Traveling Salesman Algorithm

Author: Fred Strohm

Abstract: A polynomial time algorithm for solving the Traveling Salesman Problem is described.

The Traveling Salesman Problem asks the following question: "Given a list of cities and the distances between each pair of cities, what is the shortest possible round trip that visits each city exactly once and returns to the origin city?" In the asymmetric version of the problem, the distance departing from city A with the destination being city B, may be somewhat different than the distance departing from city B with the destination being city A. By contrast, the symmetric version of the problem always sets distances between two cities to be the same in either direction. This paper describes an algorithm for solving the asymmetric Traveling Salesman Problem.

The round trip through all n cities starting and ending at the origin city can be conveniently represented as a round trip table of n links. Figure 1 shows an example of such a table.

From	To	Miles

A	C	48.4
C	E	278
E	B	318
B	D	58.4
D	A	54.7
Total		757.5

Figure 1

Each link names a departure city and a destination city and gives the distance in miles from the departure city to the destination city. A proper round trip table of n links always starts with a link naming the origin city as departure city, every subsequent link's departure city being the previous link's destination city, and ending with a link naming the origin city as the final destination city. The sum of the distances in the links in a round trip table is the total length of the round trip represented.

The actual shortest round trip could also be described by such a round trip table. Figure 1 is the shortest round trip table for the example with 5 cities which I shall be discussing.

In our algorithm for finding the shortest round trip, we first set up an initial round trip table, in a manner soon to be described. The goal is then to convert this initial round trip table into the round trip table for the actual shortest round trip. The algorithm searches for ways of changing links in the initial round trip table, so as to shorten the round trip. We consider changing links in the round trip table by exchanging destination cities between either 3 or 4 links at a time. We change 3 or 4 links at a time instead of just 2, because just 2 changes at a time will result in closed loops (incomplete shorter round trips) of less than n cities, as will be seen shortly. The distance from the departure city to the destination city in a link changes when the destination city is replaced by another city. If the sum of the distances in the 3 or 4 links being considered is less after the exchange of destination cities than the sum of the distances before the exchange, the link change becomes a candidate for changing the round trip table.

With our algorithm, we search for these 3 or 4 link changes that would shorten the round trip until no more can be found. As will soon be seen, this search for 3 or 4 link changes that would shorten the round trip is carried out in such a way that when no more can be found, the round trip table at that point is guaranteed to represent the shortest round trip through the n cities.

I shall illustrate my algorithm with an example of 5 cities. The 5 cities are A,B,C,D and E, with origin city A. Figure 2 shows the distance in miles from each departure city to every destination city, represented in matrix form. Since we are treating the asymmetric problem, the matrix is also asymmetric. Figure 1 is the shortest round trip table for these 5 cities. It was found by comparing all 4! possible round trips.

			T	O		
		A	B	C	D	E
F	A		12.3	48.4	54.9	325
R	B	11.8		41.3	58.4	318
O	C	48.5	40.5		99	278
M	D	54.7	58.5	99		377
	E	326	318	279	377	

Figure 2

Now, to construct our initial round trip table, the first link is made from the origin city to the closest destination city to it. The departure city in every subsequent link Is set to the destination city in the previous link. And its destination is set to the closest destination city to it, excluding cities that have already appeared as destination cities in the table. The last link has the origin city as destination.

Figure 3 is the initial round trip table we construct for our 5 city example.

From	To	Miles
A	B	12.3
B	C	41.3
C	D	99
D	E	377
E	A	326
Total		855.6

Figure 3

We now proceed to search for link changes that will reduce the total distance of the links in the initial round trip table. Due to the way we constructed the initial round trip table, the longer links are likely toward the bottom. So we start our search with the last link in the initial round trip table. And we consider exchanging its destination city with the destination city in links above it, one link at a time, starting with the first link in the table.

So we start by considering exchanging destination cities between the last link and the first link, changing links in Figure 3 for illustration first by only two at a time:

From	To	Miles
A	A	0
B	C	41.3
C	D	99
D	E	377
E	B	318
Total		835.3

Figure 4

We always exchange a current destination city for another current destination city, so changing the last link from E to A to A to E is not considered.

The reason we always exchange destination cities among 3 or 4 links instead of just 2 links can be seen here now. Exchanging destination cities between only 2 links always results in the creation of 2 closed loops (incomplete round trips of length less than n) in the round trip table. In this case we have a closed loop with length of only 1, a link from A to A. The other 4 links in the table form another closed loop, from B to C, from C to D, from D to E, and from E bo B. The closed loop that includes the link from the origin city can be called the top closed loop, the other loop being the bottom closed loop. The two closed loops of length less than n are eliminated by exchanging destination cities between a link in the top closed loop and a link in the bottom closed loop. This leaves a single loop of length n, a complete round trip.

So in this first case, after our initial exchange between link 5 and link 1, link 1 must exchange destination cities with link 2,3, or 4. And in our search for a link change that will shorten the round trip, we consider all of these.

Here are the possible three link changes we are considering:

Exchanging destination cities between link 1 and link 2 in figure 4 would yield the round trip table:

From	To	Miles
A	C	48.4
B	A	11.8
C	D	99
D	E	377
E	B	318

Total		854.2
Difference		1.4

Figure 5

Exchanging destination cities between link 1 and link 3 in Figure 4 would yield the round trip table:

From	To	Miles
A	D	54.9
B	C	41.3
C	A	48.5
D	E	377
E	B	318
Total		839.7
Difference		15.9

Figure 6

Exchanging destination cities between link 1 and link 4 in Figure 4 would yield the round trip table:

From	To	Miles
A	E	325
B	C	41.3
C	D	99
D	A	54.7
E	B	318
Total		838
Difference		17.6

Figure 7

We always compute the sum of the distances after a change we are considering, and the difference between this sum and the sum of the distances in the initial round trip table.

We have now considered all 3 or 4 link changes beginning with exchanging destination cities between link 5 and link 1. The third option would shorten the round trip more than the other two.

But we are not done with considering changes to link 5.

We need to consider all possible 3 or 4 link changes, starting with exchanging the destination city of link 5 with links above it, before we choose the change that most shortens the round trip. We still need to consider changing destination cities between link 5 and links 2,3 and 4..

So we next consider exchanging destination cities between link 5 and link 2, changing links in Figure 3 for illustration first by only 2 at a time:

From	To	Miles
A	B	12.3
B	A	11.8
C	D	99
D	E	377
E	C	279
Total		779.1

Figure 8

This results in a top closed loop of links 1 and 2 and a bottom closed loop of links 3, 4 and 5.

We consider all possible exchanges of destination cities between a link in the top closed loop and a link in the bottom closed loop. Except that we do not consider changing link 5 a second time.

First we consider exchanging destination cities between link 2 and links 3 and 4 in Figure 8.

Exchanging destination cities between link 2 and link 3 in Figure 8 would yield the round trip table:

From	To	Miles
A	B	12.3
B	D	58.4
C	A	48.5
D	E	377
E	C	279
Total		775.2
Difference		80.4

Figure 9

Exchanging destination cities between link 2 and link 4 in Figure 8 would yield the round trip table:

From	To	Miles
A	B	12.3
B	E	318

C	D	99
D	A	54.7
E	C	279
Total		763
Difference		92.6

Figure 10

Since link 1 is also in our top closed loop, we consider exchanging destination cities between link 1 and links 3 and 4.

When we do this, we leave the initial exchange between link 5 and link 2 in place. So we consider a change of 4 links.

Exchanging destination cities between link 1 and link 3 in Figure 8 would yield the round trip table:

From	To	Miles
A	D	54.9
B	A	11.8
C	B	40.5
D	E	377
E	C	279

Total		764.6
Difference		91

Figure 11

Exchanging destination cities between link 1 and link 4 in Figure 8 would yiled the round trip table:

From	To	Miles
A	E	325
B	A	11.8
C	D	99
D	B	58.5
E	C	279
Total		773.3
Difference		82.3

Figure 12

We have now considered all 3 or 4 link changes starting with exchanging destination cites between link 5 and links 1 and 2. Some of these link changes would shorten the round trip substantially. But we still have to consider exchanging destination cities between link 5 and links 3 and 4 in Figure 3.

So we next consider exchanging destination cities between link 5 and link 3, changing links in Figure 3 for illustration first by only 2 at a time:

From	To	Miles
A	B	12.3
B	C	41.3
C	A	48.5
D	E	377
E	D	377
Total		856.1

Figure 13

This results in a top closed loop of links 1, 2 and 3 and a bottom closed loop of links 4 and 5. So we need to consider exchanging destination cities between link 4 and links 1, 2, and 3. We do not consider changing link 5 a second time.

Exchanging destination cities between link 3 and link 4 in Figure 13 would yield the round trip table:

From	To	Miles

A	B	12.3
B	C	41.3
C	E	278
D	A	54.7
E	D	377
Total		763.3
Difference		92.3

Figure 14

When we exchange destination cities between link 4 and links 1 and 2, we leave the initial exchange between links 5 and 3 in place. So we change 4 links.

Exchanging destination cities between link 1 and link 4 in Figure 13 would yield the round trip table:

From	To	Miles
A	E	325
B	C	41.3
C	A	48.5
D	B	58.5

E	D	377
Total		850.3
Difference		5.3

Figure 15

Exchanging destination cities between link 2 and link 4 in Figure 13 would yield the round trip tabe:

From	To	Miles
A	B	12.3
B	E	318
C	A	48.5
D	C	99
E	D	377
Total		854.8
Difference		0.8

Figure 16

We have now considered all 3 or 4 changes starting with exchanging destination cities between link 5 and links 1, 2 and

3. Finally, when we consider exchanging destination cities between link 5 and link 4, this leads to the same link changes as Figure 7, Figure 10, and Figure 14.

Therefore we have now considered all possible 3 and 4 link changes, beginning with exchanging destination cities between link 5 and links above it. The change we found that would most shorten the initial round trip is shown in Figure 10. So before continuing, we go ahead and change the initial round trip table accordingly. Figure 10 becomes the current round trip table..

From	To	Miles
A	B	12.3
B	E	318
C	D	99
D	A	54.7
E	C	279
Total		763

Figure 10

This round trip table is not in proper order. But we do not put it back in proper order until after we complete the first pass through the round trip table. Links 2, 4, and 5 of this round trip table have been changed from the initial round trip table. So next we only consider changing links 1 and 3.

We start by considering exchanging destination cities between link 3 and link 1 in Figure 10, changing links for illustration first by only two at a time:

From	To	Miles
A	D	54.9
B	E	318
C	B	40.5
D	A	54.7
E	C	279
Total		747.1

Figure 17

The resulting top closed loop comprises links 1 and 4, A to D and D to A. The bottom closed loop is links 2, 3, and 5. We first consider exchanging destination cities between link 1 and links 2 and 5 in Figure 17. We do not consider changing link 3 a second time.

Exchanging destination cities between link 1 and link 2 in Figure

17 would yield the round trip table:

From	To	Miles
A	E	325
B	D	58.4
C	B	40.5
D	A	54.7
E	C	279
Total		757.6
Differnce		98

Figure 18

Exchanging destination cities between link 1 and link 5 in Figure 17 would yield the round trip table:

From	To	Miles
A	C	48.4
B	E	318
C	B	40.5

D	A	54.7
E	D	377
Total		838.6
Difference		17

Figure 19

However, link 4 is also in the top closed loop. So we need to consider leaving links 1 and 3 the same after their initial exchange, and then exchanging destination cities between link 4 and links 2 and 5 in the bottom closed loop. We do not consider changing link 3 a second time. So 4 links would be changed.

Exchanging destination cities between link 4 and link 2 in Figure 17 would yield the round trip table:

From	To	Miles
A	D	54.9
B	A	11.8
C	B	40.5
D	E	377

E	C	279
Total		763.2
Difference		92.4

Figure 20

Exchanging destination cities between link 4 and link 5 in Figure 17 would yield the round trip table:

From	To	Miles
A	D	54.9
B	E	318
C	B	40.5
D	C	99
E	A	326
Total		838.4
Difference		17.2

Figure 21

We have now consiered all the 3 or 4 link changes starting with exchanging destination cities between link 3 and link 1 in figure 10. Of all the link changes we have just considered, the one that would shorten the round trip the most was Figure 18. So we change the current round trip table accordingly. Figure 18 becomes our current round trip table.

From	To	Miles
A	E	325
B	D	58.4
C	B	40.5
D	A	54.7
E	C	279
Total		757.6

Figure 18

Since all links in the initial round trip table of Figure 3 have now been changed, we have now completed the first pass through the initial round trip table. Before proceeding to the second pass, we put Figure 18 in proper order.

From	To	Miles
A	E	325
E	C	279
C	B	40.5

B	D	58.4
D	A	54.7
Total		757.6

Figure 22

Having completed pass 1 through the initial round trip table of Figure 3, we next start pass 2 through the round trip table in Figure 22, which was the output of pass 1. Pass 2 is conducted using the same method as pass 1.

We find no way to shorten the round trip in Figure 22 by any link changes starting with exchanging destination cities between link 5 and links above it. And we find no way to shorten the round trip in Figure 22 by link changes starting with exchanging destination cities between link 4 and links above it. To save space, the link changes considered are not shown.

We next consider exchanging destination cities between link 3 and links above it in Figure 22. First we eschange destination ciites between link 3 and link 1, changing links in Figure 22 for illustration first by only two at a time:

From	To	Miles
A	B	12.3

E	C	279
C	E	278
B	D	58.4
D	A	54.7
Total		682.4

Figure 23

This results in a top closed loop of links 1, 4 and 5, and a bottom closed loop of links 2 and 3 in Figure 23. We do not consider changing link 3 a second time. So we need to consider exchanging destination cities between links 1, 4 and 5 in the top closed loop, and link 2 in the bottom closed loop, in Figure 23.

Exchanging destination cities between link 1 and link 2 in Figure 23 would yield the round trip table:

From	To	Miles
A	C	48.4
E	B	318
C	E	278

B	D	58.4
D	A	54.7
Total		757.5
Difference		98.1

Figure 24

When we exchange destination cities between links 4 and 5 In the top closed loop, and link 2 in the bottom closed loop, in Figure 23, we leave the initial exchange between links 3 and 1 in place.   So we change 4 links.

Exchanging destination cities between link 4 and link 2 in Figure 23 would yield the round trip table:

From	To	Miles
A	B	12.3
E	D	377
C	E	278
B	C	41.3
D	A	54.7
Total		763.3

Differfence		92.3

Figure 25

Exchanging destination cities between link 5 and link 2 in Figure 23 would yield the round trip table:

From	To	Miles
A	B	12.3
E	A	326
C	E	278
B	D	58.4
D	C	99
Total		773.7
Difference		81.9

Figure 26

We have considered all link changes starting with exchanging destination ciites between link 3 and link 1 in Figure 22. No link changes starting with exchanging destination cities between link 3 and link 2 in Figure 22 shorten the round trip in Figure 22. To save space, the link changes considered are not shown.

Having considered all 3 and 4 link changes starting with exchanging destination cities between link 3 and links above it in Figure 22, we found that Figure 24 shows a shortening of the round trip in figure 22. So we change the current round trip table accordingly. Figure 24 becomes our current round trip table.

From	To	Miles
A	C	48.4
E	B	318
C	E	278
B	D	58.4
D	A	54.7
		757.5

Figure 24

Links 5 and 4 in Figure 22 have been passed over without change. And links 3, 2 and 1 in Figure 22 have been changed in Figure 24. So we have now completed the second pass through the round trip table. **B**efore proceeding to the third pass, we put Figure 24 in proper order.

From	To	Miles

A	C	48.4
C	E	278
E	B	318
B	D	58.4
D	A	54.7
Total		757.5

Figure 27

Figure 27 is identical to Figure 1, the round trip table for the shortest round trip through our 5 cities.

So in pass 3 through Figure 27, no way will be found to shorten the round trip by 3 or 4 link changes, starting with exchanging destination cities between links 5, 4, 3, or 2 and links above them. To save space, the link changes considered are not shown.

Having completed the third pass through the current round trip table without finding a way to shorten the round trip, we conclude that the round trip table output by the second pass represents the shortest round trip through the 5 cities.

I will now explain why, when a pass through the current round trip table fails to find a way to shorten the round trip, the round trip table output by the previous pass represents the shortest round trip through the n cities.

The question is whether there could be discrepancies between the current round trip table and the shortest round trip table, discrepancies which made the round trip represented by the shortest round trip table shorter than the round trip represented by the current round trip table, and yet a complete pass of my method through the current round table fails to find any 3 or 4 link change that would shorten the round trip.

The minimum number of discrepancies possible between two round trip tables is 3 or 4. When a link in the current round trip table has the same departure city but a different destination city compared to the shortest round trip table, I will say that link is erroneous and has the wrong destination city and needs the destination city that would make it match the shortest round trip table. There cannot be just one discrepancy. For if one link in the current round trip table has the wrong destination city, the link assigned the destination city it needs must also have the wrong destination city. And there cannot be only two discrepancies. For if two links have each other's needed destination city, for them to exchange destination cities would create two closed loops in the round trip table, which cannot exist in any proper round trip table.

So if the current round trip table does not match the shortest round trip table, and therefore the round trip represented by the shortest round trip table is shorter than the round trip represented by the current round trip table, there is at least one 3 or 4 link discrepancy between the two tables, such that those links in the shortest round trip table sum to a shorter distance than the corresponding links in the current round trip table.

And our question is, given this situation, could a complete pass through the current round trip table using my method fail to find and correct this 3 or 4 link discrepancy?

In a pass through the round trip table using my method, we would consider exchanging destination cities between every pair of two links, and we would consider every 3 or 4 link change that would eliminate the two closed loops resulting from this initial exchange.

Now consider 3 or 4 links in the current round trip table that have the same departure city as 3 or 4 links in the shortest round trip table, but the destination cities of these links are switched compared to the corresponding links in the shortest round trip table. So none of these 3 or 4 links in the current round trip table match the corresponding links in the shortest round trip table.

If there are 3 discrepancies between the current round trip table and the shortest round trip table, when we considered exchanging destination cities between the first of the three erroneous links and the second, and then considered eliminating the resulting two closed loops by exchanging destination cities between the second and the third erroneous link, we would find the link change that would shorten the round trip.

The only way this could be prevented is if the second and third erroneous link were in the same closed loop. Because then exchanging destination cities between them would leave us with two closed loops. But we are assuming these 3 links are the only links in the current round trip table that do not match the shortest round trip table. So if making the 3 links match the shortest round trip table would leave two closed loops, the shortest round trip table would include two closed loops, which is impossible.

If there are 4 discrepancies between the current round trip table and the shortest round trip table,, when we considered exchanging destination cities between the first of the four erroneous links and the second, and then considered eliminating the resulting two closed loops by exchanging destination cities between the third and the fourth erroneous link, we would find the link change that would shorten the round trip.

The only way this could be prevented is if the third and fourth erroneous link were in the same closed loop. Because then exchanging destination cities between them would leave us with two closed loops. But we are assuming these 4 links are the only links in the current round trip table that do not match the shortest round trip table. So if making the 4 links match the shortest round trip table would leave two closed loops, the shortest round trip table would include two closed loops, which is impossible.

Because in a pass through the round trip table using my method, we would consider exchanging destination cities between every pair of two links, and every 3 or 4 link change that would eliminate the two closed loops resulting from this initial exchange, we could not fail to consider making this 3 link change or this 4 link change that would eliminate the discrepancies between the current round trip table and the shortest round trip table.

Therefore, if a complete pass is made through the current round trip table using my method, with no way found to shorten the round trip, at that point there must be no discrepancies between the current round trip table and the shortest round trip table. So my method is guaranteed to find the shortest possible round trip that visits each city exactly once and returns to the origin city.

In a pass through an n link round trip table using my method, the number of initial 2 link changes to consider is in the worst case $n(n-1)/2$. And for each of these initial 2 link changes, the number of exchanges between a link in the top closed loop and a link in the bottom closed loop to consider is in the worst case $((n-1)/2)^2$,   If it takes c passes through the round trip table before a pass fails to find any 3 or 4 link change that shortens the round trip, the cost of finding the shortest round trip using my method is less than the time to consider:

$$c*(n(n-1)/2)*((n-1)^2)/4$$

3 or 4 link changes.   So my method for finding the shortest possible round trip that visits each city exactly once and returns to the origin city is clearly a polynomial time algorithm.