PTAS for 2-Dimensional Euclidean TSP

Vijay Kothari

Rutgers University, Camden

February 6, 2010

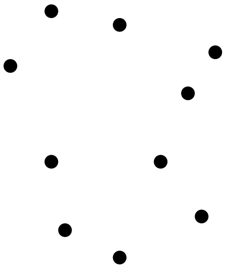
TSP Statement:

Input:

- Set $S = \{(x_1, y_1), \ldots (x_n, y_n)\}$ of points in plane
- Distance function d defined on points in S

Output:

- Least cost traveling salesman tour.



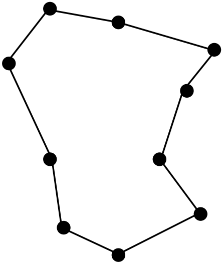
TSP Statement:

Input:

- Set $S = \{(x_1, y_1), \ldots (x_n, y_n)\}$ of points in plane
- Distance function d defined on points in S

Output:

- Least cost traveling salesman tour.



How hard is the TSP?

- The TSP, metric TSP, and Euclidean TSP are all NP-hard.
- If P$\neq$NP there does not exist a PTAS for the metric TSP.
- There exists a PTAS for Euclidean TSP.

Euclidean TSP

The PTAS involves 3 steps:

- **Perturbation:** Transform the given instance of the problem into a “nice" instance.
- **Shifted Dissection:** Randomly shift and dissect a bounding box of the “nice” instance.
- **Dynamic Program:** Employ dynamic programing on the shifted dissection.

Nice Instances

A Euclidean TSP instance is considered nice if:

1. All points have nonnegative integer coordinates
2. The minimum nonzero distance between points is at least 2
3. The maximum distance between points is $O(n)$
4. All points (x_i, y_i) satisfy $x_i, y_i \in [0, O(n)]$

Perturbation Lemma

Lemma (Perturbation Lemma)

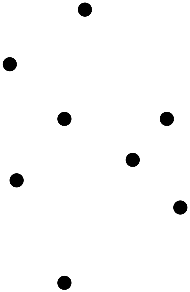
A PTAS for nice Euclidean TSP instances admits a PTAS for all Euclidean TSP instances

Proof of Perturbation Lemma

- *OPT* : cost of an optimal tour.
- ϵ' : a positive constant.
- ϵ : a positive constant we will fix later.
- We will show how to obtain a tour of cost at most $(1 + \epsilon')OPT$ if there exists a PTAS for nice instances.

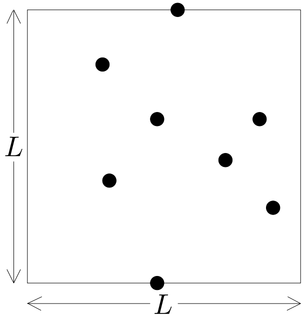
Proof of Perturbation Lemma

- $L = \max\{\max_i\{x_i\} - \min_i\{x_i\}, \max_i\{y_i\} - \min_i\{y_i\}\}$.
- Construct a bounding box with sides of length L. Note $L \leq OPT$.



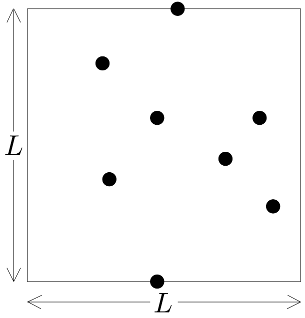
Proof of Perturbation Lemma

- $L = \max\{\max_i\{x_i\} - \min_i\{x_i\}, \max_i\{y_i\} - \min_i\{y_i\}\}$.
- Construct a bounding box with sides of length L. Note $L \leq OPT$.



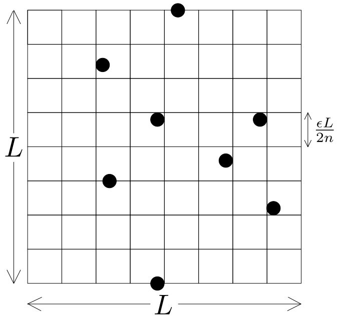
Proof of Perturbation Lemma

- Place grid of granularity $\frac{\epsilon L}{2n}$ on bounding box.
- Move each point of TSP instance to nearest intersection of the grid.
- An optimal tour for the original instance of the problem has cost at most $(1+\epsilon)OPT$ on the perturbed instance.



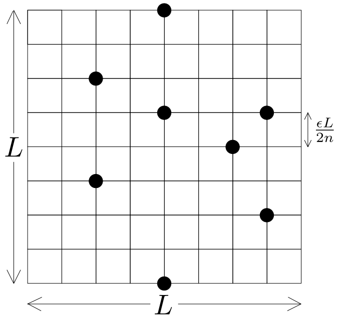
Proof of Perturbation Lemma

- Place grid of granularity $\frac{\epsilon L}{2n}$ on bounding box.
- Move each point of TSP instance to nearest intersection of the grid.
- An optimal tour for the original instance of the problem has cost at most $(1+\epsilon)OPT$ on the perturbed instance.



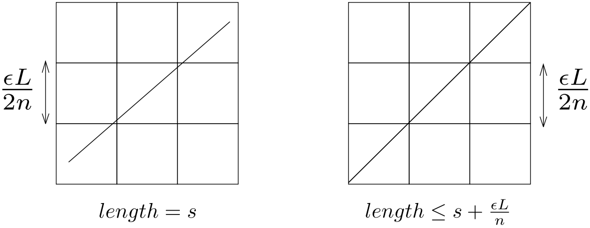
Proof of Perturbation Lemma

- Place grid of granularity $\frac{\epsilon L}{2n}$ on bounding box.
- Move each point of TSP instance to nearest intersection of the grid.
- An optimal tour for the original instance of the problem has cost at most $(1 + \epsilon)OPT$ on the perturbed instance.



Proof of Perturbation Lemma

- Place grid of granularity $\frac{\epsilon L}{2n}$ on bounding box.
- Move each point of TSP instance to nearest intersection of the grid.
- An optimal tour for the original instance of the problem has cost at most $(1 + \epsilon)OPT$ on the perturbed instance.



Proof of Perturbation Lemma

- Scale distances by a factor of $\frac{4n}{\epsilon L}$ so that the new granularity is $\frac{4n}{\epsilon L} \cdot \frac{\epsilon L}{2n} = 2$, the minimum nonzero distance between points is 2, and the maximum distance between points is $O(n)$.
- Translate the bounding box so that the lower left-hand corner coincides with the origin.
- The resulting instance is nice.

Proof of Perturbation Lemma

- *OPT*: cost of optimal tour in original problem instance.
- *OPT′*: cost of optimal tour in nice instance
- *C′*: cost of tour obtained by PTAS in nice instance for parameter ϵ
- *C*: cost of tour obtained by PTAS in original instance.

Proof of Perturbation Lemma

$$C \leq \frac{\epsilon L}{4n} C' + \epsilon L \leq (1+\epsilon)\frac{\epsilon L}{4n} OPT' + \epsilon L \leq (1+\epsilon)(OPT + \epsilon L) + \epsilon L$$

$$C \leq (1 + 3\epsilon + \epsilon^2) OPT$$

Proof of Perturbation Lemma

If we set $\epsilon = \frac{\epsilon'}{4}$ then:

$$C \leq (1 + 3\epsilon + \epsilon^2)OPT \leq (1 + 4\epsilon)OPT = (1 + \epsilon')OPT$$

This completes the proof of the lemma.

Shifted Dissection

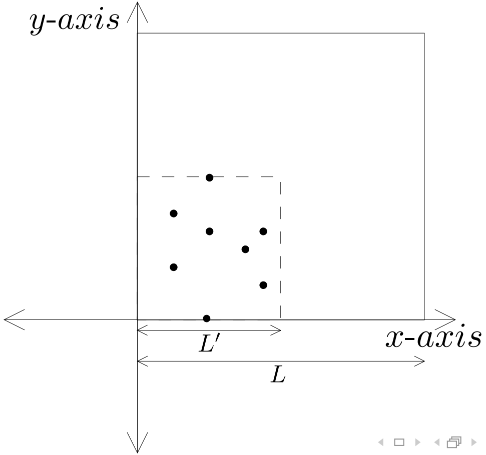
- We now have a nice instance of the TSP.
- The next step is to obtain a new bounding box, translate it using randomization, and dissect it.

The New Bounding Box

- L' : length of the smallest bounding box enclosing all points.
- L : smallest power of 2 that is at least twice as large as L'.
- $L \in O(n)$.

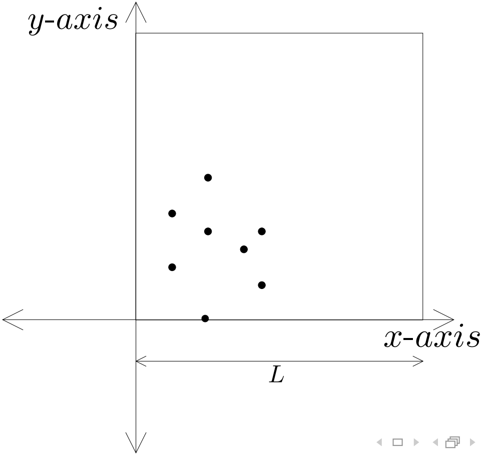
The New Bounding Box

- Place a bounding box of length L at the origin.
- Each point now lies in $[0, L/2] \times [0, L/2]$.



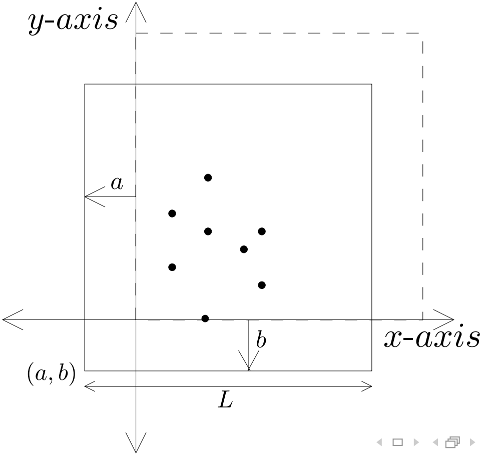
The New Bounding Box

- Place a bounding box of length L at the origin.
- Each point now lies in $[0, L/2] \times [0, L/2]$.



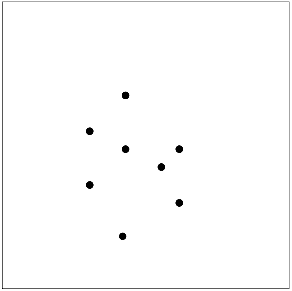
The Shift

- Choose integers a and b from $(-L/2, 0]$.
- Shift bounding box so lower left-hand corner is located at $[a, b]$.



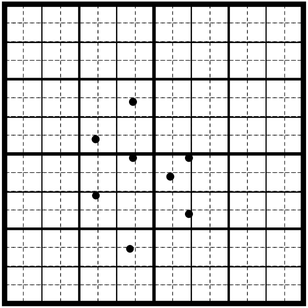
The Shift

- Choose integers a and b from $(-L/2, 0]$.
- Shift bounding box so lower left-hand corner is located at $[a, b]$.



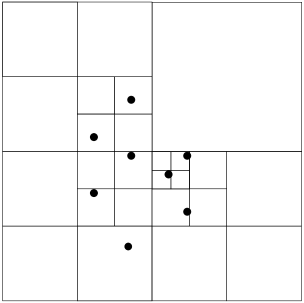
The Dissection

- The dissection is performed by recursively partitioning each square of side length at least one into smaller squares.
- The level of a square is its depth in the 4-ary tree. The bounding box has level 0, the children of the bounding box have level 0, and so on. The last level is $O(\log n)$ since $L \in O(n)$.



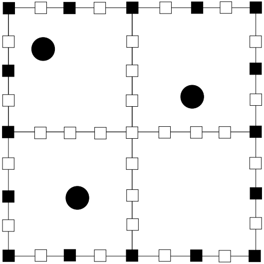
The Quadtree

- The quadtree is similar to the dissection, but we stop partitioning when a square contains at most one node.
- The quadtree has depth $O(\log n)$ and $O(n \log n)$ nodes.



Portals

- In the DP we only consider tours that enter and exit squares on prespecified points called portals.
- On the perimeter of each square we place 4m portals: one portal on each corner and m - 1 equally spaced portals between corners on each side.
- We enforce that the portal parameter m is a power of 2.



Definitions

- p-tour: is a tour that optionally includes portals.
- portal-respecting tour: a p-tour that only enters and exits squares through portals.
- r-light tour: a p-tour that crosses each side of each square in the dissection at most r times.
- partial p-tour for a square: the part of the p-tour that lies inside the square.

DP Objective and Multipath Problem

- The dynamic program solves the (m,r)-multipath problem.
- In the multipath problem we are given :
  - a particular square in the quadtree
  - a multiset of portals on this square containing at most r portals per side
  - a matching of portals into entry and exit pairs
- The goal is to find a minimum cost tour that visits every point in the square, connects every entry point to its corresponding exit point, and uses each of the designated portals exactly once.

Dynamic Program

- The cheapest r-light portal respecting p-tour is then given by the entry in the level 0 square that does not use any of the portals on the boundary of its square.
- The total number of entries in the lookup table is at most: $O(n \log n) \cdot (4m + 1)^{4r} \cdot (4r)! = O(m^{O(r)} n \log n)$

Dynamic Program

- For any dynamic program entry associated with a leaf of the quadtree we can generate a solution for that entry in $O(r)$ time.

Dynamic Program

- Other entries are characterized by a square S containing 4 smaller squares $s_1, s_2, s_3, and\ s_4$.
- For these entries, we may use at most r portals along each of the internal edges. This contributes $O((m+2)^{4r})$.
- There are $2r$ pairings of the portals on external edges into entry and exit pairs so that there are $O((2r)^{4r})$ ways to place the internal portals between entry and exit pairs of external portals.
- There are $O((4r)!)$ ways to order the portals on the path.

Dynamic Program

- The net running time for a particular instance is then: $O((m+2)^{4r}) \cdot O((2r)^{4r}) \cdot O((4r)!) = O(m^{O(r)})$
- Since the number of entries is $O(n \log n)$, the net running time for constructing the entry table is $O(m^{O(r)} n \log n)$

Overview of Analysis

Now, we must show the following:

- It's not too expensive to convert an optimal tour into a portal respecting tour.
- It's not too expensive to convert a portal respecting tour into an r-light portal respecting tour.

Crossing Lemma

We bound the number of segment crossings on vertical and horizontal lines of our dissection.

- ℓ : vertical or horizontal line in the dissection.
- $t(\ell)$: number of times that the optimal tour crosses line ℓ.
- T : total number of times that the optimal solution crosses a vertical or horizontal line of the dissection.
- $T = \sum_{\ell} t(\ell)$.

Crossing Lemma

Lemma (Crossing Lemma)

For optimal solutions to nice Euclidean instances, $T < 3OPT$.

Proof.

Consider an edge S of length s connecting two nodes (x_0, y_0) and (x_1, y_1) in the optimal tour. Note $s \geq 2$. Let $\tau(S)$ denote the number of lines S crosses. We have:

$$\tau(S) \leq |x_1 - x_0| + |y_1 - y_0| + 2 < 2s + 2 \leq 3s$$

So, $T = \sum_{\ell} t(\ell) = \sum_{S} \tau(S) < \sum_{S} 3s = 3OPT$. □

Portal Respecting Lemma

Lemma (Portal Respecting Lemma)

A solution that has a total of T crossings and a cost of OPT can be transformed into a portal respecting solution of cost $OPT + \frac{T \log L}{m}$ that maintains T crossings.

Proof of Portal Respecting Lemma

- Any time an edge in the tour crosses the side of a square, but not at a portal, we will bend the edge so that it crosses the nearest portal on the maximal level square it touches.
- It will then only cross sides of squares that are lower in the quadtree at portals as well.
- The probability that a given crossing occurs on line i is $p_i \leq \frac{2^i}{L}$
- If the maximal level square is i, the cost of the tour is incremented by $c_i = \frac{L}{2^i m}$.

Proof of Portal Respecting Lemma

The expected cost of moving crossings to nearest portals is then:

$$\sum_{\ell} \tau(\ell) \sum_{i:i \geq 1} p_i c_i = \sum_{\ell} \tau(\ell) \sum_{i:i \geq 1} \left(\frac{2^i}{L}\right)\left(\frac{L}{2^i m}\right)$$
$$= \sum_{\ell} \frac{\tau(\ell)}{m} \log L = \frac{T \log L}{m}$$

Note that this is useful since $T \leq 3OPT$ by the Crossing Lemma.

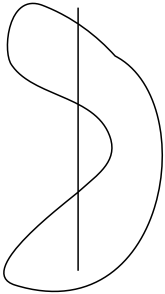
Patching Lemma

Lemma (Patching Lemma)

Given a line segment S of length s, if a closed path C crosses S at least thrice, we can add line segments on S of total length not exceeding $6s$ to the tour to get a new closed path C' that crosses S at most twice and contains the previous tour.

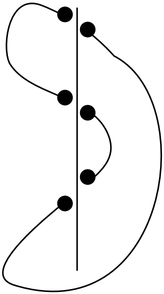
Proof of Patching Lemma

- Suppose C crosses vertical line segment S a total of $k \geq 3$ times.



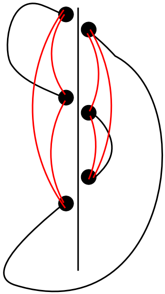
Proof of Patching Lemma

- Break the tour at the k points at which C crosses S.
- Add two copies of each point corresponding to a crossing: one corresponding to the left side of S and the other corresponding to the right.



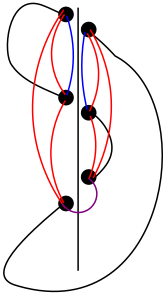
Proof of Patching Lemma

- Find a minimum cost salesman tour τ_L on L and a minimum cost salesman tour τ_R on R.



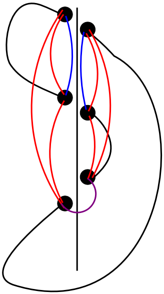
Proof of Patching Lemma

- If k is odd, match the first $k - 1$ points of each side, separately, and connect the last pair of points.
- If k is even, match the first $k - 2$ points on each side, separately, and connect the last two pairs of points.



Proof of Patching Lemma

- The resulting graph is Eulerian, is connected, contains all points of the previous tour, and crosses S at most twice.
- Each of the two added tours has cost at most $2s$, and each matching has cost at most s, for a total added cost of $6s$.



Patching an Optimal Solution

- The basic idea is to show that we can transform an optimal solution into an r-light portal-respecting salesman tour without paying too much.
- We repeatedly apply the Patching Lemma to the optimal solution to obtain an r-light tour.
- Then, we apply the Portal Respecting Lemma to the resulting tour obtain an r-light portal-respecting salesman tour.

Structure Theorem

Theorem (Structure Theorem)

If integers a and b are picked uniformly at random from the interval (-L/2, 0], then with probably at least 1/2, the (a, b)-shifted dissection has an r-light portal-respecting p-tour of cost at most (1 + ϵ)OPT for $m = O(\frac{1}{\epsilon} logL)$ and $r = O(\frac{1}{\epsilon})$. Moreover, this r-light portal-respecting p-tour can be found by the dynamic program in $O(n \log^{O(1/\epsilon)} n)$ time.

Proof of Structure Theorem

- Consider a line ℓ. Let i_ℓ be the maximal level (level closest to root) of ℓ. That is, let i_ℓ be the level containing a square that has a side coinciding with line ℓ, but if $j < i_\ell$ then ℓ does not coincide with the side of any square on level j.
- To patch an optimal solution of cost *OPT* we apply the Patching Lemma in a bottom up fashion. For each line ℓ and its corresponding i_ℓ value we apply the Patching Lemma to all sides of squares contained in ℓ from level $j = \log L$ down to $j = i_\ell$.

Proof of Structure Theorem

- Fix a particular line ℓ. Suppose we apply the Patching lemma k times to sides of squares contained in ℓ that have more than r crossings. Then, $(r+1)k - 2k \leq \tau(\ell)$.
- Let n_j denote the number of times the Patching Lemma is applied at level j. We have $\sum_{j \geq 1} n_j \leq \frac{\tau(\ell)}{(r-1)}$.
- The probability that a vertical line ℓ has maximal level i is $p_i = \frac{2^i}{L}$.
- The cost of applying the Patching Lemma at level j is $c_j = \frac{6L}{2^j}$.

Proof of Structure Theorem

The expected cost of applying the Patching Lemma in the bottom up fashion for a particular line ℓ is then :

$$\sum_{i:i\geq 1} p_i \sum_{j:j\geq i} n_j c_j \leq \sum_{i:i\geq 1} \frac{2^i}{L} \sum_{j:j\geq i} n_j \frac{6L}{2^j} \leq 6 \sum_{j:j\geq 1} \frac{n_j}{2^j} \sum_{i:i\leq j} 2^i$$
$$< 12 \sum_{j:j\geq 1} n_j \leq 12 \frac{\tau(\ell)}{(r-1)}$$

Thus, the increase in cost is at most $12\frac{T}{(r-1)} \leq 36\frac{OPT}{(r-1)}$ by the Crossing Lemma.

Proof of Structure Theorem

- Next, we apply the Portal Respecting Lemma to make the tour m-respecting.
- This adds an expected cost of at most $3\frac{OPT \log L}{m}$ to the tour.

Proof of Structure Theorem

- The expected added cost in making the tour r-light and m-respecting is at most $36\frac{OPT}{(r-1)} + 3\frac{OPT\log L}{m}$.
- If $m = r\log L$ the expected added cost is at most $39\frac{OPT}{(r-1)}$.
- If $m = r\log L$ and $r = \frac{78}{\epsilon} + 1$ the expected added cost is at most $\frac{\epsilon}{2}OPT$.
- Markov's Inequality tells us that the cost will be at most ϵOPT with probability at least $\frac{1}{2}$.