Investigation of the Traveling Salesman Problem using a Generalised Ising model

By David Williams 14316860

Abstract

Investigation of the Traveling Salesman Problem for randomly distributed points in a plane and the 48 Capitals tour using a Generalised Ising model. We used a Generalised Ising model as well as a simulated annealing model in order to find the shortest route in an acceptable time frame. We also examined the tour of the 48 USA state capitals and found the accepted optimal solution.

Introduction

The Traveling Salesman problem (TSP) is on the surface a very simple problem. Given a number of cities find the shortest closed route that visits every city exactly once. While on the surface this may seems simple, and it is if the number of cities is in the region of 4 or 5, the difficulty increases dramatically as the number of cities increases. For example, if there are just 10 cities there is already 181,440 possible routes between them. Because of this it is impossible to find an exact solutions to this problem for large number of cities in a realistic time complexity. In computer science it is probably the best know example of a NP-Complete problem.

Methods

We used a Generalised Ising Model as well as simulated annealing in order to find a solution that may not be the optimum solution but will be very close.

We define a Hamiltonian for our system, the natural one to use being the total route distance. This total distance is the sum of the distances between the cities following a given route. This is easy to do with a computer but slightly harder to quantify mathematically. Let $x_{i,j}$ be the distance between cities i and j and $c_{i,j}$ be equal to one if i is connected to j and zero otherwise. So we have a state S = ($c_{1,2}$, $c_{1,3}$,…,$c_{N-1,N,}$ $c_{1,N}$) which for a given route between the cities is then (1,1,0,0…,0,1) and its length is given by.

$$H(\vec{S}) = \sum_{i<j} c_{ij} x_{ij}$$

It can be derived that for N cities randomly and independently distributed over some area A that the minimum path length L is

$$L = k\sqrt{N}\sqrt{A} \ \ lim\ n \rightarrow \infty$$

where k is some constant which has been approximated to 0.765. It should be noted that while this is in the limit of n to infinity that for “fairly convex” areas that n does not have to be very large, n>=15 will do.

The Algorithm we use is as follows:

Create a graph of N cities and plot a random route between them and set an initial temperature

1. Swap the position of two cities in this route
2. Find the difference in length between the new and old route

3. If the new route is shorter then it is accepted if it is longer it may be accepted if
$$e^{-\Delta H/T} > r$$
where ΔH is the difference in length, T is the current temperature of the system and r is a random number between 0 and 1.
4. The temperature is then decreased (according to a schedule)
5. Steps 2-5 are carried out until the temperature has reached a predefined minimum

The route produced is then an approximation of the shortest route. During step 3 there is a number of different ways to find the difference in route length. The simplest is to find the total length of the old route and total length of the new route and subtract them. However, if there is a large number of cities there is allot of extra operations to do per iteration and this will slow the program significantly.

If we have an old order, such as (4, 6, 1, 3, 2, 8, 9, 5, 7, 4) that is to say start at city 4 then go to 6 then go to 1 and so on back to 4 and a new route where we swap the position of 1 and 5 so that we have the sequence (4, 6, 5, 3, 2, 8, 9, 1, 7, 4), it can be noted that the only cities that affect the total length difference are the immediate neighbours of the switched cities. It can then be determined that if city x and y are swapped in route that.

$$old = d(x-1,x) + d(x,x+1) + d(y-1,y) + d(y,y+1)$$
$$new = d(x-1,y) + d(y,x+1) + d(y-1,x) + d(x,y+1)$$
$$\Delta H = new - old$$

where d(a,b) is a function that gives the distance between points a and b. There can be some boundary problem if x or y is the first or last point in the list but this can fixed using a few if statements and some relabeling in our program.

In step 5 we cool down the temperature according to a schedule. There is a number of ways to do this. We have chosen a method where we have a cooling factor, which is a number between zero and one and at each iteration we multiply the current temperature by this number. So in order to bring the initial temperature to the final temperature it takes
$n = log_{Cf}(\frac{t_f}{t_i})$ where cf=cooling factor, t_f=final temp and t_i= initial temperature.

Results and discussion

The goal of this investigation was to see if a generalised Ising model and simulated annealing could give us solutions to the traveling salesman problem in a reasonable timeframe. The time it takes depends a lot on the initial conditions.

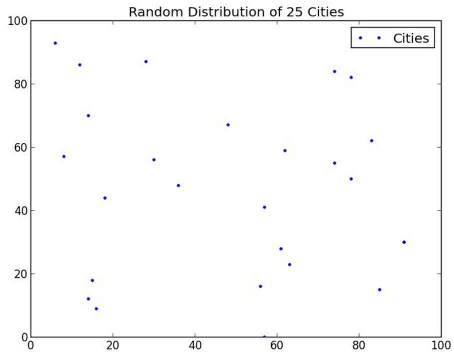

fig.1

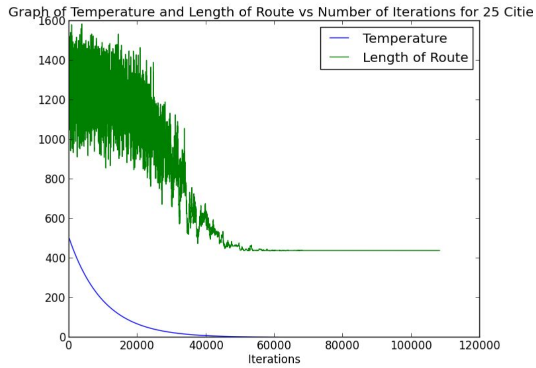

fig.2

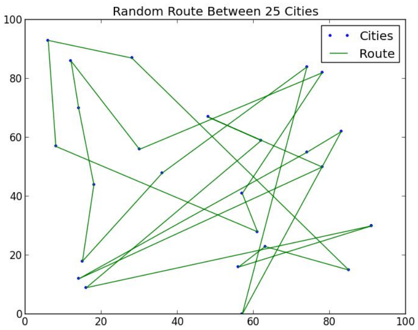

fig.3

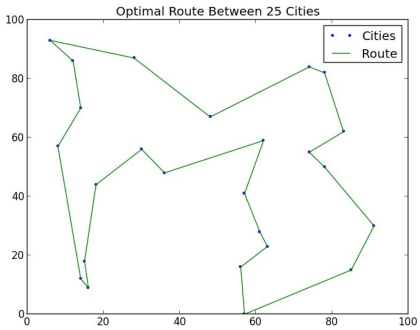

fig.4

We used an initial temperature of 500 and a cooling factor of .9999 and we can see from the above graphs that after 60,000 iterations that we have gotten from a route length of 1400 to a final route length of 478. Using our analytical estimate formula for L we get a value 382.5 which considering our n is relatively small is quite close to our computational result. The resulting route looks like a reasonable one. We can see in fig.2 that when the temperature is high there is a far great chance of the route switching to one that is a lot longer, however as the temperature decrease the chance of this happening diminishes as when the temperature gets to zero only routes that are shorter are accepted.

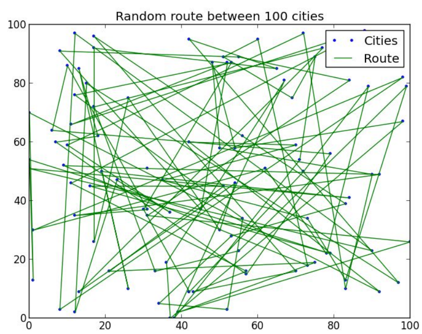

fig.5

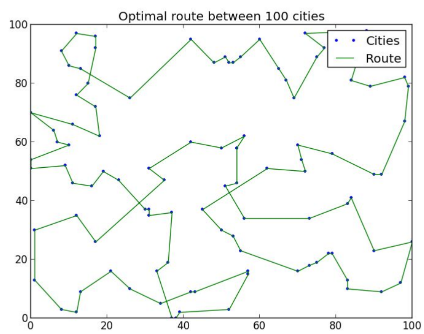

fig.6

We run our program again for 100 cities and because there are far more possible routes we increase the number of iterations by increasing our cooling factor to .99999. We started with a random route length of 5776 and got a final route length of 898. Comparing this again with our approximation value of 765 we see that we are getting closer and closer to the expected result as n increases.

We then looked at the shortest route possible between the 48 mainland state capitals. This problem has been examined by many people and the exact solution is known.
We ran the program a number of times and got a few results within a percent or two of one another.

fig.7

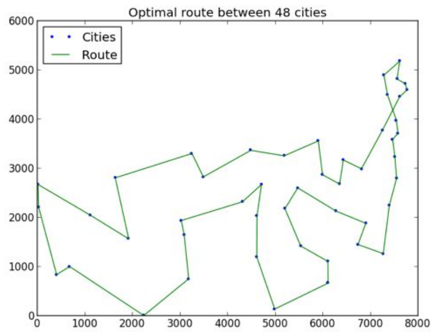

fig.8

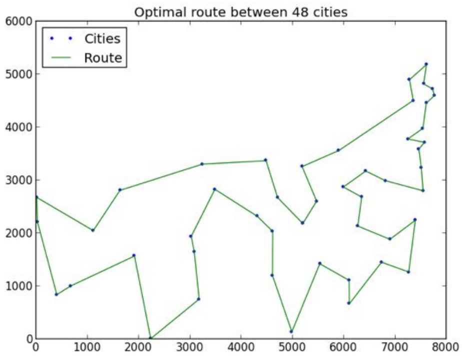

fig.9

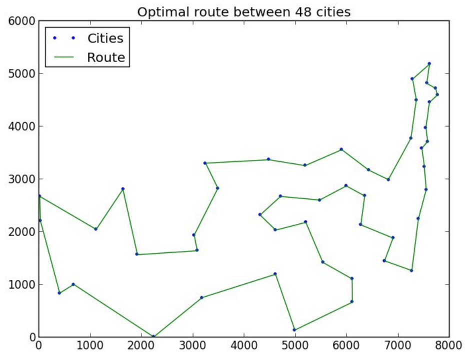

fig.10

	Fig.8	Fig.9	Fig.10
Route length	34322	35039	33523

Fig.7 is simply a map of the states for reference. We can see that fig.10 is the shortest and this is in fact the optimum solution. While it is clear that fig.4 is shorter than fig.8 it is very interesting that there is no easy way to tell if fig.9 or fig.10 is better without measuring them.

Conclusion

We saw that for a relatively low number of cities, 25, that our algorithm can very quickly find a route that is close to if not the optimal route. It found these solutions only using around 60,000 operations which is far then the number it would take to brute force the 25!/2 possible solutions.

Our estimation for 100 cities also compares well to the analytical estimation.
We managed to find the exact solution to the 48 states tour. However, we saw that our algorithm often gave us results that were up to 5% greater than that of the exact solution.
I think overall the generalised Ising model and simulated annealing is a good algorithm to solve the TSP. However, when n got large (=1000) we found that this algorithm was not sufficient to find a solution in an acceptable time frame.

References

- Urban operations research chapter 6.4.8
- Distribution Management: Mathematical Modelling and Practical Analysis by Eilon and Samuel