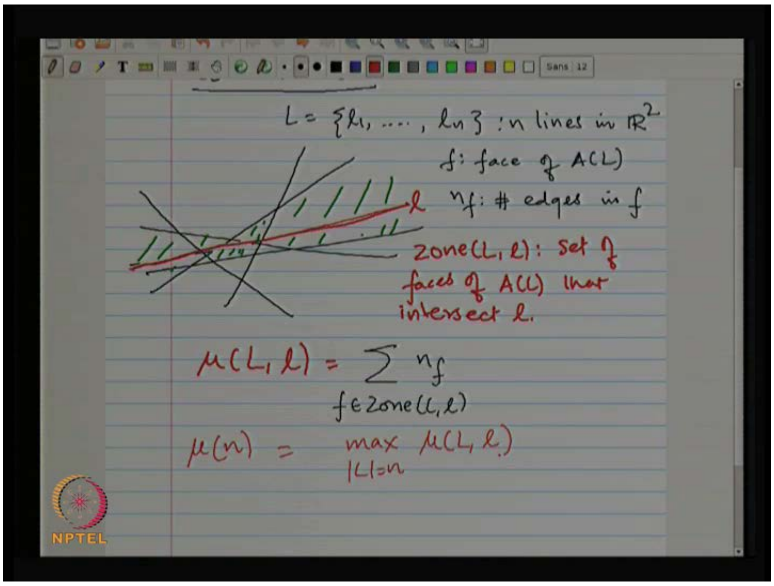
Computational Geometry
Prof. Pankaj Aggarwal
Department of Computer Science and Engineering
Indian Institute of Technology, Delhi

Module No. # 10
Arrangements and Levels
Lecture No. # 2
Arrangements (Contd.)

So, today I want to begin by finishing the proof of the Zone Theorem and then we will discuss some of the applications of the Zone Theorem and arrangements.

(Refer Slide Time: 00:34)



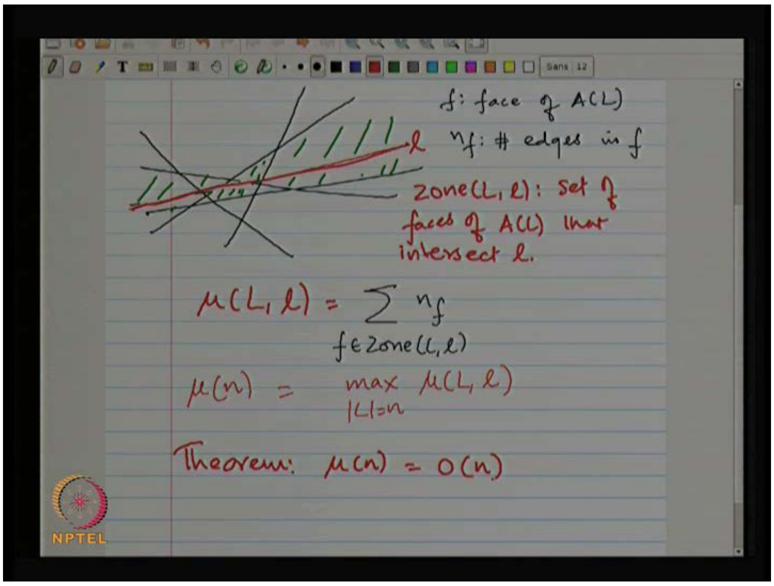
Let me just go over the notations once again. So, L is a set of n lines and I will use A L to denote the arrangement, which is a planar sub division induced by the set of lines. For each face f, two-dimensional face f, I will use n f to denote the number of edges; it does not matter I choose edges or vertices, because that are of the same. If the face is bounded, two quantities are same; if it is unbounded, it will be differ by one. Now, if I take another line…

(No audio from 01:08 to 01:24)

Zone is a set of faces of the arrangement that intersect L. So, just to remind you, zone L is a set of; (no audio from 01:44 to 01:54)

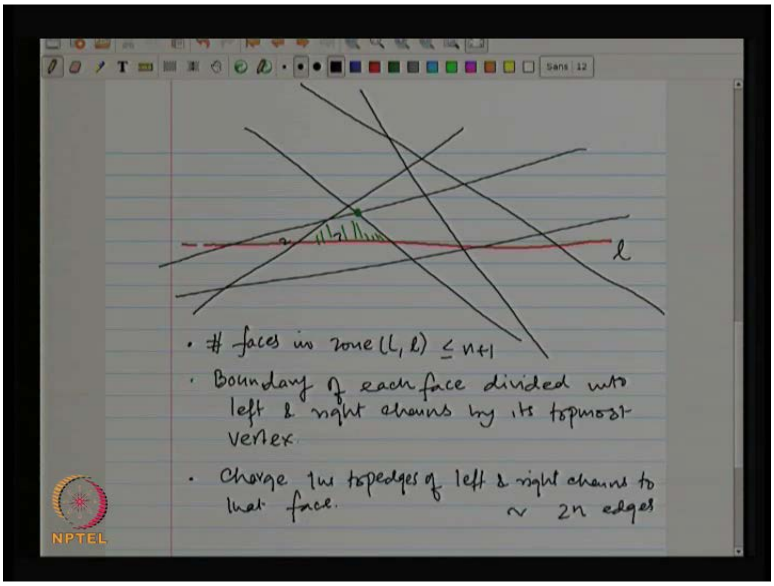
So, in this picture, (no audio from 01:58 to 02:07) those dashed faces are the faces in the zone. And what I have defined was that is a basically, that is a number of edges in the faces that lie in the zone and I denote the quantity, mu n is a maximum, this quantity over all lines.

(Refer Slide Time: 03:03)



And the theorem that I wrote was that mu n is order of n. So, this is what I want to prove today. And once I have done this, that will imply that algorithm, I have described for constructing the arrangement, takes n square times, because inserting each line takes linear time; then the whole will thing take n square times.

(Refer Slide Time: 03:36)



(No audio from 03:36 to 03:55)

Suppose, this is the arrangement, the zone; what I will do is, first I will count the number of edges in the zone that lie. So, this is the line l, whose zone I want to count. First, but I will do that I will count the number of lines that lie above, edges in the zone that lie above the line and by multiplying it by 2, I will also count the number of edges that lie below that line in the zone. I will do some double counting, because edges that intersect the line, they will be been counted twice and I will compensate for that, but it is not going to be important.

So, first observation is; first thing observe question, how many faces does this line l intersect? How many faces are in the zone?

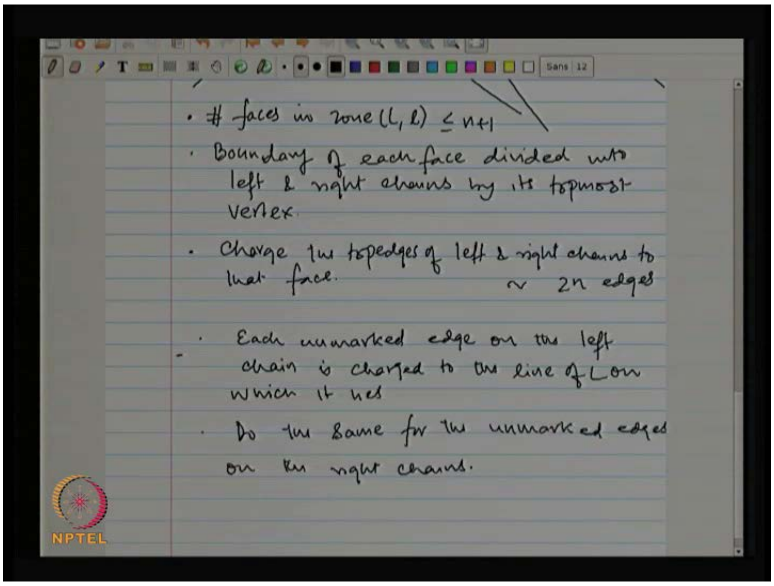
(( ))

Maximum n. Am I right? Because or n plus 1, because each line intersects the red line only once and if you so, you get n plus 1 intervals and each interval corresponds to one of the faces that is in the zone. So, this is the face 1, 2, 3 and so on. So, the number of faces in zone L is at most n plus 1. So, let us look at one of the faces and look at the portion that lies above it. So, I wanted to draw a complicated face, which I will do; so, let us look at this face. There is a top most vertex; so, this top most vertex, if you look at portion of the face that lies above the line, the top most vertex divides this boundary of this face into two parts. What I mean is the following. Here, is the line l, whose zone your looking at and let us look at a complicated face. This is the top most vertex. So, there is a left chain and the right chain; this is the vertex. So, here is the one set of edges, and here is the another set of edges. Let us call this as a left chain and let us call this as a right chain.

Now the topmost edge in each chain, I will charge it to that face itself. So, let me write it down. (No audio from 06:59 to 07:37) And charge the top edge of left and right top edges of chains to that face. So, what you are doing is, for each face your charging two of its edges to that face and I know there are only n plus one such phases. So, this is a total of; ignore 1 for now, because I do not want to be quirky. So, this is roughly 2 n edges, right.

So, what I have done is, this I have charged, this I have charged. What I have to now count is, these edges; these I have not accounted for. And here is the nice claim. Let us assume that these edges I have marked, because I have charged them, so I marked them. So, this I have marked; these are unmarked edges.

(Refer Slide Time: 09:10)



So, the claim is, each line; let me put this way, let us focus on the left chain and the right chain, I will do in the same way. Am I right? Let us focus on the left chains. And what I will do is the following; let me say it and then I will say the claim. So, you are looking at this edge and this edge and let us label this edge lines l 1, l 2, l 3, l 4.

So, let me sort of repeat myself again; what I did was, I took each face on the zone, took the topmost vertex, the topmost vertex divides its boundary of the phase into two chains, left chain and the right chain. The top boundary of the left chain and the top boundary edge of right chain, I charged to that face to that vertex. Now, the rest of them; let us look at the left chain. What I will do is, each unmarked edge in the left chain on any face, I will charge it to the line itself.

So, before write the, let me finish the charging scheme. Each unmarked edge on the left chain is charged to the line of L on which it lies and do the same for the… Is the charging scheme clear? Before I give the crucial; before I give the punch line, so this is not just one particular face; this I do for the all the faces that lie above the portion of fixed line above the line.

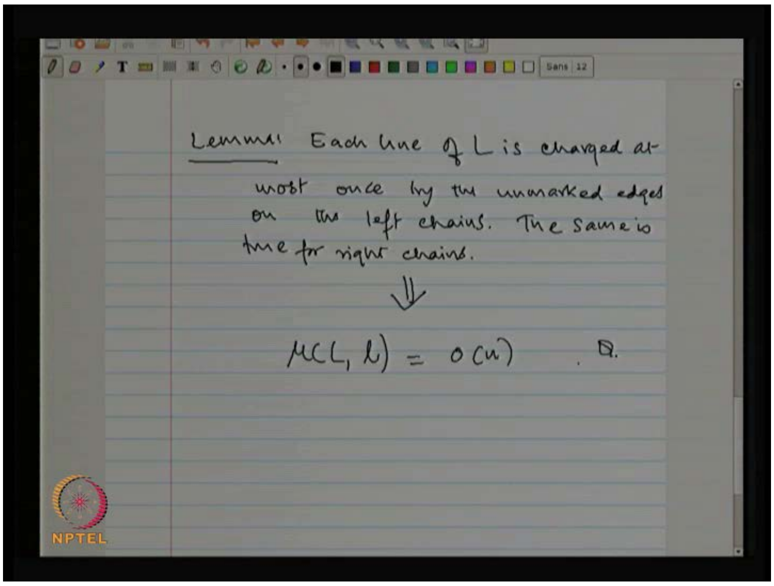
So, for each face, the two edges are charged to the that face, that gives you 2 n and the rest of the faces which have edges that have not been charged, I charge each edge; I charge to the line itself. And the claim is the following

Yeah

(( )) Why did we read the

You will see; because wait for the punched line.

(Refer Slide Time: 12:26)



Here is the Lemma. Each line of L is charged at most once by the unmarked edges on the left chains. So, this is the punched line and the same is true for the right chains; and the similar claim is true for right chains and the same is true.

Let us ponder a minute before we prove this lemma. If this is true, what this says is that each line is charged at most once for left chain and at most once for the right chain; it means that I have the total charge I made is only 2 n. So, 2 n such edges and 2 n the edges that I charged to the faces, you get it 4 n. And then this is only the portion above the line, then we repeat the same thing below the line, you get a time. So, this measure is the proof for the Zone Theorem.

So, now there is some double counting going on and if you want to improve the constant, you can be more careful; the argument that I am giving you is sloppy. But this shows that the total complexity of the zone is at most a time. So, let me repeat it, because what I did was, first I focus a portion that is above the z line and for each face on the zone, I charge two edges on the top left and top right to the face; rest I charge to the line that which they belong to. And then I claimed that each line is being charged at most twice; once by the left and once by the right. So, that basically switch and repeat the same argument for the below, it gives you total (( )).

So, the main thing that is left now is to prove this lemma; that why each line is being charged only once and that will also answer his question, why I am doing in this way? And it is again is easier to do the proof by picture.

Suppose, this was; let us say an edge or that was unmarked because let us say this was some other face it look like and this was not the top edge because top edge something like this, some like here. So, this is an unmarked edge; someone to complete it actually, I have drawn it slightly smaller. If you, in this case it goes, let us say that this is another somehow that is how it look like; so, this is an edge that is.

Thus, any one see that why a line, this line l, if it is been charged by this edge e, it will never be charged by another edge. That is why. So, what happens is that once this line intersects, suppose it is charged here, then what happens after that is that since this was not a top edge, right; this a top edge. It means that line l is being intersect by l prime and after this l prime lies below it; l. Am I right?

After this intersection point, line l prime lies below l and what is happening is that l prime is shielding l from appearing on the zone again. Because, there is no way that after this one that l can show up on the this line can show up on the zone after that again, because it is always being hidden by this line.

Because if you notice that if it appears, an edge appears in the zone, you can connect it by straight line to this line without intersecting any other line. But now if you take any other point here, when you try to connect it to the red line, you will always intersect to this l prime. But this argument crucially uses the fact that this was not the top edge; you see that if it was this edge, for this edge I could not make this argument because there is no edge which is lying below it, to the right of it here. And that is why I did this charging scheme in this way. So, this finishes the proof.

So, if you wanted to write, say it more formally, then what you have to say is that after since this is not the top edge, this is another edge l prime that lies that intersects, appears in the zone after it. And this intersection point after this face, one always lies above l prime. As a result, one cannot show up on the zone to the right of this intersection point; it means to the right of this face. As a result, this is being charged only once and the same thing happens to the right edge, right chain. Then, you have to talk about the left part and that completes the argument.

So, this shows that the Zone Theorem is true; this proves that; this implies that mu and that finishes the algorithm.

Is this clear? Or any questions?

(( )) Sorry, say it again.

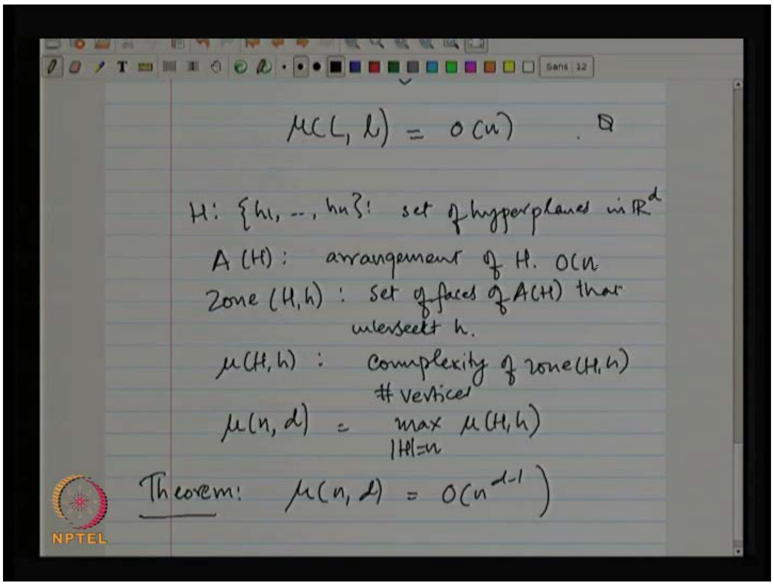
The part of the edge to the left of this page (( )) can you get by the figure

This figure? So, far from the left, ask for first time in this left, when this line appears on this zone, right. Ask a question, when is the first time? So, I am talking about the left chains. Am I right? So, ask a question, when does this line being charged on the left chain first time? And let us say this is the first time this happen, it has never happened before that; and because after what I am argue that once it has happened, it cannot happen again.

Does it make sense or any?

Now, so far, I talked about the arrangements of lines in 2D. One can talk about the similarly, arrangement of hyper planes, let us say in three dimensions you have a set of planes. If you take a set of planes and draw in d three dimensions, you will get another arrangement, but it will not be a planar subdivision; it will be a spatial subdivision. And you can still talk about the notion of the zone, that you take another plane and ask about some complexity of the faces that intersects this new plane and you can incrementally again construct the algorithm.

(Refer Slide Time: 20:16)

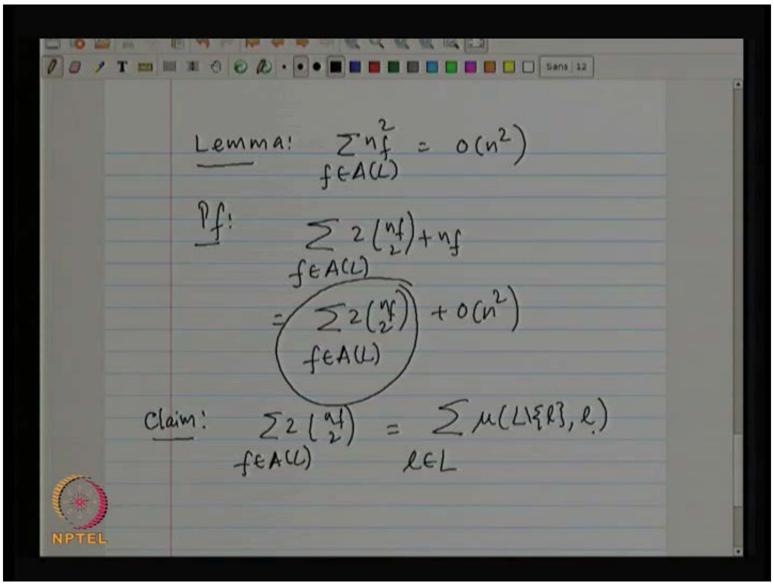


And so, you can talk about, let us say H is d dimensions and you can talk about the arrangement and then we can also talk about a zone of set of and again you can talk about the complexity of number of vertices or some convenient way of talking about.

And now, since you are talking about d dimensions, let me use two parameters, n and d to denote what dimensions we are. And the theorem is which I will not prove, it is a. Now, I should say that the complexity of the arrangement is order of n to power d because every d tuple, this is the way in two dimensions, every pair of lines intersect at a single point. In 3 dimensions, if you take 3 planes, intersection of three planes is the single point. So, the complexity of arrangement in 3D is nq. And if you have d dimensions, if you take hyperplanes in d dimensions, every d of them will intersect at a single point. So, number of vertices in d dimensions will be n to power d. So, the complexity of arrangement in d dimensions is n to power d and what is known is that the complexity of the zone in d dimensions is only n to the power d minus 1.

So, this incremental algorithm can be extended to higher dimensions; you add a hyperplane one after the another and construct the arrangement. But the proof is much more complex and it is a recursive argument and I will not go through it. If you are interested, I can give you the reference.

(Refer Slide Time: 23:01)



Now, let me do the proof the last time I did, again and then I move on. So, last time I had proved and that is what I had proved last last time. And the proof was as follows: First, some simple algebra, right. So, if you want to write x squared, something you can write it as 2 times x choose to plus x. Second term is, we know that it sums to n squared because it is a total complexity of the arrangement; number of edges over (( )) in each face. (( ))that is left is to bound this quantity.

And the claim is, so what it sort of says is that 2 times n f choose to is nothing but that do the following; we have a set of lines, remove one of the lines, repeat the following experiment for each line, pull out one of the lines, look at the arrangement of the remaining lines and think of the zone of this line in this arrangement. And, look at the complexity and you sum it, do it for each line one after another and that sum is the same as this one.

And why is that? Let us say, just to keep the life simple; may be, let me just keep just 4 lines, just to so that you can sort of follow.

So, n f to twice and f choose to is for every face. So, let us say fix one of the faces and count in the number of pairs of edges, that n f choose to wins and twice is mean that I count each pair twice. So, I do care about the pairing ordering.

Now, in order to do that; suppose, look at all pairs, fix a line l, one of the lines. Look at the all the pairs that I am counting that involve this line. So, for here let us call this line, l; so, for this edge, I will count this one, this one, this one and this side on this side and this one and this one on the other side. Am I right? Because those are the pairs, this edge will involve. And similarly for here, it will correspond to some other pairs and so on. But let us focus on this one, on this line, on this face.

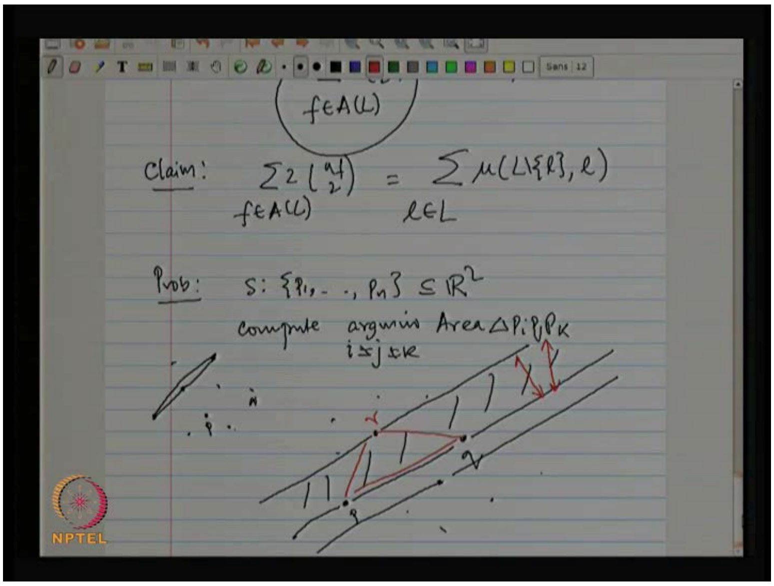
If I pull this line out and look at, it is a zone. You think about that I pull is red line out and I am looking in the zone of this red line and the arrangement of many things. And when I am counting the complexity of the zone, what am I counting? For each face that it intersects, I am counting how many edges are there and so, for this line, for this edge, in this face, this face it is intersecting and counting this one, this one, this one, this one, this one, this one, this one, this one and that is how I did the counting. So, what I did was; I do it for each such face.

So, what I did was I counted all the pairs that involve this line; then I counted the zone, here. So, all the pairs, since I am counting the pairs of edges that appear in the same face; now, if I fix a line and ask the question how many pairs are there which involve this line? That is nothing but the complexity of the zone of the claim, because edges where it is being are counted as a pair.

So, what it means is, when I do this for every line; now, if you fix a pair of lines, let us call this l prime, so, you have pair let us say, L l prime. Now, when I count notice that the way we wrote the sum, this pair is being counted twice; once, when I pull this line l out and I counted l with this pair, h prime and the second time, when I will pull this l prime out, and I counted l prime with l again. So, I counted each pair twice; so, that is why I have n f choose to.

Is that ok? So, last time I had asked the following question that I gave you a set of points and you want to find the minimum area triangle explained by triple of this points.

(Refer Slide Time: 28:32)



So, remember that I… The following problem I had asked, set of points in R squared; compute

(No audio from 28:46 to 29:06)

So, you want to compute a triple; so, whose area. So, that if look at the triangle, area is as small as possible.

So, for example, this might be the case, here. Now, as we talked at n cube is a trivial algorithm, the question is how we do n squared time?

What we will do is, we will use duality. Before that, we make the following observations, the following. Fix a pair of points, p q; so, let us fix a pair of points and let us look at the… ask the following question, which triangles I should look at, that involve the pair p q? So, the n minus 2 such a triangles, am I right? Because you fix two of them; that is n minus 2 points are left; with each of them, you get a triangle; out of this n minus 2, which one we should look at?

Now, since you are looking at the minimum area triangle, then the point that you should look at the points and you can sort them, how far they are the from this line. Am I right?

And, among them, what you have to do is, you can think about choosing only one one, which has a minimum height. Am I right? So, if you have all the other points, then you sort them by the line and see each of them you will only look at the closest one.

So, in particular, if this… in particular, what this implies is that if this is the… let us look at the do the both side; I will be little sloppy and I will look at the both of the (( )). So, if you look at this side, if I am going to look at…

(No audio from 31:09 to 31:17)

If I am going to consider p q r triangle, then say this is the closest one and this perpendicular distance, then the step that is I have drawn, that I drew the line parallel to p q passing through r, this state does not contain an any point in it. Am I right?

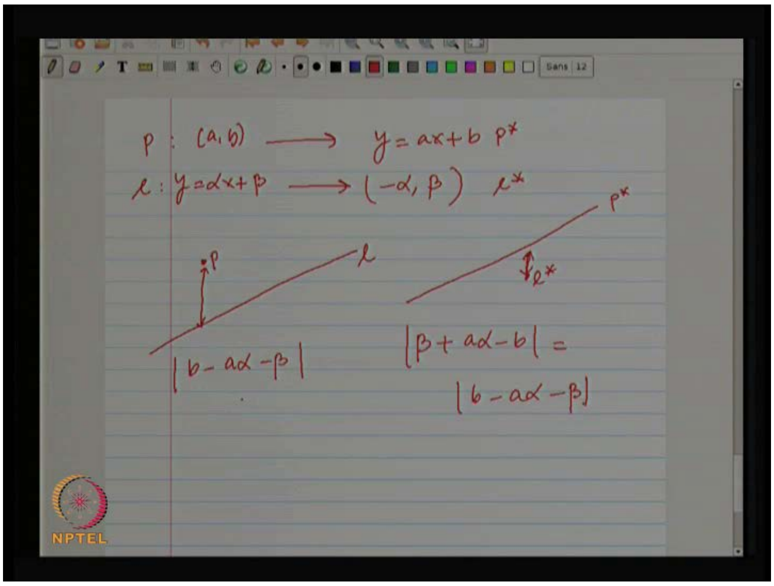
Other thing you observe, that says a snoop of p q is fixed; it is also fine. So, talking about the vertical perpendicular distance, we can also sort the points by the vertical distance with respect to this line. Your sorting sorted order will not change. Of course, the distance is changing, but the sorted order will remain the same, whether you do by vertical distance or you take the perpendicular distance; because, everything we are multiplying by the same sine; sine of theta, where theta is the slope.

So, why I am saying all these? Any questions?

You had? (( )) vertical distance is from what point?

From the line, from the line p q. See, you have this line p, passing through p q. Take any point, look at this vertical distance and sort by this. Now, comes to the duality and why the duality will play a role?

(Refer Slide Time: 33:05)

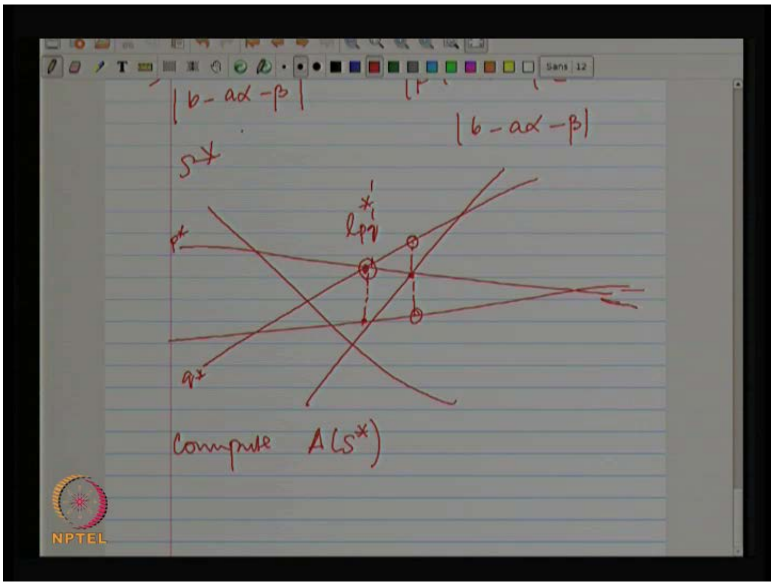


Remember that there are many forms of duality; and the one which I will is use the following that a point a b is mapped to a line v and the line mapped to the point; these are better. So, (no audio from 33:25 to 33:52) the nice thing about this duality is that the vertical distance (()); what I mean is that the vertical distance here is the same 12, certain not in the figure, but this vertical distance here; they are the same. And, it is not sort of this sort of different, because if you look at the vertical distance in the primal plane, it is basically d minus alpha a minus… The vertical distance between p and l is b minus a alpha minus beta and if you look at it here, again it becomes beta plus a alpha minus b, which is the same as b minus a alpha minus beta.

So, the duality preserves the vertical distance; point maps to a line; line maps to a point; the vertical distance it will point a primal. In the primal plane, vertical distance between point and line is the same as the vertical distance between line and the point. Now, what we are going to do is, we take the set of points, that were given to you, map them to lines; use the. This is duality.

(No audio from 35:18 to 35:31)

(Refer Slide Time: 35:19)



Now, what is the line passing through pair of points correspond to in the dual? Intersection of two points, am I right? Because if so you have a... So, you have the two points p and q, each of them maps to a line in the dual and the line in the primal passing through p and q, corresponds to the intersection point of the two lines.

So, let us say that this was let us say this was p star; this was q star. Then this intersection point of this one is… Now notice that, what you are interested in you are interested in a in finding a point, the third point which has a minimum vertical distance from this line; am I right? And the duality preserves the vertical line the vertical distance. So, what that means is, let from this intersection point, you want to find the line which has the minimum vertical distance.

So, if you draw the vertical distance from this side or this side, this is the point... Now, in this case, there is nothing (( )), but suppose if you had chosen this one, then what you are interested in these were these two lines, you wanted to look at; which is these two points that this point and this point that I drew here.

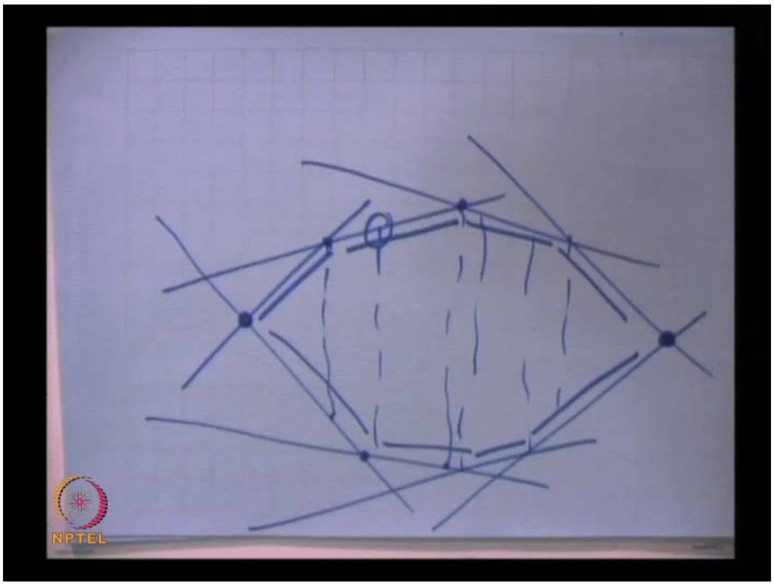
Now, if I have the arrangement, then what do you do?

(No audio from 37:12 to 37:24)

To which zone? Well, you do not have to do even that; it is simpler than this. So, what you want to do is, for each vertex, you want to find line that lies immediately below it and the line that lies immediately above it, right. That is what you need to find.

So, if you if I give you the entire arrangement, you have computed the entire arrangement, you can compute the arrangement in n squared time, right. Because compute; first step is compute, so this is S star, the arrangement of S star; compute the arrangement. Now, from the notice is that this vertex, the face this can on which we have to consider is, the edge that we are looking at is lies in the same face as this vertex lies. Am I right? So, what you have to do is, you have to traverse each face. So, you look at the one of the faces; you do the following for each face.

(Refer Slide Time: 38:28)



Suppose, you have one face; it has left and right vertex, leftmost vertex, rightmost vertex, you have top chain and you have a bottom chain. So, what you think about is you think about marching both on the top and the bottom simultaneously; think about like merge sort. Think about this is one list, this is the second list; you start going through this one and when you visit and you are marching, along a step, right; the way you do in the merge sort; so think about merging these two lists together; these two lists of vertices.

When you reach this vertex, figure out which edge you are currently scanning, match this one with this one; next time you will reach this vertex, then you see which edge you are scanning; this is this one, then again you will come, reach this vertex, then see which edges scanning here and so on.

So, just taking each face and scanning the… thinking of merging the top and the bottom boundary of this each edge and for each vertex, you can find what is the edge that lies below it? And the same thing you will do it for each face and that is why, you will have (()).

Traversing this is tough; takes time proportional to the number of vertices, here and you know the total complexity of arrangement is n squared; so, total time take it will take is n squared. So, this way you can find the… and you have all the triples you need. So, you can find the minimum area triangle in n squared time, and the million dollar question is can you do better? People believe that you cannot do better, but there is no lower bound.

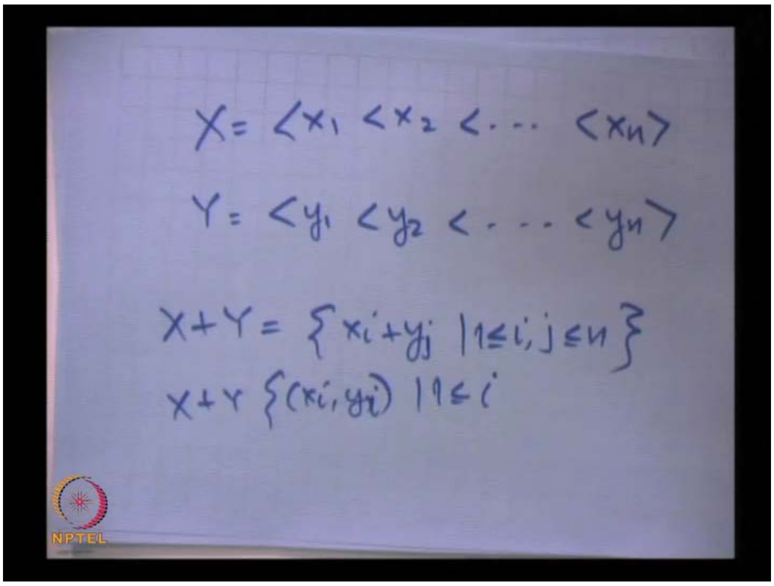
Actually, there is a even a simpler; let me ask you a simpler question; a special case of this is, as I give you set of points or three of them are collinear; any of three them lie on a line. This is simpler problem; how will you if I compute how will you figure that out?

Yeah, that is right. So, if there is a vertex which has degree more than 4; which has degree 6 or more; that is the case in 3 point; that is right. So, this even a simpler than this problem; somewhat simpler; am I right? Because that is the special case. And again, there is a n squared; you can do easily, compute the arrangement and then check it as you said, can you do better than n squared? There are some lower bound in some strange models, but these models are very strong; what specify, they says that if you take an arrangement, there are kind of n squared independent triangles.

And that sort basically people had, but in the traditional models that you saw, lower bound models that you see; like you have seen the decision tree model and some other cellpro model, it is not known; and that is the sort of an open question. And this is in some sense related to what is called as x plus y sorting. Have you heard of the problem, x plus y sorting? Suppose, you have two sorted lists.

(No audio from 41:41 to 41:58)

(Refer Slide Time: 41:45)



Let us see each of them lengthened that to sorted list; just real numbers or if you like, you think of them as integers; that does not matter.

X plus Y set X plus Y, as I wrote it last time; (No audio from 42:09 to 42:18) so, the set X plus Y has n squared items; you can easily sort them in n squared log n time; that is triviality. Can you sort them in n squared time? Because you have lot of information; because you know that x is sorted, y is sorted. Can you use this information to do the sort better than; sort in n squared time? There is no lower bound that you need, n squared log n comparisons because these not independent comparisons. So, that is the big open problem.

And again, there are some lower bound which sort of proof which suggests that you cannot do better, but it is a not a...

No, n squared is not known; that is why... Well, you cannot do better than n squared because there are n squared items. So, can you do it in n squared that is not known?

(No audio from 43:08 to 43:23)

Sorry, say it again

(No audio from 43:25 to 43:53)

So, now, X is a set of points in the plane or it is a just a real number? So, when you say x i plus x plus y I, so, what are the… So, you have a set, you are saying that you… Let us say, given a set, X. (()). So, I am not sure what the points are. Suppose, you have given X as a real numbers, Y as a real numbers and you look at X plus Y, you define as not x i plus y j; I suppose, you are defining as x i y j. y j you need, because if I just do x i y i, then I will also taken the diagonal. (( )) (( )) (()) if (( )) and let us say that (( )).

If x, so you are saying that if all.

(( ))

So, if… So, so yes, in some cases, you can do because if suppose if you know that, the problem becomes simpler because what happens is that all the points settle on a single line; if the line has a negative slope, then all those points will appear on the maximum. Am I right?

(( ))

So yeah, so if you…

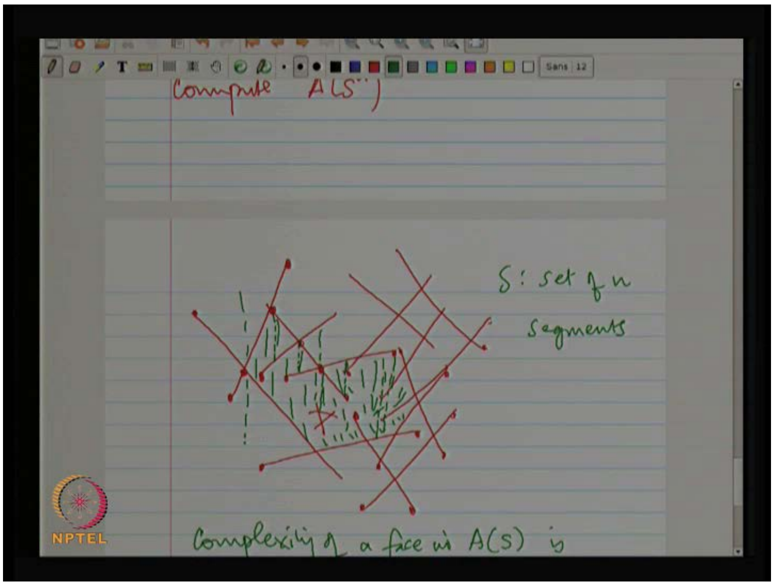
(( ))

So, so if so, you are raising a sort of an interesting question because there are refined bounds; people have asked the similar question for sorting that suppose, if you have only n distinct items, so, k distinct items, how much what is the lower bound in the sorting? And, you are asking the similar kind of questions that suppose, if the points are degenerate, then can you do sorting better? And the answer is yes and n log n does not hold in the lower bound, the better lower bounds in that case and those that is what is happening in this case.

So, for example, like there are also results; for example, like if you give something bounds on the beds, that all the bounds have some bounded beds or if you have some bound on the entropy of these co-ordinates, then their better bounds are known; the better bounds are known on sorting as well. So, that is those things (( )) answer is yes; you can do some better. Any other questions?

So, what I want to do the 5 minutes; let me do the following. So, so far I talked about the arrangement of lines, but you can talk about the arrangements of the other geometric objects. So, for example, instead of full lines, let us just look at line segments.

(Refer Slide Time: 47:21)



And, now you notice that if I just draw a set of line segments and I can again talk about the arrangement, because whose the same way that vertices are intersection points and also we need to talk about end points also, put as vertices.

(No audio from 47:46 to 48:02)

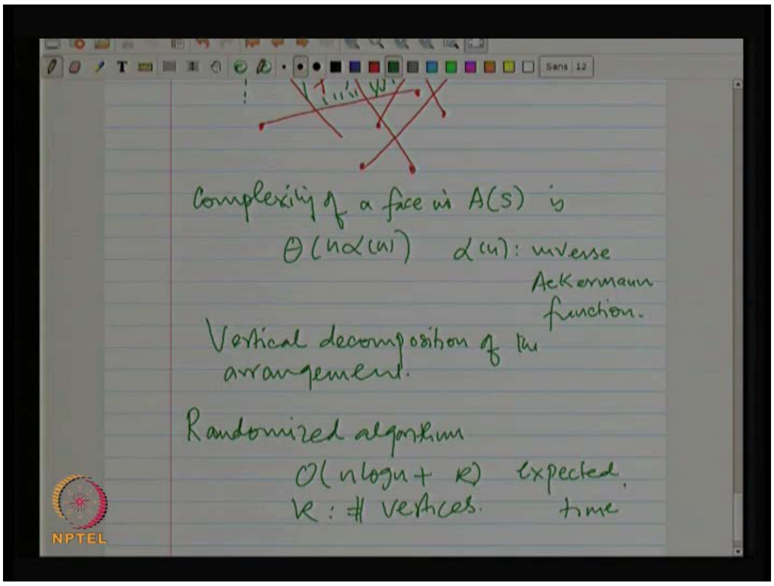
But one thing you notice now is the faces are quite a mess, let us say if I put even… So, if you look at one face, the face is a single connected component; if you the mathematically, it is defined the following; take the set of segments, think of them as the set of points, each segment; take the union of the segments of these points, remove them from the plane, then what you did? You raise think about the plane and you raise something there; just may be, think of a making a cut along each of the segment. Plane is being cut into many pieces and each connected component of the (( )) of this what is left is called a face.

So, for example, now here is a very complex face. What you notice is that unlike lines, where each face was the convex polygon is highly non-convex. Not only it is non-convex, it boundaries also not connected; in this case, the boundary consist of two components and then, you ask the question, what is the complexity of a face? How many vertices can appear on a single face is no longer that obvious that the number of vertices on a single face is only linear or non-linear; it could be, could it be as large as quadratic, all the vertices appear. And that baffle people for a long time and it turns out it is near linear, but it is not linear. So, the complexity of a face… So, let us say, S is a set of… here, alpha n is a inverse Ackermann function.

So, assume all of you know Ackermann function, right. And then, the interesting part is that this is tight; this is I wrote as theta not as big O. So, there are set of segments, for which a single face can have a complexity omega of n alpha n and that is very strange because alpha is not a natural function that we consider or you will think off. But nevertheless it is near linear, because we know that alpha is very small, has very small values in for the very large values of n.

Now, the face is very such a complex because in the case of lines, the face was nice convex polygon; it was easy to represent. And now, it is very hard to represent because how we represent such a face? And a common way of doing it is what is you have seen, I believe, trapezoidal composition. What you do is you take every vertex on the face and you draw a line until it hits another line segments. So, I have drawn too way too complicated, so, it is hard for me to finish it, but what you do is that for every end point and from every vertex, you draw a vertical line segment up and down until it hits something. And now what happens is that each face becomes a trapezoid or a triangle and so, it is much nicer and then you work with this trapezoid.

(Refer Slide Time: 51:40)



So, this is called Vertical decomposition or trapezoidal decomposition and I will skip the thing, again we can construct the arrangement and this vertical decomposition by doing the lines way which will take n log n plus and (( )) log n time and the case number of vertices. These Randomized algorithms that takes time n log n plus k, where k is the number of vertices, expected time. So, let me stop here.