Enhanced Approaches to Solving the Multiple Traveling Salesman Problem

Conference Paper · January 2015

DOI: 10.2514/6.2015-0889

3 authors:

Nathan Boone
University of Cincinnati

Anoop Sathyan
University of Cincinnati

Kelly Cohen
University of Cincinnati

AIAA SciTech
5-9 January 2015, Kissimmee, Florida
AIAA Infotech @ Aerospace
AIAA 2015-0889

Enhanced Approaches to Solving the Multiple Traveling Salesman Problem

Nathan Boone[1], Anoop Sathyan[2], and Dr. Kelly Cohen[3]
University of Cincinnati, Cincinnati, Ohio, 45221

Two modifications to the solution methods for the Multiple Traveling Salesman Problem (MTSP) will be presented with a focus on their potential applications to UAV swarm route planning. The scenario discussed in this paper involves a set of randomly distributed cities and a specified number of UAVs. To solve the MTSP, cities are divided into clusters based on their location in space, typically using K-means clustering, and each UAV is assigned to a cluster. Next the Traveling Salesman Problem (TSP) is solved for each cluster using a constructive heuristic called the 2-Opt to determine the shortest route through the set of cities. An optimal solution to the problem is one that minimizes the longest route of any UAV. The first modification that will be presented is a modification to the 2-Opt which improves computation time by beginning with an initial route determined using the nearest neighbor algorithm, rather than a route determined at random. The second modification is a new clustering method that builds upon K-means clustering to reduce the maximum tour distance of any cluster. These results could lead to large reductions in UAV mission times.

Nomenclature

TSP	=	Traveling Salesman Problem
MTSP	=	Multiple Traveling Salesman Problem
NN	=	Nearest Neighbor
Max/Largest Distance	=	Largest tour distance of any cluster
Max Minus Min Distance	=	Largest tour distance of any cluster minus the smallest of any cluster

I. Introduction

THE Traveling Salesman Problem is a challenging optimization problem that involves finding the shortest tour through a given set of cities that visits each city once and returns to the starting point[1]. One more complicated variation of this problem, the Multiple Traveling Salesman Problem (MTSP), involves more than one traveling agent, making it more relevant to real-world scenarios. Perhaps one of its most important applications is in the case of several UAVs that must visit a set of targets in the most efficient way possible. As UAVs grow in importance, so too will the uses of the MTSP as a route planning method. The uses of the MTSP are not limited to UAVs. Another situation in which the MTSP arises is in the case of GPS satellite networks that must coordinate the observation sessions of multiple receivers and find the best order of sessions for the receivers. Furthermore, the MTSP can be found in the school bus routing problem, in which it is necessary to minimize the routes of all the buses while ensuring that none are overloaded[2,3]. Due to its wide applications, finding improved methods for solving the MTSP was the topic of this research.

A. MTSP Solution Approaces

The MTSP may be solved effectively by first clustering the points based on their location in space using K-means clustering. The clusters are then solved individually using an algorithm called the 2-Opt[4]. The 2-Opt is a constructive heuristic that modifies an initial tour by swapping the endpoints of two route edges. After switching the endpoints, it checks to see if this results in a shorter path length. If path length was reduced, the algorithm makes

[1] Undergraduate Student, Department of Aerospace Engineering and Engineering Mechanics, University of Cincinnati, AIAA Student Member.
[2] PhD Student, Department of Aerospace Engineering and Engineering Mechanics, University of Cincinnati, AIAA Student Member.
[3] Associate Professor, Department of Aerospace Engineering and Engineering Mechanics, University of Cincinnati, AIAA Associate Fellow.

this change permanent and moves on to another edge pair. In our research, we have sought to improve each aspect of this MTSP approach: in both the method of clustering the points and how the clusters are solved.

In the MTSP's application to UAVs, it is important that the maximum distance of any cluster is minimized so that the total mission time is minimized[5]. If all UAVs travel approximately the same, short distance, then a mission will take less time to complete. K-means clustering does not take this factor into account. By creating a clustering method that uses information specific to the MTSP, we have resulted in a clustering method that significantly reduces the maximum distance of any cluster when compared to K-means.

The clustering method we have designed makes use of fuzzy logic. Fuzzy logic has demonstrated its effectiveness in a growing variety of meaningful applications, including aerospace, because of its ability to emulate human reasoning in order to solve vague or uncertain problems[1,6]. The basic premise of fuzzy logic is that variables can have truth values between 0 and 1, rather than either 0 or 1 as in traditional binary logic. In a fuzzy system, membership functions convert inputs to fuzzy truth values, and a series of if-then rules map these inputs to outputs. These fuzzy truth values allow an element to belong to multiple sets to varying degrees. One element may have a membership of 0.7 in one set and, at the same time, a membership of 0.3 in another. This means that several of the if-then rules can be activated at once, each to varying degrees. The output of the fuzzy system is a single, crisp value that reflects the degree to which each rule was activated[1].

II. The Nearest Neighbor Improvement

The 2-Opt's initial tour has usually been created by simply visiting each target in a random order. We tested the hypothesis that instead creating an initial tour with the nearest neighbor (NN) algorithm would reduce the 2-Opt's computation time by allowing the algorithm to converge more quickly. The NN algorithm, also called the "greedy" algorithm, builds a tour by visiting the closest target to its current location. This method is quick but not very accurate, and while the first distances in the tour tend to be small, the last ones are often very large.

Testing this modification did show a slight reduction in computation time with no substantial change in the tour's optimality. Table 1 shows the data collected, with the nearest neighbor data normalized to the random initial tour method data for comparison. The relative time savings are especially large for smaller numbers of cities, but due to larger computation times, larger numbers of cities lead to greater raw time savings as shown in Figure 1. These time savings are especially important in the MTSP, where they add up due to solving the TSP multiple times.

Table 1. NN 2-Opt results relative to random 2-Opt for a single TSP.

Cities	Mean NN Times	Trials that Decreased in Time	Mean NN Distances	Trials that Decreased in Distance
20	0.863251	86%	0.995458	40%
50	0.883907	79%	0.993267	54%
100	0.92322	67%	1.002419	48%

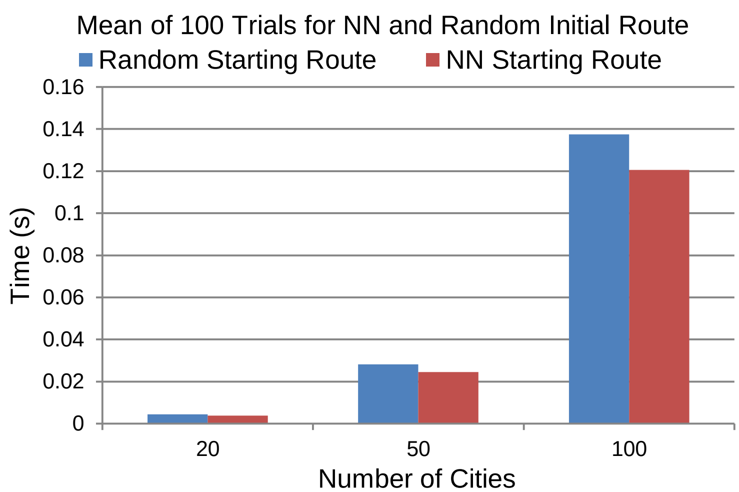

Figure 1. Computation times for each method.

III. Improved Clustering Method

A. Methodology

An algorithm called "TSPCluster" was created to implement a new clustering method designed for the MTSP. The main idea behind the clustering is to first conduct an initial K-means clustering and modify it by taking points from the cluster with the largest tour distance and adding them to one of the smaller clusters. To determine which points to switch, convex hulls, fuzzy logic, and the TSP solution are used.

Figure 2 provides a flowchart describing how the algorithm works. After solving each cluster to determine which is the largest in terms of tour distance, the algorithm must determine which of these points to move to a different cluster. It would be infeasible to test every point in the cluster's impact on the tour distance, so the search space must be limited. To do this, a convex hull is created around the largest cluster. A convex hull is the largest polygon that encloses a certain set of points, and can be thought of as a rubber band stretched around the outer edge of a cluster. Only considering points that are on the convex hull of the largest cluster also ensures that the algorithm only selects points that are on the far outer edge of the clusters, which tends to result in a far more optimal clustering.

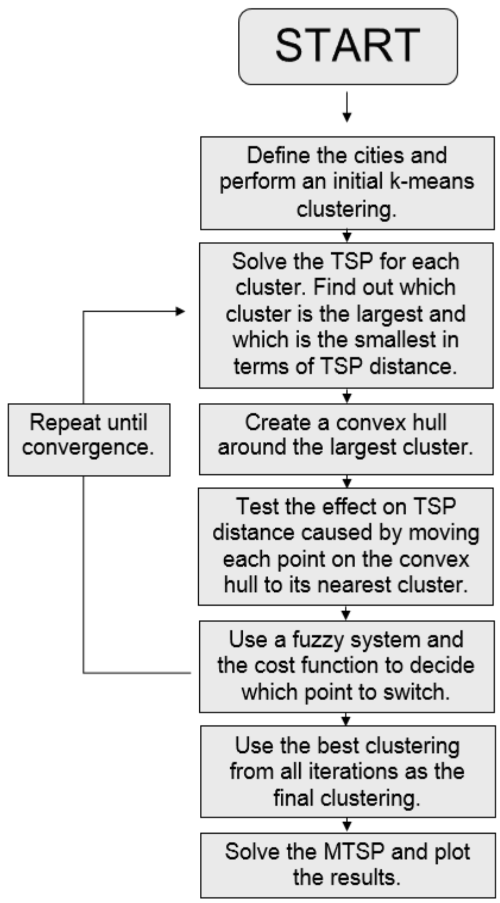

Figure 2. The clustering algorithm.

Once the algorithm has identified candidates using the convex hull, it individually tests the optimality of moving each point to its nearest cluster. Optimality is determined using two measures of cost: max distance and max minus min distance. Max distance is the largest tour distance of any cluster, while max minus min distance is the difference between the largest tour of any cluster and the smallest. Minimizing max distance is the overall goal of the algorithm, but max minus min distance is an important measure of how close the solution is to being optimal. Max minus min distance would be zero in a perfect clustering. The percent change in max distance and max minus min distance for each convex hull point form the inputs to a fuzzy system that determines membership grades, which describe how likely a particular point is to move. The membership functions for this fuzzy system are shown in Figure 3.

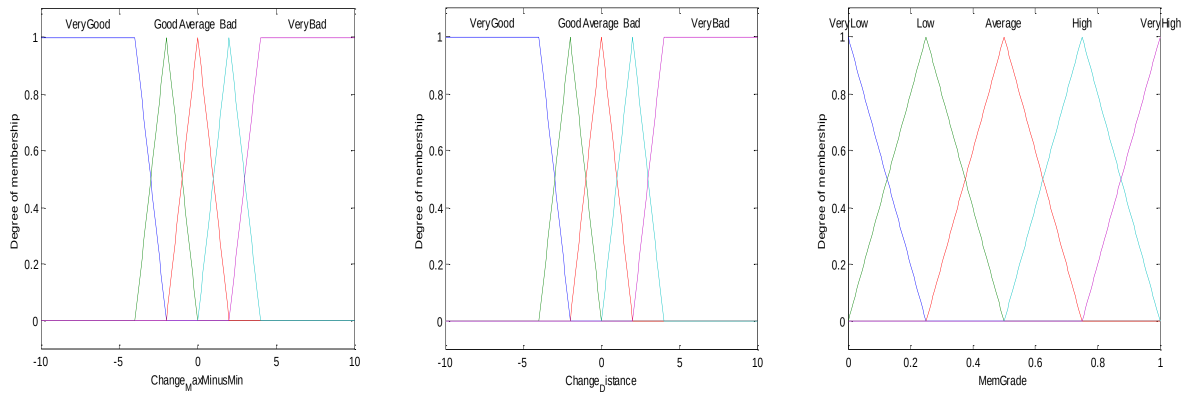

Figure 3. The membership functions for the two-input one-output fuzzy system.

After the fuzzy system has determined the membership grades, the point with the largest membership grade is selected. The cluster identifier vector is changed to reflect this point's move from the largest cluster to its nearest neighbor cluster. The algorithm repeats until it has gotten stuck at a certain clustering or until it has completed a certain number of iterations. Before returning the final clustering, the algorithm looks back at all its previous iterations and selects the clustering that gave the lowest value of max distance. This ensures that max distance will always be reduced or unchanged relative to K-means, and that it is reduced as much as possible.

B. Achieving Better Solutions

Solution quality may be slightly improved by introducing more random chance once the algorithm has converged to prevent local minima. If an option in the TSPCluster function is enabled, and the algorithm has converged, then for the next iteration a point will be randomly selected out of the points with the four best membership grades. Enabling this typically allows for a 1-2% additional decrease in the average largest tour distance, but it can also cause around a 50% increase in computation time. The user can decide if this tradeoff makes sense for their application.

IV. Results of the Clustering Method

The largest tour distances and the max minus min distances obtained by K-means clustering and TSPCluster over 100 trials of a 6-cluster, 300-city MTSP are shown in Figure 4. The cities were randomly distributed over a 10x10 grid. These plots show that TSPCluster gives fairly large improvements over K-means consistently. Largest tour distance decreased by an average of 14.3% when compared to K-means in this problem, with each one of the 100 trials decreasing in largest distance. Random variations were not used in any of the results presented in this section.

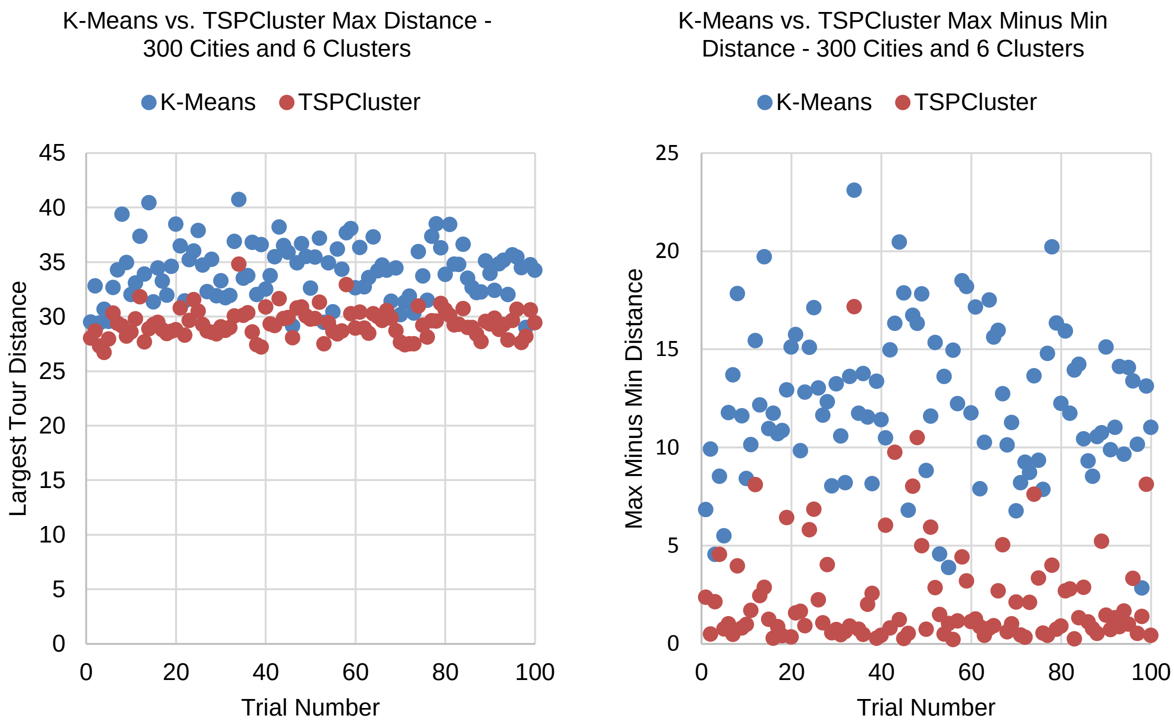

Figure 4. (a) Largest tour distance for each method and (b) max minus min distance for each method.

Figure 5 shows how TSPCluster's computation time scales for various numbers of cities. Since this clustering method is based on the 2-Opt, it scales similarly, following an approximately quadratic curve. The MTSP has traditionally been regarded as an NP-hard type problem, with a computation time that increases exponentially for more complex problems, so this shows that our method scales very well.

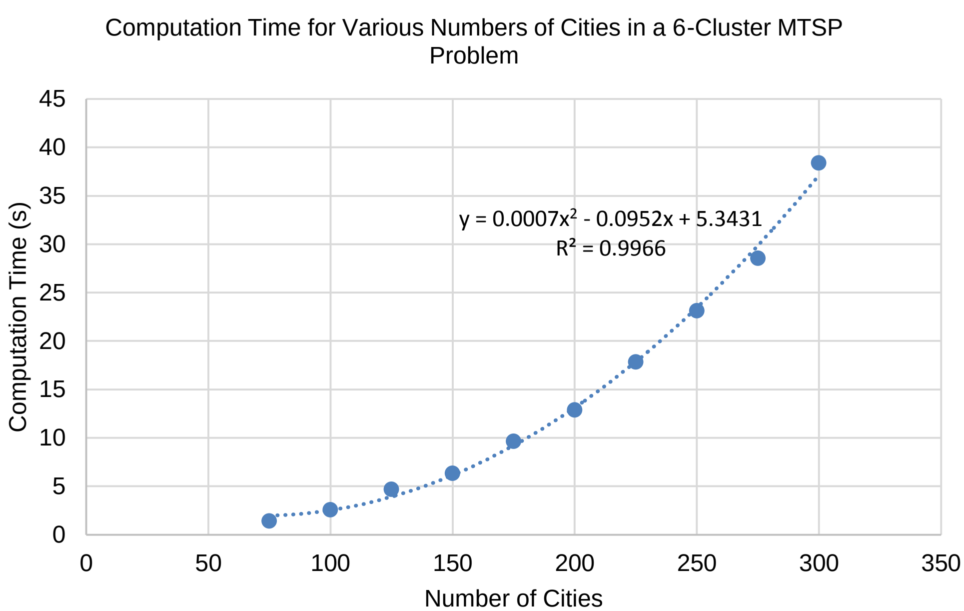

Figure 5. Computation time scalability.

In Figure 6, the TSPCluster largest distance normalized to the K-means largest distance is shown for varying numbers of clusters and cities. It can be seen that TSPCluster tends to perform better with more complex problems. This is because larger numbers of cities and clusters allow the algorithm more opportunities to reduce max distance. However, the normalized largest distance eventually flattens out, and it usually does not fall below 0.85 even for extremely complex problems unless random variations are introduced.

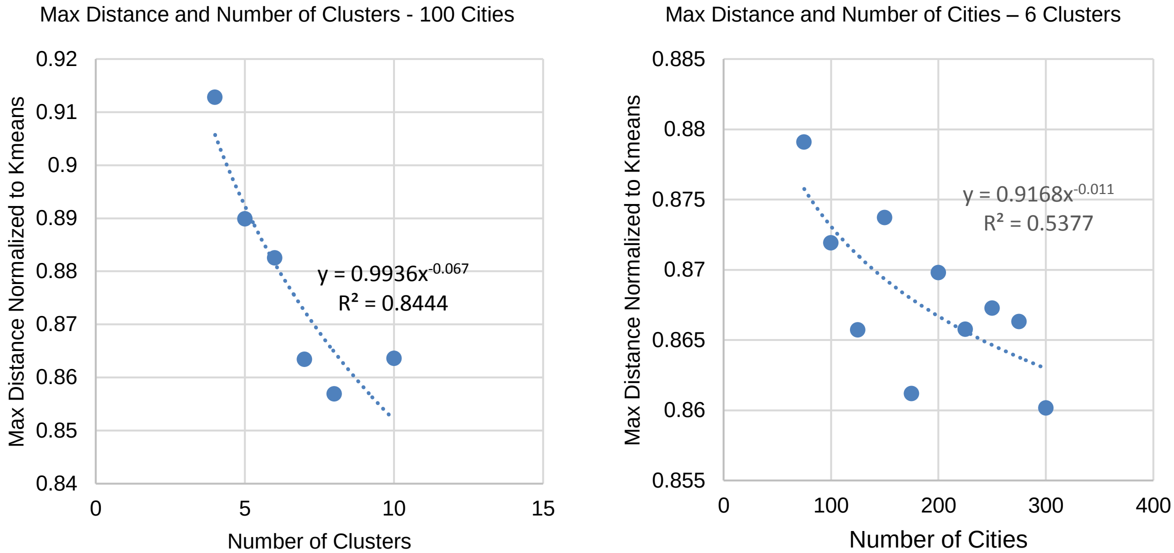

Figure 6. Largest distance scalability for (a) various numbers of clusters and (b) various numbers of cities.

TSPCluster's performances and computation times for problems of varying complexities are shown in Table 2. Though the clustering method gives very good results, this comes at the cost of an increase in computation time. However, complex problems can still be solved in a few seconds, and a faster computer could decrease the raw times substantially. Furthermore, in many applications of the MTSP reducing the largest tour distance is more important than reducing the computation time. In applications to UAV route planning, the mission times are much greater than the computation times, so reduction in mission times due to a smaller largest distance should be more than large enough to justify a few seconds of initial computation time.

Table 2. Results for various problem complexities.

Cities	Clusters	Max Distance	Max Minus Min Distance	Computation Time: New Method (s)	Computation Time: Normalized to K-Means
100	4	0.910744	0.175421	2.539029	83.650018
125	5	0.879573	0.167900	3.959156	104.46873
150	6	0.858509	0.179147	6.262300	136.76130
150	9	0.844209	0.291232	5.688773	154.86235
200	8	0.850227	0.270908	11.54757	182.82394
500	10	0.873489	0.456221	74.66213	252.54932

V. Conclusion

This research has presented improvements to each of the two components of the MTSP solution: in both the clustering approach and the method for solving the clusters. The nearest-neighbor-first modification to the 2-Opt was shown to be effective because it is a simple way to decrease computation time with no drawbacks. This allows for an even faster 2-Opt, which was already among the best algorithms for solving the TSP. The results from the clustering method also show promise. The reduction in the largest tour distance is especially important for UAV route planning, and a 15% reduction in max tour distance means mission times would be reduced by a similar amount. Another important advantage of this method is that it can allow a mission to be conducted with fewer UAVs relative to K-means while still achieving the same largest tour distance as with K-means.

Future work could include the application of the nearest-neighbor-first idea to other TSP solution methods such as the Kernighan-Lin algorithm. Another possibility would be to try applying other algorithms or possibly develop a new algorithm to form the 2-Opt's initial route. This could result in an even faster approach to the TSP. Additionally, future work might also involve improving the performance or efficiency of the clustering method. It is not known how close our clustering method is to the optimal solution, and it would be interesting to see if the clustering can be improved any further. It may be possible to bring max minus min distance closer to zero and result in a more optimal solution. Another potential source of improvement may be to replace the algorithm's TSP solver with something that achieves the same results without the computation time.

VI. References

[1]Mitchell, S., Ernest, N., and Cohen, K., "Comparison of Fuzzy Optimization and Genetic Fuzzy Methods in Solving a Modified Traveling Salesman Problem," AIAA Paper 2013-4664, AIAA Infotech@Aerospace Conference, Boston, MA, 19-22 August 2013.

[2]Bektas, T., "The Multiple Traveling Salesman Problem: An Overview of Formulations and Solution Procedures", the Omega International Journal of Management Science, Omega 34 (2006) pages 209-219.

[3]Rajesh Matai, Surya Singh and Murari Lal Mittal (2010), "Traveling Salesman Problem: an Overview of Applications, Formulations, and Solution Approaches", ISBN: 978-953-307-426-9, InTech.

[4]Sathyan, A., and Cohen, K., "Comparison of Approximate Approaches to Solving the traveling Salesman Problem", 39th Annual AIAA Dayton-Cincinnati Aerospace Sciences Symposium, Paper 39DCASS-042, March 5, 2014, Dayton, OH.

[5]25Campbell, A. M., Vandenbussche, D., and Hermann, W., "Routing for Relief Efforts," Transportation Science, Vol. 42, No. 2, May 2008, pp. 127-145.

[6]Sathyan, A., Ernest, N., and Cohen, K., "Genetic Fuzzy Approach for Control and Task Planning Applications," Abstract for AIAA Infotech@Aerospace Conference, Kissimmee, FL, 5-9 January 2015.