Major League Soccer Scheduling Problem

Canto Zhu, Muchen Zhou, Qiuyi Jia, Xiang Luo

Dec. 16 2016

1 Introduction

Major League Soccer is the highest level men’s professional soccer league in United States and Canada, consisting of 20 teams, with 17 in United States and 3 in Canada. The League is divided by two Conferences: Western and Eastern with 10 teams each. The regular season of the League runs from early March to late October each year. It is important to look at MLS mainly for three reasons. Firstly, it is a fast-growing league, with only 10 teams competed in 1996 when the League started. Secondly, the popularity of the game has increased dramatically, with the average per-game attendance in MLS increased by 40% in the past decade, which exceeds that of NHL and NBA. Lastly, the big TV broadcasting contract signed with ESPN, Fox Sports, and Univision has demonstrated the huge commercial value of the League.

Our model aims to make an efficient schedule that balances the travel distance, opponent strength, and game attractiveness of each team and game weeks. We will use integer programming to achieve these optimizing goals.

2 Rules

Each team in the League will play a total of 34 games in the regular season. 24 games are intra-conference and 10 are cross-conference games. The 10 cross-conference games are scheduled as 5 home and 5 away and switch home-away schedule year by year. Among the 24 intra-conference games, 18 of which are scheduled as double round robin with other 9 teams within the conference. The rest are 6 additional intra-conference games, 3 home and 3 away with different opponents.

Our model will focus on the Western Conference since there are larger distance differences between cities. Once we solved one conference, it will be relatively simple to repeat the same process on the other conference.

3 Data

We collected several kinds of data for our model. The data for distance are from distance24 API to acquire the direct distance between cities in North America. As for the calculation of fairness, we used team valuations released by Forbes in 2016 for team value, past two years average winning percentages (one year for newly-formed teams) for the winning rate, and attendance percentage instead of capacity rate for attendance rate as a way to measure fans enthusiasm for their teams. All results are normalized.
4 Assumptions

1. Assume that no team plays more than one game per week.

A regular MLS season is played between March and October, with a span of roughly thirty weeks, and typically on the weekends. Since each team has to play 34 games in total, we assume that there are 34 weeks so that no team plays more than once per week. In real schedules, there might be less weeks so that some teams have to play twice in a week. MLS solves this problem by adding mid-week games on Wednesdays and Thursdays, which we will address in later parts of this paper. In brief, our project solves the 34 week problem by integer programming, then adapts it to real life schedule by manually adding the additional mid-week games.

2. Assume that after each game, teams always return to their home city.

Following Assumption 1, we further assume that every away team returns to its home city after each game, since they would have enough time to rest between games. The greatest benefit it brings is the simplicity in calculating travelling distances for a game, which is just the distance between home cities of the two matching teams.

5 Optimization Goals

1. Minimizing total traveling distance for the 6 additional games

For the purpose of optimization, we are only trying to optimize the total traveling distance for the 6 additional intra-conference games for the nature of its relative randomness. Since the other 18 intra-conference games are fixed in distance, and the other 10 cross-conference games are fixed in distance in a two-year span with home and away schedules taking turns each year, there is no need to optimize their distances. Similar optimization problem is the traveling tournament problem (TTP) introduced in the seminal paper of Easton et al. (2001). It is a challenging combinatorial optimization problem in sports scheduling that derives the most significant aspects of traveling distances constraints in scheduling a timetable.

2. Home-away balance and Fairness

Besides distance optimization, we are also trying to achieve a home-away balance to ensure that no team plays too many consecutive home games or away games. To achieve fairness for each team, we want to balance the strength of each team’s opponents. That is, no team is playing with too many strong teams nor too many weak teams. The strength of each team is determined by its average winning rate in the past two seasons. Our fairness balance is calculated by a normalized weighted sum of team value, game attendance rate, and historical winning rate.

3. Attractiveness

As for the attractiveness of the games, we adhere to realistic situation by considering special weeks and holidays, such as July the Fourth, and conflicts with other popular sports for scheduling more attractive games. Besides, we take individual team’s opponents into consideration for attractiveness. For example, when LA plays against San Jose, it might be a lot more attractive since it’s a famous rivalry called California Clasico. We are also considering the weather effect in which for the first two weeks’ games, we try to have the season open with some of the more warm-climate teams hosting, for instance, eastern teams will play more aways at western teams’ homes.
6 Formulation

This section presents the formulation to the distance-minimizing optimization problem.

6.1 Definitions

Define $T = 10$ as there are 10 teams in both the Western Conference and the Eastern Conference.

Define $K = 34$ as there are 34 games (weeks) to be scheduled.

There are three categories of variables: distance variables, attractiveness variables, and decision variables.

6.1.1 Distance variables

There is a distance associated to each pair of teams, corresponding to the distance between their home cities. Given a number $T$ of teams, distances $d_{ij}$ between the home cities of teams $i$ and $j$, for every $i,j = 1,\ldots,T$ are the flight distance between the two cities, with $d_{ij} = 0$ if $i = j$. The flight distances are collected from online resources.

6.1.2 Fairness and Attractiveness variables

There is a fairness value associated to each team, calculated by the weighted sum of winning rate, attendance rate, and commercial value of the team. It represents the strength of a team, but the product of such value of the two teams is also a good indicator of attractiveness of a game. Although other factors may also influence game attractiveness, this is easier to be quantified and compared.

Given a number $T$ of teams, fairness $a_i$ of team $i$, for every $i = 1,\ldots,T$ are $w_1 \times \alpha_i + w_2 \times \beta_i + w_3 \times \gamma_i$, where $w_1$ is the weight for winning rate, $w_2$ is the weight for attendance rate, and $w_3$ is the weight for commercial value, and $\alpha_i$ is the normalized winning rate of team $i$, $\beta_i$ is the normalized attendance rate of team $i$, and $\gamma_i$ is the normalized commercial value of team $i$.

Currently, we set $w_1 = \frac{1}{3}$, $w_2 = \frac{1}{3}$, and $w_3 = \frac{1}{3}$. These values are based on estimations of how much each category contributes to strength of a team.

In addition, there is a fairness/attractiveness value associated to each game, calculated by the product of the fairness value of the opposing teams. Given a number $T$ of teams, fairness $A_{ij}$ of the game between team $i$ and $j$, for every $i = 1,\ldots,T$ are $a_i \times a_j$. We ensure the fairness value of each game is within certain range.

6.1.3 Decision variables

There are three sets of decision variables for three distinctive components of the set of games to be scheduled:

(1) $x_{ijk}$: the 6 additional games within conference

(2) $y_{ijk}$: the 10 games across conference

(3) $z_{ijk}$: the 18 regular games within conference

and $x_{ijk}, y_{ijk}, z_{ijk} \in \{0,1\}$, where 1 indicates if team $i$ plays away with team $j$ at week $k$ and 0 otherwise.

Note $i,j \in \{T\}$ for $x,z$ while $i,j \in \{2T\}$ for $y$ since $x$ and $z$ concern with only games within conference games while $y$ concerns also with games across conference games, and $k \in \{K\}$.
6.1.4 Summary of all variables

In summary, we define the following variables:

- \( d_{ij} := \) distance between the home of team \( i \) and team \( j \)
- \( \alpha_i := \) normalized winning rate of team \( i \)
- \( \beta_i := \) normalized attendance rate of team \( i \)
- \( \gamma_i := \) normalized commercial value of team \( i \)
- \( a_i := \) the weighted coefficient of winning rate, attendance rate, and commercial value of team \( i \) where \( w_j := \) the weight for winning rate, attendance rate, and commercial value in calculation of fairness \( a_i \) is calculated by \( a_i = w_1 \times \alpha_i + w_2 \times \beta_i + w_3 \times \gamma_i \)

- For the 6 additional games, \( x_{ijk} = \begin{cases} 1, & \text{if team } i \text{ plays away with team } j \text{ at week } k \\ 0, & \text{otherwise} \end{cases} \)

- For the 10 games across conference, \( y_{ijk} = \begin{cases} 1, & \text{if team } i \text{ plays away with team } j \text{ at week } k \\ 0, & \text{otherwise} \end{cases} \)

- For the 18 regular games within conference, \( z_{ijk} = \begin{cases} 1, & \text{if team } i \text{ plays away with team } j \text{ at week } k \\ 0, & \text{otherwise} \end{cases} \)

where \( i, j \) are indices of teams, \( i, j \in [T] \) for \( x, z \) (within conference) and \( i, j \in [2T] \) for \( y \) (across conference) , and \( k \) is index of weeks, \( k \in [K] \).

6.2 Integer Programming Problem

The integer programming problem can be formulated according to MLS rules and variable definitions as follows:

- Objective function

The objective is to minimize the total distance traveled by all teams:

\[
\min \sum_k \sum_i \sum_j d_{ij} x_{ijk} \quad i, j \in [T], k \in [K]
\]

- Constraints

  - Constraints concerned with all \( x, y, z \):

    (1) Distance from and to the same city is 0:

    \[ d_{iik} = 0 \quad \forall i, k \]

    (2) Each team does not play with itself:

    \[ x_{iik} = 0, y_{iik} = 0, z_{iik} = 0 \quad \forall i, k \]

    (3) Each team plays at most once per week:

    \[
    \sum_j (x_{ijk} + x_{jik} + y_{ijk} + y_{jik} + z_{ijk} + z_{jik}) \leq 1 \quad \forall i, k
    \]
(4) Ensure attractiveness for each week’s games is above some threshold $P$:

$$\sum_i \sum_j (a_i \times a_j)(x_{ijk} + y_{ijk} + z_{ijk}) \geq P \quad \forall k$$

- Constraints concerned with only $x$:

(5) Each team plays 3 home, 3 away additional games within conference:

$$\sum_i \sum_k x_{ijk} = 3 \quad \forall j, \quad \sum_j \sum_k x_{ijk} = 3 \quad \forall i$$

(6) Any two teams play at most once additionally:

$$\sum_j (x_{ijk} + x_{jik}) \leq 1 \quad \forall i, k$$

(7) Ensure fairness is within range $[L, U]$:

$$L \leq \sum_k \sum_i \alpha_i (x_{ijk} + x_{jik}) \leq U \quad \forall j$$

- Constraints concerned with only $y$:

(8) Each team plays 5 home, 5 away games across conference:

$$\sum_j \sum_k y_{ijk} = 5 \quad \forall i, \quad \sum_i \sum_k y_{ijk} = 5 \quad \forall j$$

(9) Any two teams across conference play one game:

$$\sum_j (y_{ijk} + y_{jik}) = 1 \quad \forall i, k$$

- Constraints concerned with only $z$:

(10) Each team plays 9 home, 9 away regular games within conference:

$$\sum_j \sum_k z_{ijk} = 9 \quad \forall i, \quad \sum_j \sum_k z_{ijk} = 9 \quad \forall j$$

(11) Any two teams play 1 home, 1 away games:

$$\sum_k z_{ijk} = 1 \quad \text{and} \quad \sum_k z_{jik} = 1 \quad \forall i, j$$

where $x_{ijk}, y_{ijk}, z_{ijk} \in \{0, 1\}$.

7 Implementation

After exploring various tools, we use Matlab as the integer programming solver, since it outperforms many other tools (mainly in terms of runtime) with our large problem scale. Matlab also has well-documented integer programming solver libraries.

This section discusses some core implementation details.
7.1 Generate constants, distance, fairness, and winning rate

Set the values of $T$, $K$, distance, fairness, and winning rate.

```matlab
1 M = csvread('Data - Basic Info.csv'); % distance
2 A = csvread('Data - Fairness.csv'); % fairness
3 w = csvread('Data - WinningRate.csv'); % winning rate
4
5 T = 10; % number of teams
6 K = 34; % number of games/weeks
```

The distance data, fairness data, and winning rate data are collected from online resources and calculated as explained in variable definitions.

Note $M(i,j)$ in code corresponds to $d_{ij}$ in this paper, $A(i)$ in code corresponds to $a_i$ in this paper, and $w(i)$ corresponds to $\alpha_i$ in this paper. The naming is slightly different in code for better readability.

7.2 Generate objective and constraint matrices and vectors

The objective function vector `obj` in `intlincon` consists of the coefficients of the decision variables $x_{ijk}$. Even though $y_{ijk}$ and $z_{ijk}$ are not in the objective function, they are decision variables to be determined. So we still include them in `obj` but with their coefficients set as zeros. Therefore, there are naturally $T \times T \times K + 2T \times 2T \times K + T \times T \times K$ coefficients in `obj`.

```matlab
1 obj1 = zeros(T,T,K); % 3 dimensional vector for x
2 obj2 = zeros(2*T,2*T,K); % 3 dimensional vector for y
3 obj3 = zeros(T,T,K); % 3 dimensional vector for z
```

Generate the coefficients of $x_{ijk}$ in the objective function

```matlab
1 % populating objective function
2 for i = 1:T
3 for j = 1:T
4 for k = 1:K
5 obj1(i,j,k) = M(i,j);
6 end
7 end
8 end
```

Combine the entries into one `obj` vector

```matlab
1 obj = [obj1(:);obj2(:);obj3(:)]; % convert 3 dimensional obj fun to vector
```

Now create the constraint matrices.

The width of each linear constraint matrix is the length of the `obj` vector.

```matlab
1 matwid = length(obj);
```

7.2.1 Linear inequalities constraints

Linear inequalities include constraints (3), (4), (6), and (7). We illustrate constraint (6) as an example. The rest of them are similar. Constraints (6) expands to $T \times K$ separate constraints, since $i$ ranges from 1 to $T$ and $k$ from 1 to $K$. The constraint matrices are quite sparse, so save memory by using sparse matrices.
... matheight = matheight + T*K % add rows to Aineq & bineq for constraint(6)
... % similarly, add rows for other constraints

Aineq = spalloc(matheight, matwid, T*T*K*10); % allocate sparse matrix
bineq = zeros(matheight, 1); % allocate bineq as full

% Zero matrices of convenient sizes
clearer1 = zeros(size(obj1));
clearer12 = clearer1(:);
clearer2 = zeros(size(obj2));
clearer22 = clearer2(:);
clearer3 = zeros(size(obj3));
clearer32 = clearer3(:);
cnt = 1;
% any two teams play at most once additionally
for i = 1:T
  for k = 1:K
    xtemp = clearer1;
    % sum xijk over all j as second index
    xtemp(i, :, k) = xtemp(i, :, k) + 1;
    % sum xijk over all j as first index
    xtemp(:, i, k) = xtemp(:, i, k) + 1;
    % convert to sparse matrix
    xtemp = sparse([xtemp(:); clearer22; clearer32]);
    % fill in the row of Aineq
    Aineq(cnt, :) = xtemp';
    % fill in the row of bineq
    bineq(cnt) = 1;
    cnt = cnt + 1;
  end
end

% similarly, add other linear inequalities constraints

7.2.2 Linear equalities constraints

Linear equalities include constraints (1), (2), (5), and (8) to (11). We illustrate constraint (5) as an example. The rest of them are similar.

Constraints (5) expands to $T + T$ separate constraints, since there are two separate parts for home and away, $i$ ranges from 1 to $T$.

... matheight = matheight + T + T; % add rows to Aineq & bineq for constraint(5)
... % similarly, add rows for other constraints

Aeq = spalloc(matheight, matwid, matzeros);
beq = zeros(matheight, 1);
cnt = 1;
% three home games
for j = 1:T
  xtemp = clearer1;
  % sum xijk over all j and k
  xtemp(:, j, :) = 1;
  % convert to sparse matrix
  xtemp = sparse([xtemp(:); clearer22; clearer32]);
  % fill in the row of Aeq
  Aeq(cnt, :) = xtemp';
18  % same for beq
19  beq(cnt) = 3;
20  cnt = cnt+1;
21  end
22
23  % three away games
24  for i = 1:T
25     xtemp = clearer1;
26     % sum xijk over all i and k
27     xtemp(i, :, :) = 1;
28     % convert to sparse matrix
29     xtemp = sparse([xtemp(:); clearer22; clearer32]);
30     % fill in the row of Aeq
31     Aeq(cnt, :) = xtemp';
32     % same for beq
33     beq(cnt) = 3;
34     cnt = cnt+1;
35  end
36  ...
7.3 Bound constraints and integer variables

The integer variables are all those from obj.
1  intcon = 1:length(obj);

The upper bounds and lower bounds are for the above set of integer variables also. Upper bounds are all ones, and lower bounds are all zeros.
1  lb = zeros(length(obj), 1);
2  ub = zeros(length(obj), 1);
3  ub(:) = 1;

7.4 Solve the problem

After generating all the solver inputs, call the solver to find the solution.
1  [sol, fval, exitflag, output] =
2      intlinprog(obj, intcon, Aineq, bineq, Aeq, beq, lb, ub);

7.5 Examine the solution

The solution is feasible, with approximately 3-minute runtime.

...  
LP: Optimal objective value is 40032.000000.

Cut Generation: Applied 8 clique cuts,
and 1 zero-half cut.
Lower bound is 40032.000000.

Branch and Bound:

<table>
<thead>
<tr>
<th>nodes explored</th>
<th>total time (s)</th>
<th>num int solution</th>
<th>integer fval</th>
<th>relative gap (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>1000000</td>
<td>1</td>
<td>1</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Optimal solution found.

Intlinprog stopped because the objective value is within a gap tolerance of the optimal value, options.AbsoluteGapTolerance = 0 (the default value). The intcon variables are integer within tolerance, options.IntegerTolerance = 1e-05 (the default value).

To obtain the schedule:

1. % Reshape the result matrix by x, y, z’s variables vector sizes
2. \( x_s = \text{sol}(1:length(\text{clearer12})); \)
3. \( X = \text{reshape}(x_s, T, T, K); \)
4. ... % translate 3 dimensional vector X to print as
5. ... % ‘team i plays with team j on week k’
6. ... % do the same for 3 dimensional vectors Y and Z

The output is the concrete schedule.

8 Results and Comparison

8.1 Distance

We first discuss the results of travelled distance. We primarily focus on the optimization of six additional intra-conference games. The optimized result with respect to team is shown in the table below. Please be aware that all of the distance is in unit of kilometer.

<table>
<thead>
<tr>
<th>Team</th>
<th>MLS Scheduled Distance</th>
<th>Our Scheduled Distance</th>
<th>Percentage Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>FC Dallas</td>
<td>4311</td>
<td>4676</td>
<td>+8.47%</td>
</tr>
<tr>
<td>Colorado Rapids</td>
<td>3726</td>
<td>4263</td>
<td>+14.41%</td>
</tr>
<tr>
<td>LA Galaxy</td>
<td>5955</td>
<td>5560</td>
<td>-6.63%</td>
</tr>
<tr>
<td>Seattle Sounders FC</td>
<td>5985</td>
<td>2903</td>
<td>-51.50%</td>
</tr>
<tr>
<td>Sporting Kansas City</td>
<td>4014</td>
<td>4805</td>
<td>+18.71%</td>
</tr>
<tr>
<td>Real Salt Lake</td>
<td>3638</td>
<td>2749</td>
<td>-24.44%</td>
</tr>
<tr>
<td>Portland Timbers</td>
<td>4629</td>
<td>2914</td>
<td>-37.05%</td>
</tr>
<tr>
<td>Vancouver Whitecaps FC</td>
<td>4435</td>
<td>4435</td>
<td>0.00%</td>
</tr>
<tr>
<td>San Jose Earthquakes</td>
<td>2874</td>
<td>4388</td>
<td>+52.68%</td>
</tr>
<tr>
<td>Houston Dynamo</td>
<td>7715</td>
<td>3339</td>
<td>-56.72%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>47282</td>
<td>40032</td>
<td>-15.33%</td>
</tr>
</tbody>
</table>

We notice that in the west conference, 5 teams travel less, 1 team does not change and 4 teams travel more. In average, we reduced 15% of travel distance of intra-conference games. We also want to check the travel distance per team. By checking the original schedule, we can see that the range of travel distance per team is (2874,7715) and our scheduled travel distance per team is (2749,5560), which is significantly reduced both in range and maximum of travel distance per team. This is important because we want to make sure every team travels less, which is our primary optimization goal, and equally so that it’s fair to each team.

8.2 Fairness

In our proposed optimization goal, we use historical winning rate and a combination of historical winning rate, normalized team value and attendance rate to determine the fairness quantitatively. First, we will take
a look at winning rate section. Figure 1 shows each team’s opponents’ average winning rate. We calculate the winning rate of the opposite teams by

\[ W_{o[i]} = \frac{1}{6} \sum_{j=1}^{6} W_j \]

Where \( W_{o[i]} \) denotes team i’s opponents average winning rate and \( W_j \) denotes team j’s historical winning rate.

Notice that we only consider the six intra-conference games because other games are fixed every year or in two-years cycle. We can notice that our schedule performs in a similar manner as the MLS schedule with a little bit larger variance. This is ideal because we do wish stronger team to have a schedule that is a little bit more challenging so it can be interesting to watch within an acceptable range.

The next value we want to examine is the weighted combination of historical winning rate, normalized team value, and attendance rate. This value represents the strength of a team, but the product of such value of the two teams is also a good indicator of attractiveness of a game. Here we use both the average and maximum values through weeks to check the fairness and attractiveness of our schedule.

Firstly, the average game attractiveness per week is calculated by

\[ T_t = \frac{1}{N_t} \sum_{i,j=1}^{N_t} a_i \times a_j \]

where \( T_t \) denotes the average game attractiveness at week t, \( N_t \) denotes the number of games at week t, \( a_i, a_j \) denotes the fairness/attractiveness of team i and j.

In terms of average game attractiveness, our schedule performs similarly with the original MLS schedule. The comparison is presented in Figure 2. The reason for using average game attractiveness per week to examine the overall attractiveness of that week so we can optimize our schedule regarding other sports league. In the weeks where there are other big and important sporting events, we can arrange game weeks where the average attractiveness is smaller, otherwise we can put game weeks with higher average attractiveness. That being said, we find out that our schedule is slightly better on both tails, denoting that we can attract more people during other league’s off season and lose less viewership during other league’s important dates.

The second value is maximum game attractiveness per week, which can help us to decide which game to put
Figure 2: Average Game Attractiveness Per Week

Figure 3: Average Game Attractiveness Per Week
on prime time. The comparison is shown in Figure 3. Again, our schedule performs very similar compared with MLS’s original schedule.

9 Discussion

9.1 Robustness

Since the number of teams and weeks are pretty much fixed, the only factors that might influence robustness are the weights in fairness values. By testing different values, we found out that the weights have minimal influence on the final schedule. This influence shows that commercial value, winning rate, and attendance rate are equally important in determining the strength of a team. Therefore we decided to equally distribute their weights in representing their share of influence in the model.

As for other changing coefficients, which are the upper and lower bounds L, U, and P for fairness and attractiveness, we tested by evaluating final results and finding the tightest bound that gives a feasible solution.

9.2 Further Discussion

1. Bye weeks

MLS sometimes have bye weeks, in which a team doesn’t play any game during the whole week. It ensures that players have enough time to rest and extends the length of a regular season for commercial reasons. Since most of the bye weeks are determined by various complex factors like national team games and other major sport events, it is very hard to summarize a formulation in this model. But in real life, it is necessary to study the reasons and ways to schedule bye weeks if given enough data and resources.

2. Complete schedule

After scheduling for the Western Conference, we can easily apply the same method to the Eastern Conference as well. Obviously there is no conflict between the 24 infra-conference games, but it is necessary to consider the 10 cross-conference games. Since crossing the states is a relatively long flight, it might be more efficient to schedule two away games within a week so that the away team doesn’t need to fly too much. But at the same time we still need to maintain the home-away balance.

10 Conclusion

As many factors may influence a sports scheduling problem, our model simplifies and quantifies major variables like traveling distance, fairness, and attractiveness to achieve an efficient, fair, and interesting game schedule. The result shows that our model performs equally good with and even better than the official schedule, specifically in reducing travelling distance, balancing team strength, and enhancing game attractiveness. We tried to consider as many non-quantifiable variables as possible by adapting the programming output to real life cases, but there are still many factors that could be taken into consideration. Fortunately, due to a relatively high robustness, it is repeatable with different conferences or seasons, thus can be tested and subject to changes in future studies.
### A Final Schedule

The final schedule is obtained by MATLAB output and manually alternating some games by non-quantifiable factors, like weather and special events. Home away balance and mid-week games are also considered and checked in this final schedule.

<table>
<thead>
<tr>
<th>Round</th>
<th>March 5</th>
<th>Dallas vs Kansas City</th>
<th>LA vs Portland</th>
<th>Seattle vs Colorado</th>
<th>Vancouver vs San Jose</th>
<th>Houston vs Salt Lake</th>
<th>Weather effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Round 1</td>
<td>March 6</td>
<td>Dallas vs Kansas City</td>
<td>LA vs Portland</td>
<td>Seattle vs Colorado</td>
<td>Vancouver vs San Jose</td>
<td>Houston vs Salt Lake</td>
<td>Weather effect</td>
</tr>
<tr>
<td>Round 2</td>
<td>March 12</td>
<td>Vancouver vs Kansas City</td>
<td>Portland vs San Jose</td>
<td>Columbia vs LA</td>
<td>Colorado vs Salt Lake</td>
<td>Houston vs Salt Lake</td>
<td>Georgia vs Utah</td>
</tr>
<tr>
<td>Round 3</td>
<td>March 19</td>
<td>San Jose vs Houston</td>
<td>Seattle vs LA</td>
<td>Portland vs D.C. United</td>
<td>Philadelphia vs Delaware</td>
<td>Colorado vs Kansas City</td>
<td>Salt Lake vs Vancouver</td>
</tr>
<tr>
<td>Round 4</td>
<td>March 23</td>
<td>San Jose vs Philadelphia</td>
<td>Vancouver vs Portland</td>
<td>26-March Salt Lake vs Seattle</td>
<td>Kansas City vs Utah</td>
<td>LA vs Philadelphia</td>
<td>Salt Lake vs Chicago Houston vs Colorado</td>
</tr>
<tr>
<td>Round 5</td>
<td>March 30</td>
<td>Chicago vs Dallas</td>
<td>Houston vs Vancouver</td>
<td>2-April Chicago vs LA</td>
<td>NYC vs Salt Lake</td>
<td>NY Red Bulls vs Colorado</td>
<td>Portland vs Kansas San Jose vs Seattle</td>
</tr>
<tr>
<td>Round 6</td>
<td>April 6</td>
<td>Houston vs Chicago</td>
<td>Seattle vs San Jose</td>
<td>9-April Vancouver vs Salt Lake</td>
<td>NYC vs Portland</td>
<td>Dallas vs Houston</td>
<td>Colorado vs LA</td>
</tr>
<tr>
<td>Round 7</td>
<td>April 16</td>
<td>Portland vs Colorado</td>
<td>LA vs Houston</td>
<td>Kansas City vs Salt Lake</td>
<td>Dallas vs Toronto</td>
<td>San Jose vs Columbus</td>
<td>D.C. United vs Vancouver Toronto vs Seattle</td>
</tr>
<tr>
<td>Round 8</td>
<td>April 23</td>
<td>San Jose vs Portland</td>
<td>Dallas vs D.C. United</td>
<td>Colorado vs New England</td>
<td>Houston vs NY Red Bulls</td>
<td>Montreal vs Salt Lake</td>
<td>Seattle vs Vancouver Kansas City vs LA</td>
</tr>
<tr>
<td>Round 9</td>
<td>April 29</td>
<td>Dallas vs San Jose</td>
<td>Columbus vs Colorado</td>
<td>New England vs Seattle</td>
<td>Kansas City vs Vancouver</td>
<td>LA vs Houston</td>
<td>Portland vs Salt Lake</td>
</tr>
<tr>
<td>Round 10</td>
<td>May 7</td>
<td>San Jose vs Saint Louis</td>
<td>Kansas vs Seattle</td>
<td>Houston vs Salt Lake</td>
<td>Colorado vs Montreal</td>
<td>NY Red Bulls vs Dallas</td>
<td>Portland vs LA</td>
</tr>
<tr>
<td>Round 11</td>
<td>May 11</td>
<td>Portland vs Colorado</td>
<td>Toronto vs Kansas</td>
<td>14-May Houston vs Seattle</td>
<td>NY Red Bulls vs Vancouver</td>
<td>San Jose vs Toronto</td>
<td>Seattle vs PhiladelphiaNYC vs LA Dallas vs NYC</td>
</tr>
</tbody>
</table>

### B References

