Conrad, Blake

Abstract

In this work the Traveling Salesman Problem (TSP) is solved by implementing feasibility cuts to eliminate subtours. The initial problem solves a relaxed formulation of the TSP introducing multiple subtours. After implementing cutting planes and feasibility cuts, the subtours are eliminated sequentially to reduce the problem to an optimal solution. Problem formulation and computational results discussed. All code files and data are embedded in this document.

imse 884 final project

TSP Subtour Elimination by Feasibility Cuts

**IMSE 884 Final Project**

**Blake Conrad**

Contents

[Methodology 3](#_Toc38185058)

[Background 3](#_Toc38185059)

[Data 3](#_Toc38185060)

[Problem Formulation 3](#_Toc38185061)

[Subtour Feasibility Cuts 4](#_Toc38185062)

[Computational Results 4](#_Toc38185063)

[Original Solution 4](#_Toc38185064)

[Feasibility Cuts 5](#_Toc38185065)

[Optimal Solution 8](#_Toc38185066)

[Conclusion 9](#_Toc38185067)

[References 10](#_Toc38185068)

# Methodology

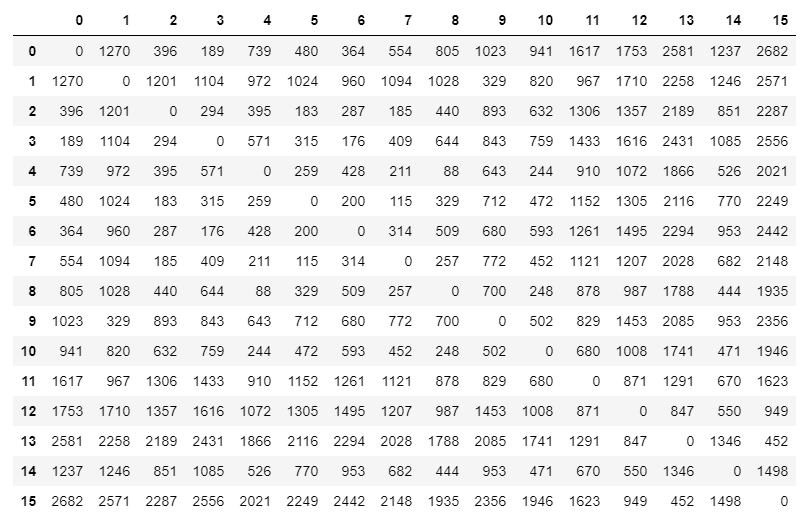
## Background

The Traveling Salesman Problem (TSP) is one of the most well studied combinatorial optimization problems. Many heuristics have been developed, but the optimal solution is still NP-Hard. In this paper we present the utility of cutting planes in the solution to the traveling salesman problem. Presented is the underlying data, the problem formulation, and the iterative feasibility cuts used to eliminate subtours in the TSP. Following a careful examination of the computational results and the optimal solution is analyzed.

## Data

The National Football League (NFL) has stadiums in almost every large city across the U.S. A sample of 16 of these cities was collected between stadiums. Figure 1 illustrates the distances between stadiums.

Figure –Distance Matrix Between NFL Stadiums



## Problem Formulation

The TSP has a variety of problem formulations, but the most common and simple of them is to minimize distance while constraining travel arrangements until feasible. Figure 2 illustrates the Integer Program (IP) formulation.

Figure 2 – Simplified TSP Formulation

The objective of this problem formulation is to minimize the total distance traveled. The goal is to select binary decision variables to decide if one should travel from city i to city j in the minimization. Constraints (1) and (2) indicate all cities must be traveled in and out of exactly once. Constraint (3) indicates traveling back and forth between pairs of cities is not allowed. Constrain (4) asserts that the decision variables must be binary.

## Subtour Feasibility Cuts

The idea of cutting planes is to construct an inequality that will cut off portions of the decision space to help find an integer solution. In the case of TSP, these cutting planes are in the form of feasibility cuts that cut off dozens of subtour combinations. The goal of this technique is as follows:

* Solve the original IP
* Identify subtours in the TSP
* Add feasibility cuts into the original formulation
* Resolve the IP

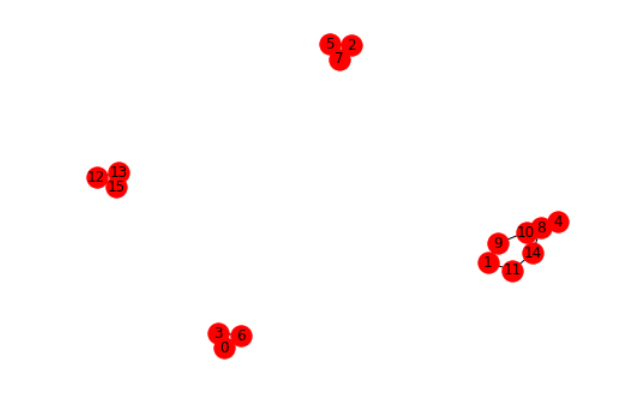
The above steps will be repeated until an optimal solution is found.

# Computational Results

## Original Solution

The original solution to the IP formulation in the *Methodology* section is reflected in Figure 3.

Figure 3 – Original TSP Solution From Methodology



## Feasibility Cuts

The solution to the original IP is clearly suboptimal because it has subtours. By implementing cutting planes in the form of feasibility cuts, we can trim the decision space down until it becomes feasible. Figures 4, 5, and 6 shows the constraints added, the output from the IP after implementing those constraints, and the new respective decision matrix. Specifically, Figure 4 adds subtour elimination constraints for the subtour of (0,3,6) and all possible combinations of paths between them. Since the tour is of size 3, the cut will restrict travel to at most 2 of the 3 of any possible paths between them. This is repeated for nodes (12,13, 15) and (1, 11, 14, 10, 9, 8, 4). As the size of the subtour grows, the number of combinations of the nodes within it grow as well, making constraints very large as the problem gets bigger. Figure 5 illustrates the output decision matrix. Figure 6 shows the graphical version of Figure 5 so it is obvious where subtours have been eliminated.

Figure 4 – 1st Feasibility Cuts Added

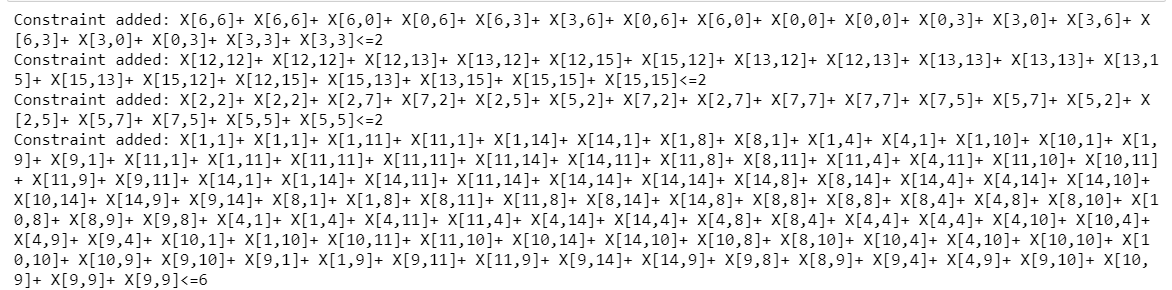


Figure 5 – 1st Output Decision Matrix From Feasibility Cuts

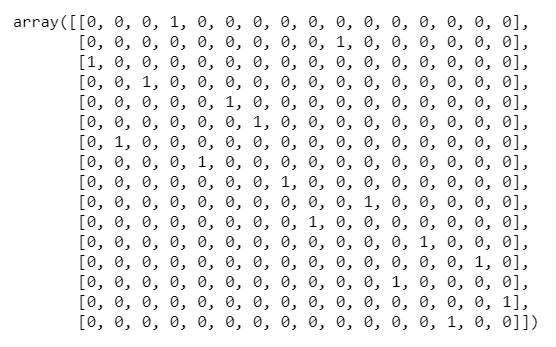


Figure 6 – 1st Output From Feasibility Cuts

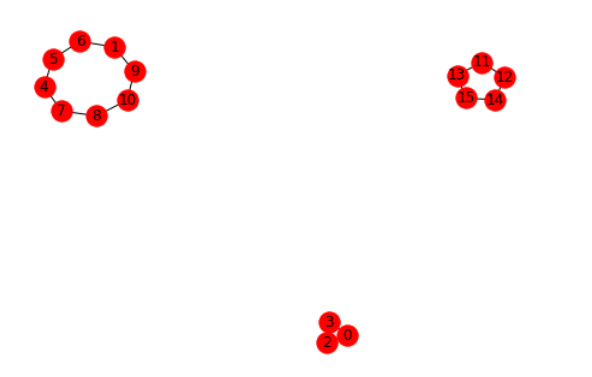


Figure 6 clearly shows the elimination of 1 subtour and extending of 2 from Figure 3. This shows promise in the technique of feasibility cuts. Figure 7, 8, and 9 illustrate an identical procedure to Figure 4, 5, and 6 except that the new cuts are cumulatively added to the previous two, that is Figure 3 and 6.

Figure 7 – 2nd Feasibility Cuts Added

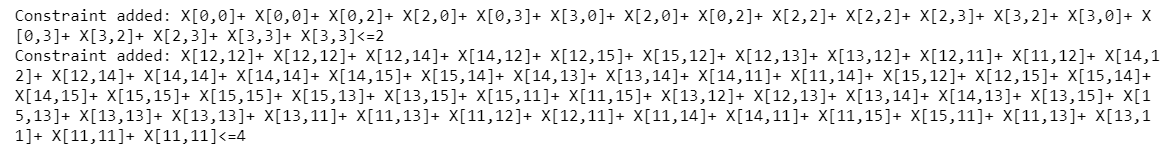


Figure 8 – 2nd Output Decision Matrix From Feasibility Cuts

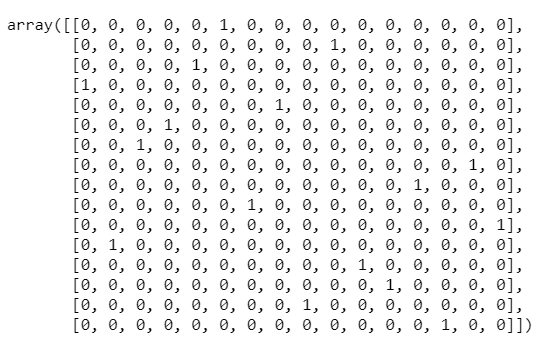
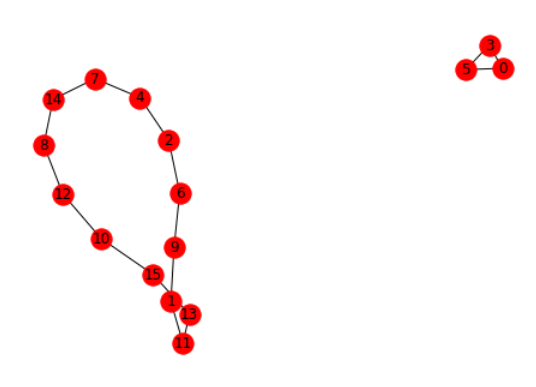


Figure 9 – 2nd Output From Feasibility Cuts



It can be shown that the previously implemented subtours for (4,5,6,1,9,10,8,7) (13,11,12,14,15) and (0,2,3) in Figure 6 were effective, as they are no longer in a subtours together. There is also a single growing tour that is dominating the minimum distance. It appears that the feasibility cuts are exposing an overarching TSP solution that is feasible. We still see a smaller subtour between (0,3,5). This is eliminated and the output reflected from the resulting IP in Figure 10, 11, and 12.

Figure 10 – 3nd Feasibility Cuts Added

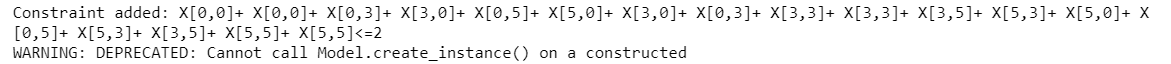


Figure 11 – 3nd Output Decision Matrix From Feasibility Cuts

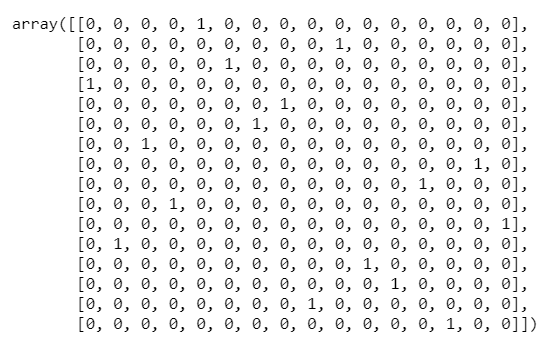
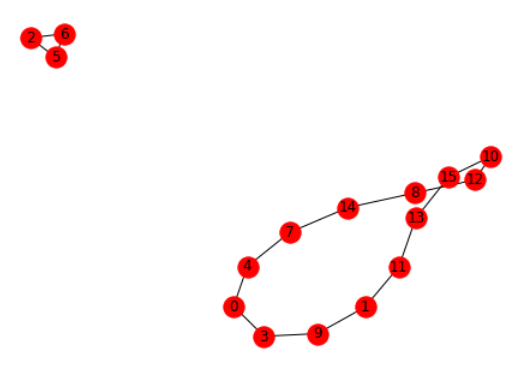


Figure 12 – 3nd Output From Feasibility Cuts



## Optimal Solution

Identical to Figure 9, Figure 12 shows another subtour has arisen between (2,6,5). After adding one more feasibility cut between those nodes the final solution is had in Figures 13, 14, and 15.

Figure 13 – Final Feasibility Cuts Added

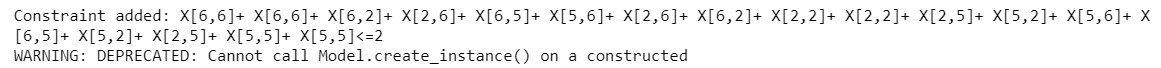


Figure 14 – Final Output Decision Matrix From Feasibility Cuts

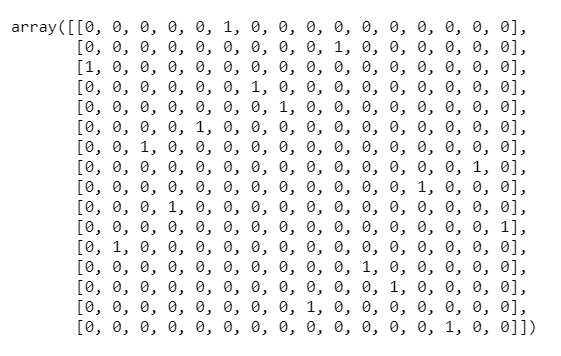
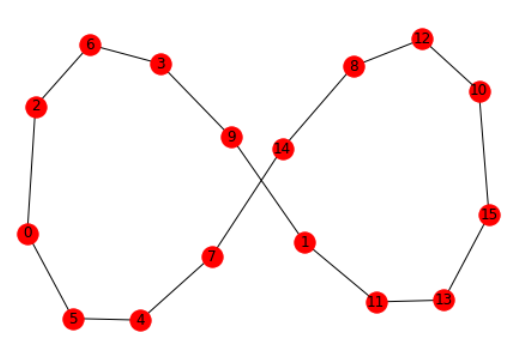


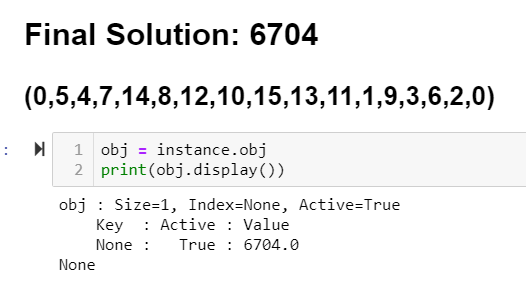
Figure 15 – Final Output From Feasibility Cuts



## Conclusion

We see in Figure 15 that the optimal solution is a valid SP tour with the following solution and distance found in Figure 16. We see that the distance was 6704 miles traveling from city 0 all the way back to city 0, traveling to every city exactly once.

Figure 16 – Final TSP Solution From Feasibility Cuts



# References

The full code and data file used for this solution are below. Technologies used were Pyomo with the GLPK MILP solver. Only the first 16 cities were used from the data file.





