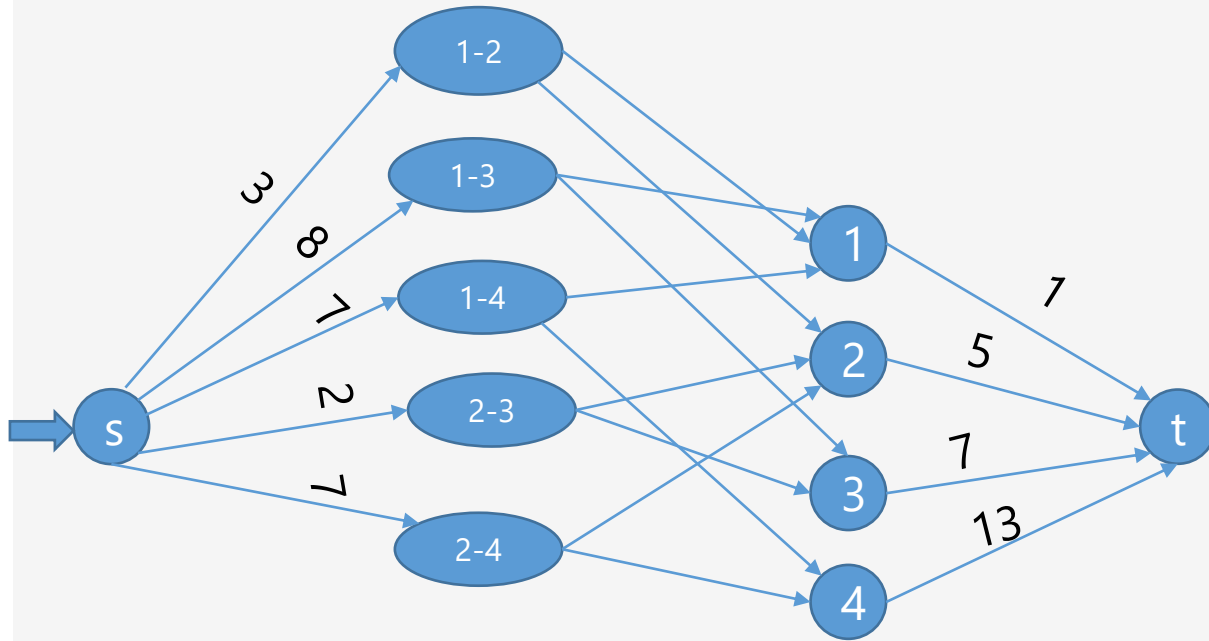
- The games of team 5 are removed – we assume it wins all of them

- From the source, have outgoing edges with capacity as the number of games between any pair of the teams

- From game nodes have outgoing edges with infinite capacity to the involved teams

- From each team node have an outgoing edge with the maximum number points the team can win without eliminating team 5

Can we feed into S a flow of 27 (the number of remaining games) ? If YES then team 5 is NOT eliminated and if NO then it is eliminated! In other words, team 5 is eliminated if the max flow of the network is less than 27

Using Maximum flow for elimination – why does it work ?

If we can feed into S all the remaining games (27 in our case) then we have found a scenario in which all game were played and team 5 finished first(not eliminated) otherwise it is eliminated !
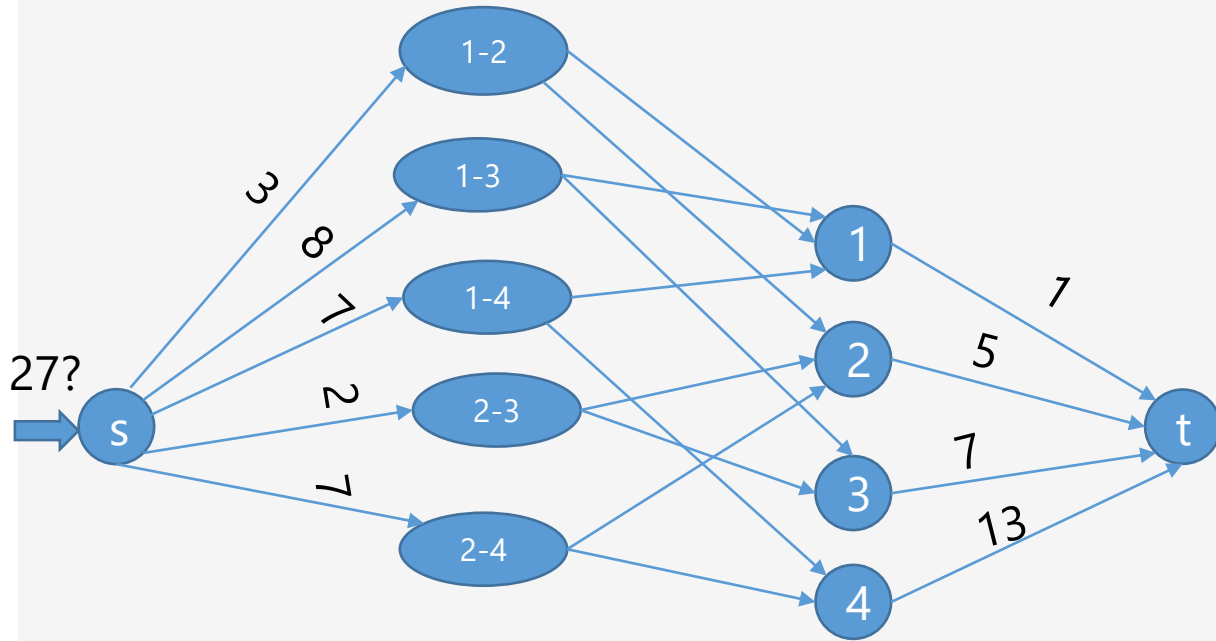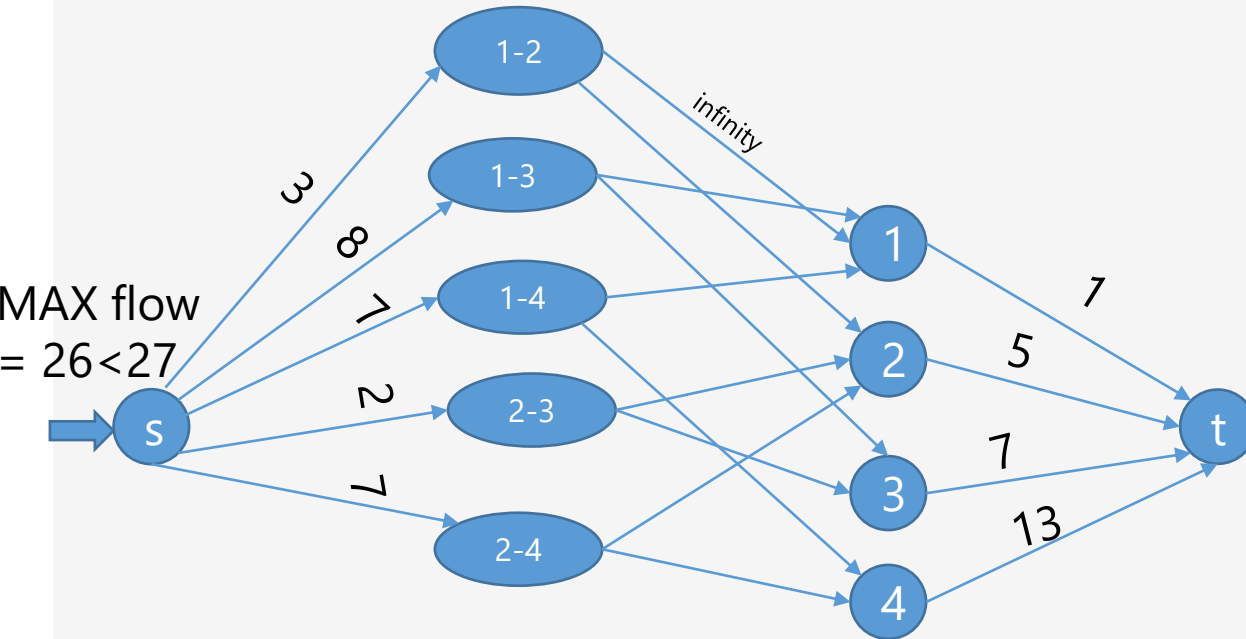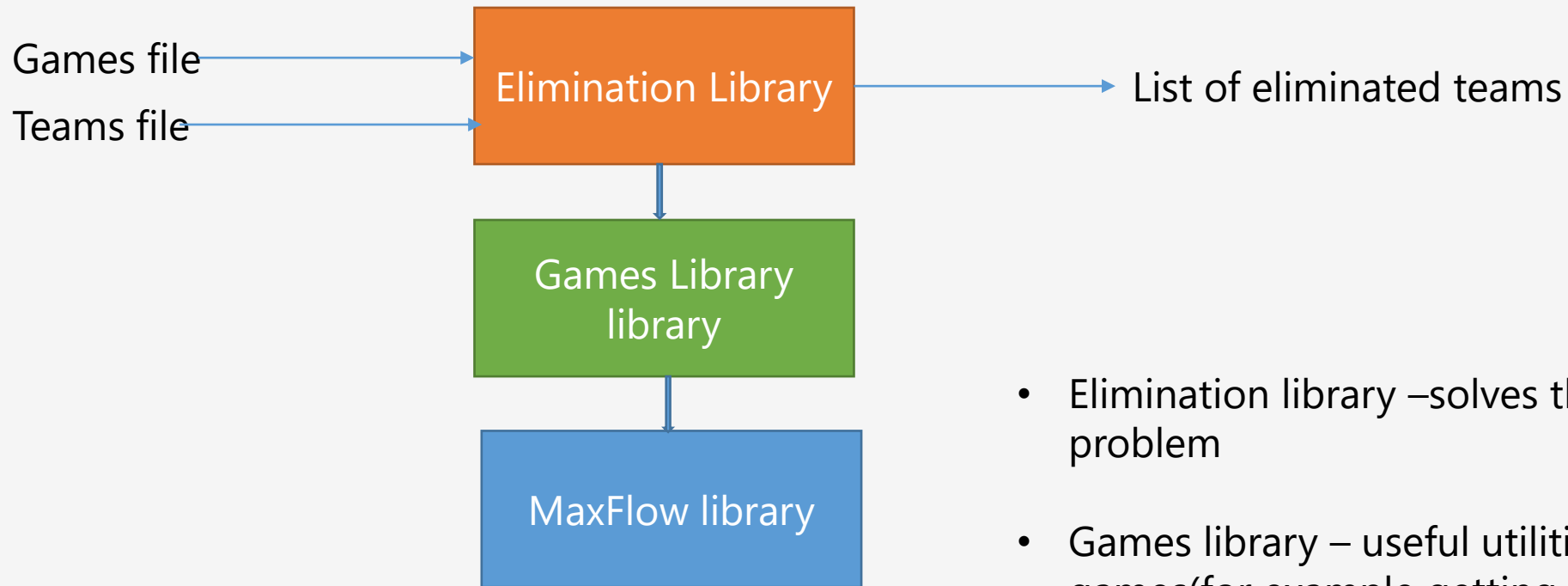
So is team 5 eliminated ?

We will have to find the maximum flow of the network
The standard technique is to use Edmund Karp algorithm
 Running it yields maximum flow of 26 < 27 and **hence team 5 is eliminated**

Implementation in Haskell

Games file

Teams file

Elimination Library

List of eliminated teams

Games Library library

MaxFlow library

- Elimination library –solves the elimination problem

- Games library – useful utilities to manipulate games(for example getting the number of points of the leading team)

- MaxFlow library –solves the max flow problem using Edmund Karp algorithm

Implementation in Haskell – input files

Teams file

Team	Conference
Toronto Raptors	east
Philadelphia 76ers	east
Boston Celtics	east
Brooklyn Nets	east
New York Knicks	east
Milwaukee Bucks	east
Indiana Pacers	east
Detroit Pistons	east
Chicago Bulls	east
Cleveland Cavaliers	east
Orlando Magic	east
Charlotte Hornets	east
Miami Heat	east
Washington Wizards	east
Atlanta Hawks	east
Denver Nuggets	west
Portland Trail Blazers	west
Utah Jazz	west
Oklahoma City Thunder	west
Minnesota Timberwolves	west
Golden State Warriors	west
LA Clippers	west
Sacramento Kings	west
Los Angeles Lakers	west
Phoenix Suns	west
Houston Rockets	west
San Antonio Spurs	west
Memphis Grizzlies	west
New Orleans Pelicans	west
Dallas Mavericks	west

Games file

Round	Date	Home team	Away team	Result
1	16/10/2018 20:00	Boston Celtics	Philadelphia 76ers	105 - 87
1	16/10/2018 22:30	Golden State Warriors	Oklahoma City Thunder	108 - 100
1	17/10/2018 19:00	Charlotte Hornets	Milwaukee Bucks	112 - 113
1	17/10/2018 19:00	Detroit Pistons	Brooklyn Nets	103 - 100
1	17/10/2018 19:00	Indiana Pacers	Memphis Grizzlies	111 - 83
1	17/10/2018 19:00	Orlando Magic	Miami Heat	104 – 101
1	17/10/2018 22:00	Sacramento Kings	Utah Jazz	117 - 123
1	17/10/2018 22:30	LA Clippers	Denver Nuggets	98 - 107
1	17/10/2018 22:30	Phoenix Suns	Dallas Mavericks	121 - 100
1	18/10/2018 20:00	Philadelphia 76ers	Chicago Bulls	127 - 108
1	18/10/2018 20:00	Washington Wizards	Miami Heat	112 - 113
1	18/10/2018 22:30	Portland Trail Blazers	Los Angeles Lakers	128 - 119
1	19/10/2018 19:00	Orlando Magic	Charlotte Hornets	-
1	19/10/2018 19:30	Brooklyn Nets	New York Knicks	-
1	19/10/2018 20:00	Toronto Raptors	Boston Celtics	-
...				

Usage example of elimination.hs

1) Download the games of season 2019-2020 locally :

downloadGamesfromWeb "https://fixturedownload.com/download/nba-2019-EasternStandardTime.csv" "test/nba-2019-EasternStandardTime.csv"

2) Get the list of eliminated teams for season 2019-2020 at the time the season was suspended :

eliminationMaxFlowFromFile "test/teamsnba.csv" "test/nba-2019-EasternStandardTime.csv"

And the eliminated teams are :

["Brooklyn Nets","New York Knicks","Detroit Pistons","Chicago Bulls","Cleveland Cavaliers","Orlando Magic","Charlotte Hornets","Washington Wizards","Atlanta Hawks","Portland Trail Blazers","Minnesota Timberwolves","Golden State Warriors","Sacramento Kings","Phoenix Suns","San Antonio Spurs","New Orleans Pelicans"]

Challenges

1) Some libraries did not work on Windows so I had to switch to ubuntu

2) Testing at scale is challenging as brutes force solution is impractical. I used small examples + some knows example from the internet

Possible improvement

One idea to solve the testing issues is to convert the maxflow problem into a linear constraints problem and use simplex method to solve it. That way I could do more testing in scale.

Questions ?