How to implement a dynamic programming solution to the 2D bitonic euclidean traveling salesman problem?

I understand that a bitonic tour crosses all vertices with one monotonic path traveling from the left most point to the right most point, then monotonically from the left most point to the right most point, with no self-crossing. I have read (Introduction to Algorithms, 3rd Edition by Cormen et al.), that a solution can be obtained by dynamic programming (solving subproblems, and saving their solution for later solutions of larger subproblems). I have read several solutions online, but I still can't understand how to characterize the subproblems... and build the solution in a bottom-up fashion. Pictures would be nice, or an example.

graph-theory

1 Answer

Say, we are given n points $x_1, \dots, x_n$ in the plane. Let's assume them to be ordered by their x-coordinates. I.e. x_1 is leftmost and x_n is rightmost.

Let's think a bit about the minimal bitonic tour on all vertices. We can be sure that the edge (x_{n-1}, x_n) will be contained in such a tour, so it suffices to find the length of a minimal path going from x_n strictly to the left upto x_1 - leaving out x_{n-1} - and then from x_1 strictly to the right upto x_{n-1} (and this is the problem I will solve by dynamic programming). I'll call such a path a *bitonic path* from x_n to x_{n-1}. We observe the following:

- Any bitonic path from x_n to x_{n-1} must start with a first edge (x_n, x_k) for some $k < n - 1$.
- Since we must visit all points, and since - from x_k - we can only continue to the left, all of the points $x_{k+1},\ x_{k+2},\ \dots,\ x_{n-1}$ must necessarily be visited on the way from left to right (and in this order). So necessarily, our bitonic path ends with $x_{k+1} \to x_{k+2} \to \cdots \to x_{n-1}$.

What have we figured out so far? A bitonic path from x_n to x_{n-1} has the form

$$x_n \to x_k \to \quad ??? \quad \to x_{k+1} \to x_{k+2} \to \cdots \to x_{n-1}$$

where $x_k \to ??? \to x_{k+1}$ is itself a bitonic path from x_k to x_{k+1} with $k < n - 1$.

If we want to minimize the length of the whole bitonic path from x_n to x_{n-1}, we must also minimize the length of the bitonic path from x_k to x_{k+1} (which is the same as the length of a bitonic path from x_{k+1} to x_k).

For any $i > 1$, let $\ell(i)$ denote the length of a length-minimizing bitonic path from x_i to x_{i-1}. The above tells us that

$$\ell(n) = d(x_n, x_k) + \ell(k+1) + \sum_{m=k+1}^{n-2} d(x_m, x_{m+1})$$

for some $k < n - 1$. So we must have

$$\ell(n) = \min_{1<i<n} \left[d(x_n, x_{i-1}) + \ell(i) + \sum_{m=i}^{n-2} d(x_m, x_{m+1}) \right]$$

But the exact same reasoning on such a bitonic path applies also for $p < n$. I.e. we have obtained the recursion

$$\ell(p) = \min_{1<i<p} \left[d(x_p, x_{i-1}) + \ell(i) + \sum_{m=i}^{p-2} d(x_m, x_{m+1}) \right]$$

with $\ell(2) = d(x_2, x_1)$. We can use this recursion to successively calculate $\ell(p)$ for $p = 2, \dots, n$, and the length of a bitonic tour on all points $x_1, \dots, x_n$ then calculates as

$$\ell(n) + d(x_{n-1}, x_n)$$

answered Jan 18 '12 at 11:31

Sam
186 3

What about time complexity analysis? :) – Mantas Dec 10 '13 at 23:32