Computational Geometry

Prof. Sandeep Sen.
Department of Computer Science and Engineering
Indian Institute Of Technology, Delhi

Module No. # 04

Convex Hull Different Paradigms and Quickhull

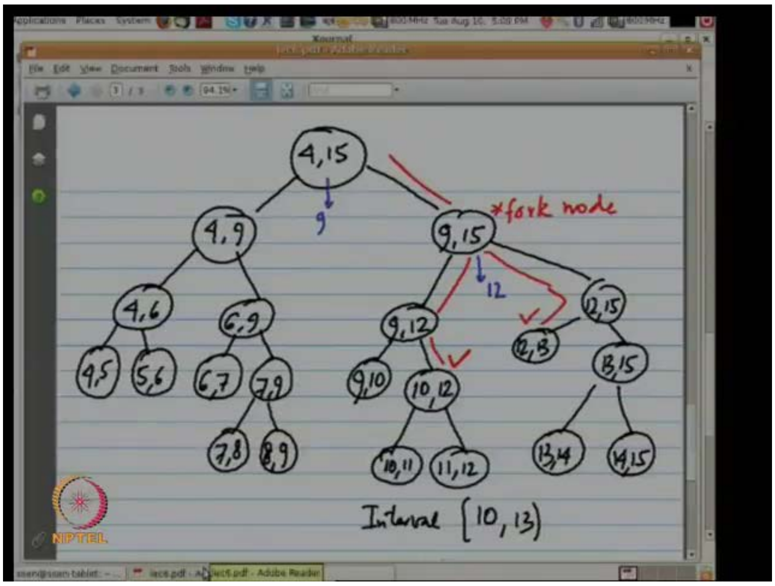
Lecture No. # 07

Convex Hull

Welcome to lecture 7 of computational geometry. We will start on a new topic rather interesting topic in the context of computational geometry namely convex hulls, but before I get into the convex hull problem, just let me know if there is any questions from the last lecture.

So, I hope that you understood the basic essence of what I am calling interval trees. So, interval trees are rather useful in many other context service, see later particularly when we deal with things is called rein searching. But I just thought I would bring up one thing to you notice.

(Refer Slide Time: 01:10)



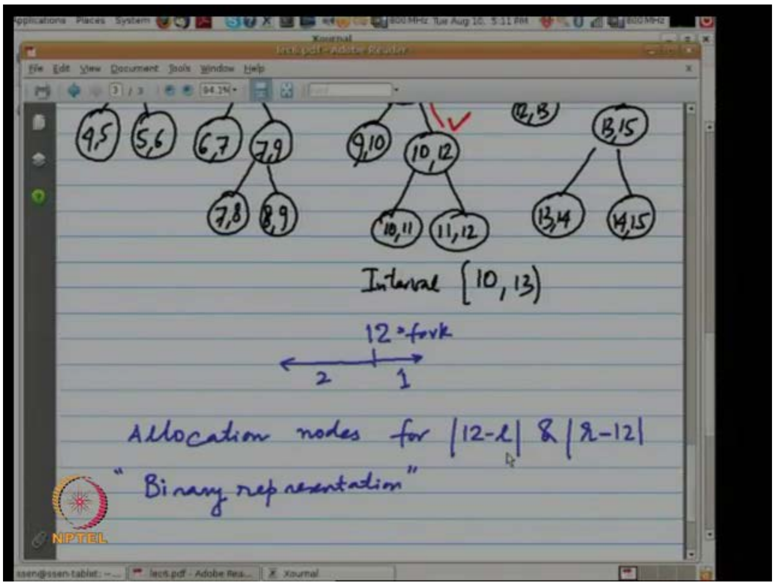
So, we did this example last time for the interval trees and we observed that for any interval the maximum number of allocation nodes for a interval's tree of n leaf nodes is at most 2 times log n.

And you know kind of informally even proved it and you know formula is the proof and refine it, but there I just realized that there may be an easier explanation of the whole thing, if you think about the interval. So, interval is, I cannot write this p f i.

So, the interval is let say l comma r in this case l equal to 10 and r equal to 13, so when we do this allocation, we encounter what is called as a fork node where the path splits into 2 parts.

Initially, the interval is small enough that will actually not span the entire node. So, it will fall to either left or to the right sub trees. So, you keep traveling till you hit a node where part of the interval is to the right of this node, and part of the interval is on the right sub tree of this node, and that is what is called the fork node.

(Refer Slide Time: 02:35)



So, at the fork node, for every node there is kind of a splitting value right for the node a 9 come of 15, all the values less than 12 are on the left hand side, and all the values are the greater than 12 on the right hand side.

So, now since the interval splits into 2 parts, the part of the left interval will basically is beginning from 12, up to or let us say it is l, 12.

So, if it is l, 12 the right interval is 12, r. So, we have to find allocation nodes is a left sub tree for the length 12 minus l, and in the right sub tree we have to find it for the length r minus 12.

Now, you can actually think about this whole thing as a, it is very analogous to binary representation, because these trees are basically chunks of 2 and 4 and 8 and so on so forth.

So, we are trying to find the allocation nodes of the left sub of interval which is of length 12 minus l. So, any binary representation can have at most log n bits, it is not in log n it will depend on the lengths.

So, if the length is b bits, so it will be not b more than b bits actually b allocation nodes. So, that is another sort of justifying or it is actually like a formula that you can even without actually doing the allocation in the tree, I can actually figure out analytically which nodes it will be allotted to it.

So, that is another way of thinking about it, we have just a small observation. So, we will come back to this notion of interval tree. In fact, this interval tree forms you know also known as some variation is known as a segment tree, some variation is known as range trees, something that we will again touch upon in near future. So, today we will start on this topic called convex hulls.

Why constructing the tree, why do not you find the medium, so that the root could be decided for a (( ))

What is the question, why do we need to find the medium?

Why don’t we need, I asked you in the last class that we do not need this tree to be balanced.

No, I did not say that I said that the tree has to be balanced.

If it is balanced then we need to find out the medium in every tree. So, that will add some complexity we are constructing the tree.

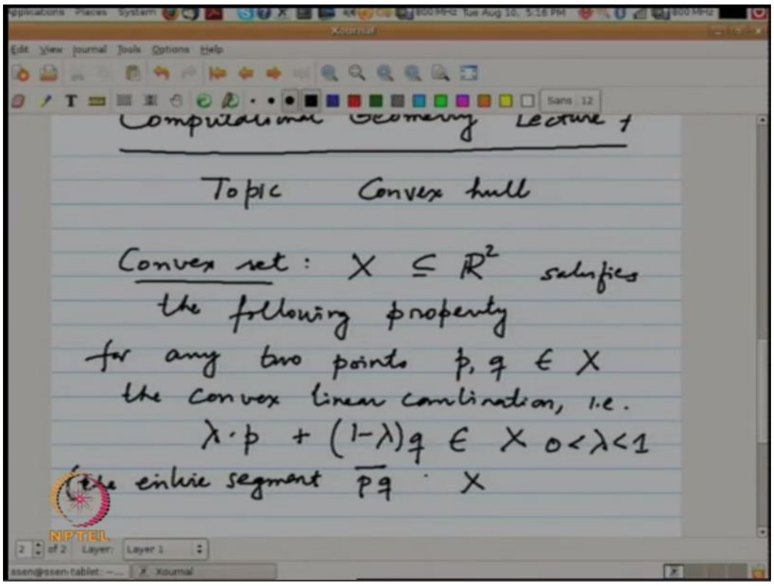
No, in many cases the tree how much time do you think the tree takes to be constructed, n log n. So, even if you find the median at every level it will still be n log n.

See you are actually starting with this sorted set of values. So, median is already is there that information is already there, even if it is not there you find median of n can take order n times, then you are finding median of 2 sets of size n over 2 that will also take order n time.

So, even if you had to find the median it will still it is still be n log n. So, that does not add any complexity. So, convex hulls, I believe many of you will be familiar with, but nevertheless let me go through some of the formal definitions

So, before defining even convex hull, let me define the notion of what is called a convex set and this is in the geometric setting.

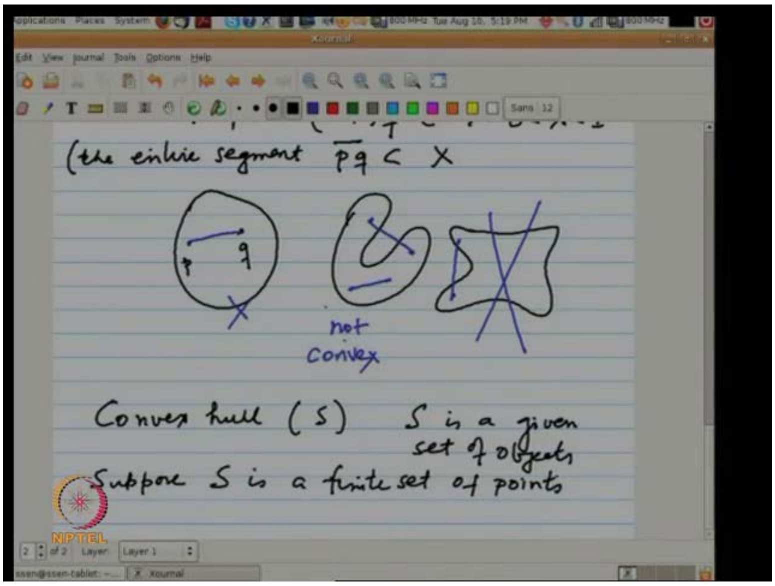
(Refer Slide Time: 06:07)



So, a convex set basically satisfies, let us say a convex set x is a subset of. So, if you are talking about plane and we were talking about r square satisfies the following property for any 2 points p q in x, the convex linear combination that is lambda p plus 1 minus lambda q also belongs to x for lambda between 0 and 1, or in other words this segment the entire segment p q lies within the set.

So, let me draw an example. So, when you are talking p and q these are vectors, they are 2 dimensional points and this definition extends to any dimension. So, what is it.

(Refer Slide Time: 08:21)



So, if you have a set like this, and I can have various sets I can have a set like this, I can have a set like this, I can have a set like this.

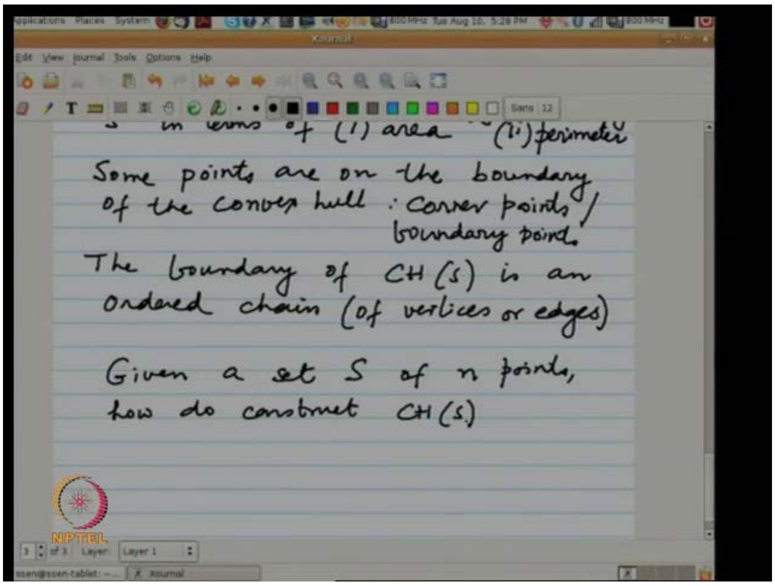
Now, if I take 2 points p q, suppose I take point p q it says the entire segment lies within the set. So, this is my x. So, that is 1 example I can take p q to be this, the entire segment lies within the set, but the definition says for any 2 point.

But, if I take p and q to be this, the entire segment does not live in x. So, by this definition this is not convex, how about this again if I chose these 2 points this segment does not lie completely within the set x.

So, again this is not convex, so convex is something essentially something like a fact. And in higher dimension again it is an analogous extension. So, for any subset of 3 dimensions, again it takes any 2 points in the set the entire segment should lies within the set only then it is convex. Now, what is convex hull. So, convex hull is defined actually for a set of given objects. So, convex hull of s is a given set of objects, now objects could be anything like points, lines whatever triangles this that.

So, let us understand the basic definition with respect to point sets, suppose s is a set of finite, well not necessarily, but I am saying that suppose s is the finite set of points then the convex hull of s is the smallest.

(Refer Slide Time: 11:15)



Convex hull of s is the smallest convex set that contains s right, example so, suppose these are my given set of points s, we want to define some kind of a region which must be convex and contains all the points.

So, of course, I could have this disk is convex essentially, so I could have a large enough disk that contains all the points, but then it also says something about being the smallest. So, this is clearly not the smallest, because I can set of shrink that is, you can look at this example and sort of shrink the disk and make it a little smaller, so sort of cut corners.

Containment, it happens that is also in terms of parameter in terms of perimeter, but let stick to the containment part. So, the smallest convex set means what, that the 1 that I have not drawn in dashes is completely contain with in the previous one, and this is also contains the given set of points, struggle a little bit more and may b even it makes it smaller.

So, why even have some kind of a curve, so maybe we should have something like this. So, what I have drawn right now is with straight line segments connecting some points. So, visually at least it is clear that you cannot make it or cannot shrink it any further.

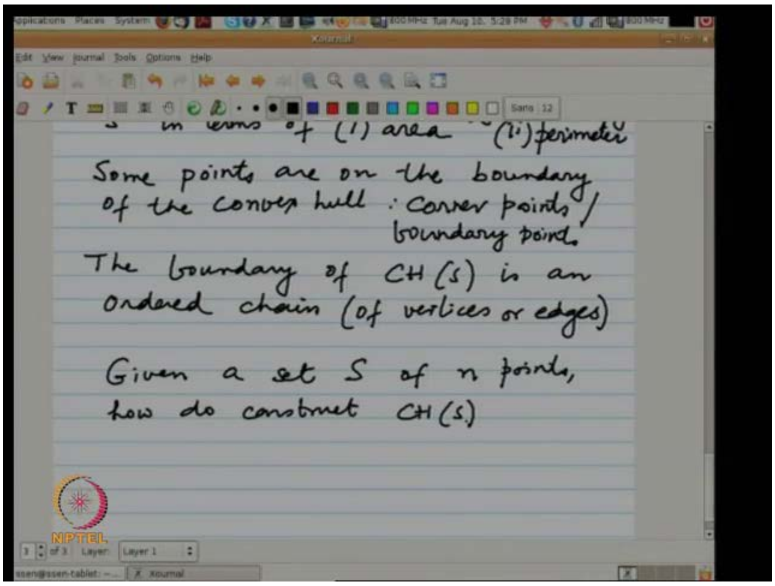
See, it must be convex, it must contain all this all the points and it must be the smallest. So, this one contains the 1 that I have drawn the straight line segments, contains all the points some other points are actually on the boundary. So, if that is the boundary of the convex hull.

And some of the points are inside and if I want to make it any smaller, there is a problem the problem is that you look at the at the corner points, these must be in the convex hull. So, I cannot ignore them, and if I try to make it any smaller, what I should do I have to sort of may be do bending like this, because those points must be included in the convex hull. So, the my only chance is try to sort of try to bend these things, and the other option is to have some holes, but then we just noticed that if there is a hole then it does not satisfy the property of convexity, because I can have 2 points p and q and entire segment is not going to continue.

So, it cannot have any perforations, the boundary cannot be bent in words, otherwise the same problem that I have these 2 points the boundary draw it, by join it, by a straight line as the entire straight line is not going to contained in the convex hull. So, at least in formulae, this seems like, we have actually hit upon the smallest convex set which is also you know by definition a convex polygon, it is a polygon. So, a convex in 2 dimensions, a convex smallest convex set is convex polygon.

Now, I just made if you comment that it can be proved that the convex hull is also the smallest area convex polygon containing the set of points, it turns out with the smallest convex in terms of perimeter also these are properties that can be proved I am not doing right now. So, it can be proved, so, I will write c h for convex hull. So, convex is the smallest convex polygon containing s in terms of area perimeter, but they have these to be rigorously proved, but I am just making the observations.

(Refer Slide Time: 17:44)



The next question is. Let me also define these things, so some points are on the boundary of the convex hull.

Now, again you can argue that the some points must be in the boundary of the convex hull otherwise you can squeeze further, if there are no points on the boundary I should be able to at least in finite similarly somehow shrink the boundary, so some points have to be on the boundary and these are all called corner points or boundary points.

The boundary of the convex hull is an ordered chain either you can call it as vertices of vertices or edges whatever you call.

So, the subsets of the points, the finite set of points that are given to us will define the corner point, so the boundary of the convex hull, and the description of the convex hull is an ordered chain which is basically some subset of the given set of points, that cloud be also be a natural representation of the convex hull.

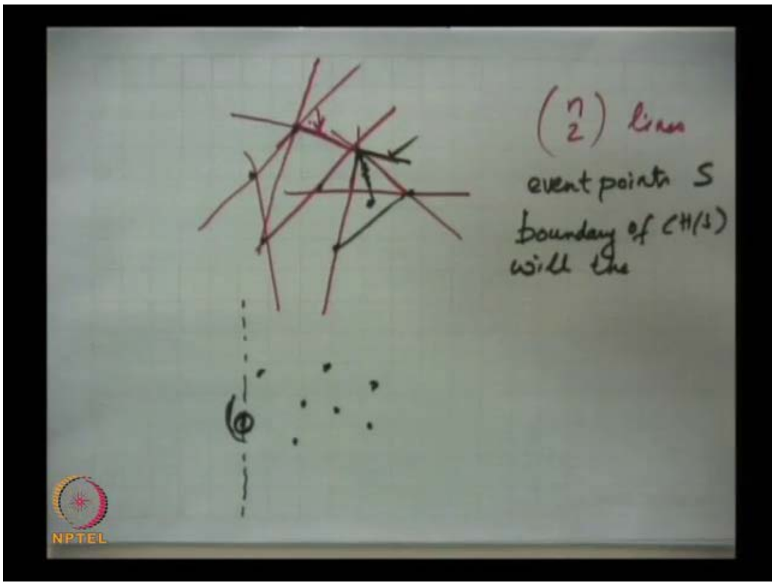
This is by the way is only for 2 dimensions, the similar definition analogous definition extends through 3 dimensions. Now, once you go to 3 dimensions then notion of this ordering any more other definitions go through, that there will be some points on the boundary again same thing, and it will give you a convex what is called a convex polyhedron, convex polytope what you call it, but then the boundary of the convex hull is no longer something that can be ordered like we can do in 2 dimensions, but it is a more complex structure, you just think about like a football or something another the boundary of the football. We will probably discuss little about that, but right now let us focus on the 2 dimensional convex hull.

So, how do we construct a 2 dimensional convex hull. So, given a set of points the algorithmic question is that given a set S of n points, how do we construct CH of s.

Construct all the lines for the square possible lines and formal lines we have, and these points are the points which are on the maximum line and the minimum line(())

So, let me articulate what is been said. So, what is been said is following, let me use the other medium.

(Refer Slide Time: 21:23)



So, what has been proposed here is that you'll take some of this points given to you, you draw the pair wise lines, all possible lines where the too many of them I cannot draw the all of them.

So, you draw all these n chose 2 lines and then perform a line sweep, that is what is been said. So, you perform a line sweep and at any point. So, what is being said here, what is going to define the convex hull here, what is an event point here, all the points are event point, which points, the given set of points or all the intersection of the lines. No, the given set of points.

So, the given set of points is all the event point and as you sweep past what you do you do, in track of what is the minimum and maximum of lines.

Well, let us explore that first, so the minimum and maximum of what? Of y is the value of the lines.

You are saying the event points are the points given 0 points. So, event points is basically s and boundaries boundary of convex hull will be the, what are you saying that which lines. So, there are many lines were intersecting, so which is going to define the boundary, the maximum or the minimum, the minimum connected to the event point.

So, what are you saying that there a number of. So, this event point or whatever they are many point. So, each point basically defines n minus 1 lines, there are n points. So, you are saying that you look at all these things and as you cross the minimum in terms of y, so after a cross this should be basically what should define the boundary, and then after this minimum and so on.

And of course there could be some complication like this, what are we are going to do about this, just wait a minute. So, there is some ambiguity here, this is 1 possible which should be actually the convex hull boundary, but then this is minimum then what you are going to do about it, but then I could have a point here, the maximum will be the upper boundary.

I thought you are saying minimum.

No, maximum at which point you are finding the upper boundary as well as the lower boundary.

No, but this is not the lower boundary why should this be a lower boundary the lower boundary will be the lower boundary will be something like this. No, why should I miss this

Because there is a line below this and there is a line above it, so the convex hull will be (( ))

No, there is a laser line above it, there are many lines which is the one that you are following.

The highest one.

Now, this is the highest one. So, I am not saying that these things will not work out, but I think there could simpler attempts, and then this can probably refine into some kind of algorithm, but it is not going to be particularly.

So, we have drawn the nth choose 2 lines, so it is at least n square and then we are doing lines sweep, and you have do even points. So, what you are saying I think can be defined into a proper algorithm, but why not think about something even simpler.

So, can you tell me which point must be on boundary of the convex set?

The top most and the bottom most.

So, the top most and the bottom most point must be on the convex hull is that clear. So, it is the left most and right most. So, once I know there is some point on the convex hull, so let us look at another example, so these are given is given set s and this is the left most point, I have identified it is left most point in terms of x coordinate. So, this point must be on the convex hull, can you argue why, this one, the one that is in circle.

So, was the previous diagram visible, when we were looking at it. So, now, I am giving another set of points and this is a left most point. So, why should left most point be on the boundary.

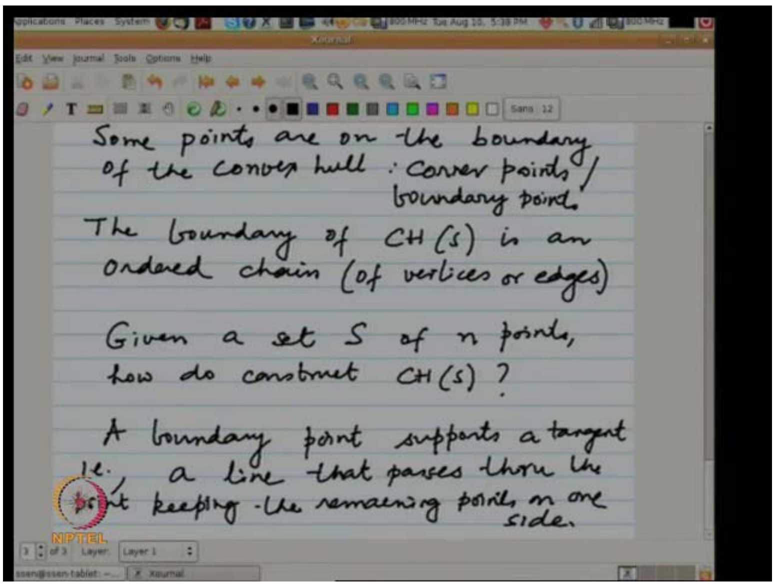
Because, it is a convex polygon the angle that is here on the (( )) by these by the 2, our other points that is to be less than 180 degree, so it is keeping the kind of (( )) it is not on the boundary and it will be not on the convex hull (( ))

That is a question. So, why cannot the boundary. So, why does this point have to be in boundary, why is in the convex of hull something like this.

Because we can draw line from that point and if it is the less most point any other point will have like, if you do not know any other point, join them this point and that point(()).

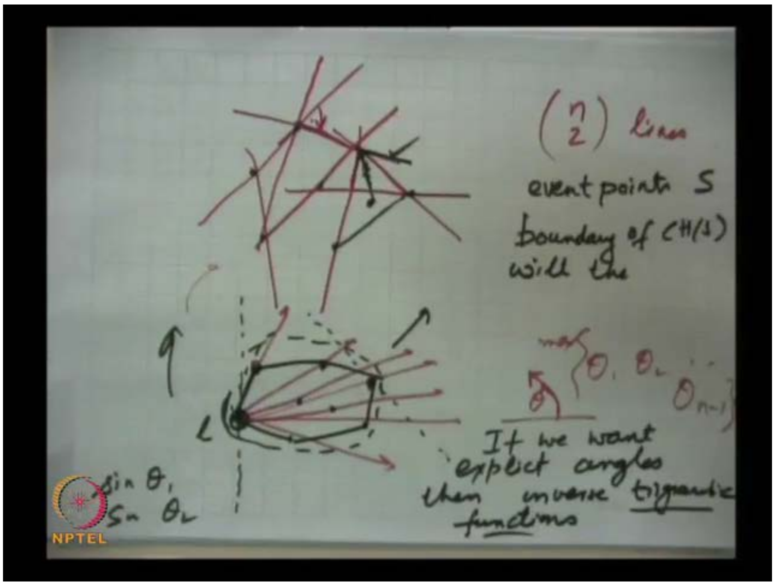
For this actually there is a better characterization, so I did not quite characterize the boundary points. So, boundary points can also be thought about as let me write here in the boundary points also such that you can draw what is called it a tangent through a boundary point. A tangent through a boundary point and a tangent is a line that passes through point and the remaining points should be one side of the tangent. So, there is notion of boundary point and tangent.

(Refer Slide Time: 29: 21)



So, let me write it here, a boundary point supports a tangent I e if a line that passes through the point keeping the remaining points on one side. So, if this is another characterization of boundary point then clearly this is enough to say that if I draw a vertical line through the left most point.

(Refer Slide Time: 30:24)



Clear all other points should be to the right of it and therefore, it must be a boundary point. In fact, this is pretty useful characterization of a boundary point, and by that actually, no which ever orientation you choose whether it is this orientation or this orientation in along in any orientation you can very easily.

See this point must be boundary point, because you know along that orientation this is the right most or whatever this is an extreme point. So, an extreme point is or boundary point can always support a tangent, because you can draw a line through it such that the remaining points will be one side of it.

So, the left most point is on the boundary that is given to us now what we can do is, we can think about this convex hull again, as you know if there were something like nails that we are driven out to the plane, you put a rubber band around it, the convex hull takes the shape, basically whatever shape the rubber band takes that is also the convex hull. It is kind of tight between 2 boundary points and straight line segments joining them. So, once I have identified 1 convex hull, I can imagine that I am going to tie a rope around this sort of circle a rope around this, and whichever points I hit basically should define the boundary points of the convex hull.

So, once I have identified, let us say l which is a left most point, to get the next point all I am saying is that I take a rope and just try to tie it around. So, we can move the rope till it hits this one.

So, this is the next point of support of that rope and then of course, I draw this, the rope basically is tight around this, then again I tied the rope, I pull the rope and then it needs the next support point, and like this basically I will end up identifying the boundary points.

I thought this would be a more natural way of finding the convex hull. And, if you do this procedure what is the running time and how many operations are you doing, and how do you even execute this thing, I mean we are not physically actually going to pull a rope around. So, algorithmically how should we do it?

We have to get the angles from this point to this.

So, it appears that we are actually talking about something like the angle, so if you look at this point, you know we draw this all possible. So, if you join it with the other points, and you measure your angle in the anti clock wise direction or something, then how would I like to measure the angle.

Well we are actually trying to walk in a clockwise manner, so which is a slope that we should see, what is this point basically.

So, if you measuring angle in the anti clockwise direction, this maximizes the angle right. So, if this is the theta, so everyone all this points will define some theta 1 theta 2 etc up to theta n minus 1, and the max of that will be the next point of support. The, next stage going out of there and likewise you continue from here and so on.

But, if you are just sort of tying the rope around, I mean you going in one direction you are going to trace it like that.

You can always change the axis right, I mean change the change the frame of reference, which ever point you are in you change your frame your reference such that everything in the in the in the positive coordinate.

So, how do you do angle computation. We cannot measure the length like that.

So, which change to polar coordinates you know we do some.

We can just find the slop between 2 points.

So, well one way of defining angle is you can take that, so you can compute either cross product to dot product. You can treat this points, you can take this as origin, compute the dot products, and then of course from there, from the lens you can normalize them and you can compute either, sin inverse or cos inverse and either the dot product or the cross product whichever you're more comfortable with.

So, both will be normalized in some sense whatever it is, so you can normalize and get it, so that will amount 2. So, if we want explicit angles then we need to do inverse trigonometric functions right

We need the order right we do not need have an exact measure, so we need which is greater with smaller (( )).

That's a very good observation. So, are you saying that we may not want actually compute the theta, but just from the fact that we do a sin theta; we can make some observations about that.

And we even just find out the difference between y coordinates, and a length of a perpendicular in that right angle triangle as that decreases that angle theta will decrease(( )).

So, I am very happy that you the people have quickly realize that we need not actually do inverse trigonometric function, because these are transient dental functions and there is no real good arithmetic model for this, you can only do some kind of approximation.

At this point we would like to avoid trigonometric functions, we would only like to use normal arithmetic functions like multiplications, divisions, perhaps may be square root at this point I am not even talking about square root to the extent possible we should stick with the algebraic functions not use trigonometric functions.

And the observation was that you know even by just computing sin theta, and without the theta we can perhaps compare to angles. If, I know sin theta 1 and sin theta 2, I should be able to do some comparisons between theta 1 and theta 2. I will suggest another method and this will be actually something similar to this, but it will be more useful and it is more standard, and for that we will rely on essentially the cross product thing.

Just cannot be seen just the maximum value of the y 2 minus y.

The maximum value of that length to the perpendicular which give us y at the height.

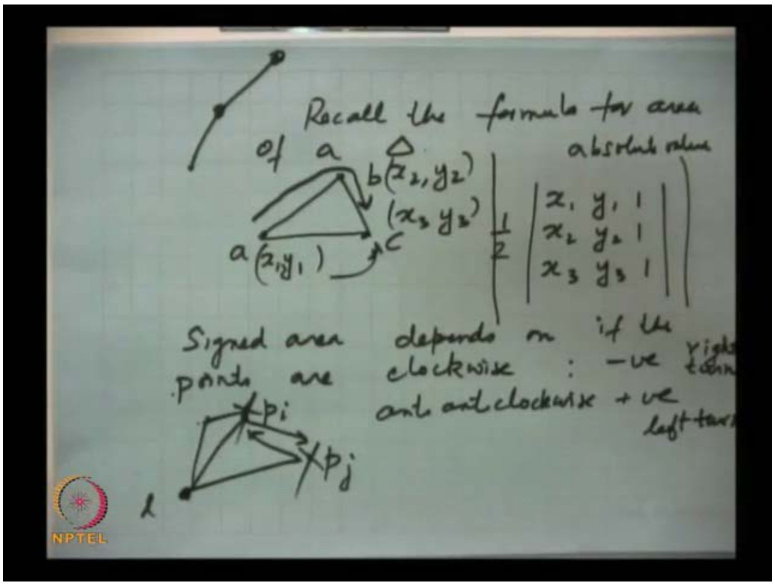
No, I am not sure about what you are saying.

Here it is a left most point have you seen the next point which have the highest value of y

Highest value y, highest value of this could be much higher.

No, those kinds of things will not work. So, it is a 2 dimensional problem.

(Refer Slide Time: 38:53)



So, you have a point, you can have a very high point, but it will go like this, this is not the next point.

(conversation not audible 39:01 to 39:04)

No, this is not the highest value, this is the highest value. So, you cannot simply like that. So, let me give you a sort of more computationally, more meaningful way of doing it so, for that just recalls the formula for area of triangle.

So, let us say 3 points a b c let us call them x1 y1 x2 y2 x3 y3. Does anyone remember the formula for the area of this triangle, ok tell me. So, x1, y1, 1 x2, y2, 1 x3, y3, 1, good you have a good memory and the half of this.

This is the area of triangle and this actually is derived from the cross product. So, the area of this triangle actually, I mean to be more precise the area of this triangle is actually the absolute value, but if you look at the singed area that is I do not take the absolute value. The signed area depends on if the points are clockwise. So, you remember a lot of analytical geometry, so clockwise and anti clockwise. So, the way we wrote it down here x1, x2, x3. So, it is actually clockwise, so in the case of clockwise, it turns out it should be negative and the counter clockwise it is positive.

So, either you are going like this or going like that. Now, this turns out to be a very useful primitive for a lot of geometric problems including this one and how is that.

So, this is essentially a left turn or a right turn primitive. So, when it is clockwise it is right turn, when it is counter clock wise it is a left turn, in the original 3 points a b c is a right turn and a b c is a left turn. Now, you see when we are looking at finding the let us say the largest angle in the counter clockwise direction, essentially let us say this is the left most point l we are looking to eliminate 1 of the 2 points, whatever these points are let us say p i and p j. So, I need to know whether we are, you know p j is like this or like this.

I am avoiding, I do not want to, I do not even have to find the slopes explicitly, the good thing about this determinant is that it does not have to do even divisions anything it just multiplication, you want avoid division to the extent possible.

So, just by figuring out whether you know l p i p ‘j is a left turn or a right turn, we know whether to eliminate p i or to p or p j. So, it is like whatever like a minimum or maximum computation. So, I have this point, the anchor point and for the remaining points I am just trying to find out, and whenever we compare 2 points we will retain the 1 that basically is the right turn.

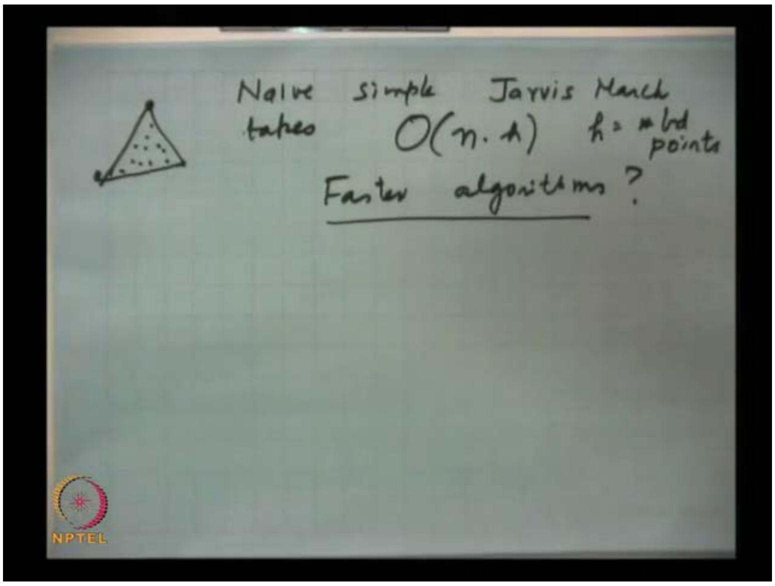
So, this point will get eliminated and this will remain then we take pickup another point may be this one and then point will get eliminated and this will remain, it is like basically minimum maximum computation except that the primitive that is being used is this left turn right turn.

And it is a purely algebraic function, no trigonometric computations, no angles and not even division. So, this turns out to be very useful primitives for many other geometric problems, and if you think about it convex hull is sort of the first problem we are looking at which truly has a 2 dimensional character, in the previous problems we looked at your 2 d maximum etc. we could only do with the comparison model you know they are no need for any kind of slopes something like that, you know we just had to take to compare the x coordinate or the y coordinates you know nothing else is required.

Convex hull you know we do require some kind of slope computation, but we are able to circumvent that using another kind of primitive, but it certainly it is not a problem where you can just do use comparisons that will not suffice, it is not some just less than or greater than know we are actually we have to compute the determinant or something else, something that we will give us some idea about the slope. So, this left turn or right turn will turn out to be useful for many other applications.

So, if this is the case you use this primitive then clearly the running time of this thing that we just talked about, what will be that. So, we will take a maximum minimum computation at every step.

(Refer Slide Time: 45:51)

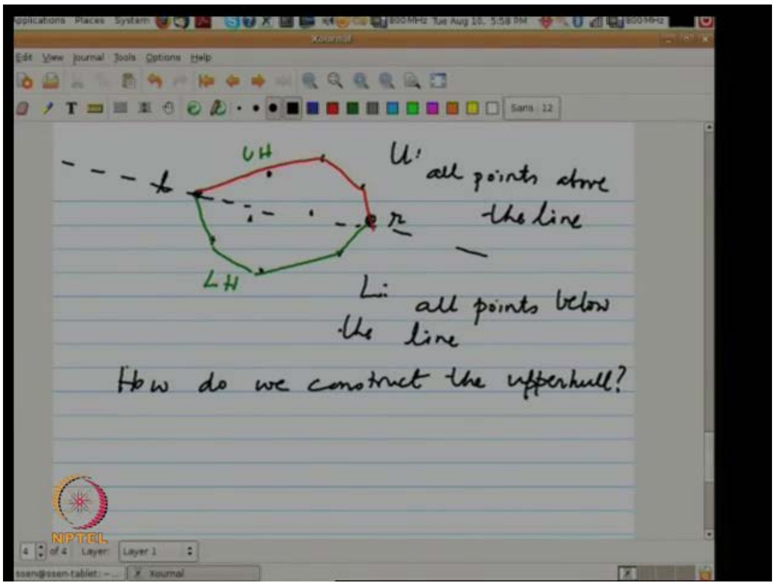


So, this naive simple algorithm, it has a name also called Jarvis March actually, it takes some order. So, I will actually be more precise in writing this, I will say takes n times h where h is the number of boundary points, what is the smallest convex hull.

Triangle; because that is a smallest convex polygon. So, if it happens that your input is like this and all other points are inside, this gives you a linear time algorithm actually. So, it is not, it is pretty good when the number boundary points are small, but when all the points could be on the boundary, and then you are in n square kind of situation, which is what even that complicated thing about joining every pair of points, joining with the line and doing a line sweep would give you something even more than n square, because you have to do a line sweep with n square lines. So, this is simpler and it is in the worst case it is n square, but then for some cases it is n h, question is that you now can we do better that always is the question, can we do better.

So, Faster algorithm; I will only just give you the idea and we will do the analysis tomorrow next class. So, for that whatever you do is I will define the notation of what is called an upper hull and a lower hull.

(Refer Slide Time: 48:09)



So, if we are giving this points and we identify the left most point, so this is the left most point identify the right most point, this is the right most point, let us join it by a line. So, all points above the line and these are all points, so call it u and all points below the line call it l. I have these 2 joints subset of points, the upper hull is basically the part of the convex hull that is above this boundary, let me have some more points, so this is my upper hull and the part below the boundary in green is a lower hull, this is my upper hull, this is my lower hull. Why have you defined this.

So, what we will do is that we can identify the left most and the right most point in linear time, and separate out this 2 point sets subsets l and u and we will just describe the algorithm for the upper hull and for the lower hull, it will be an identical analogous algorithm. So, we will only focus on constructing the upper hull.

So, how do we construct the upper hull that is what we are going to focus on.

Do you think to merge the upper hull and lower hull you will get the combination.

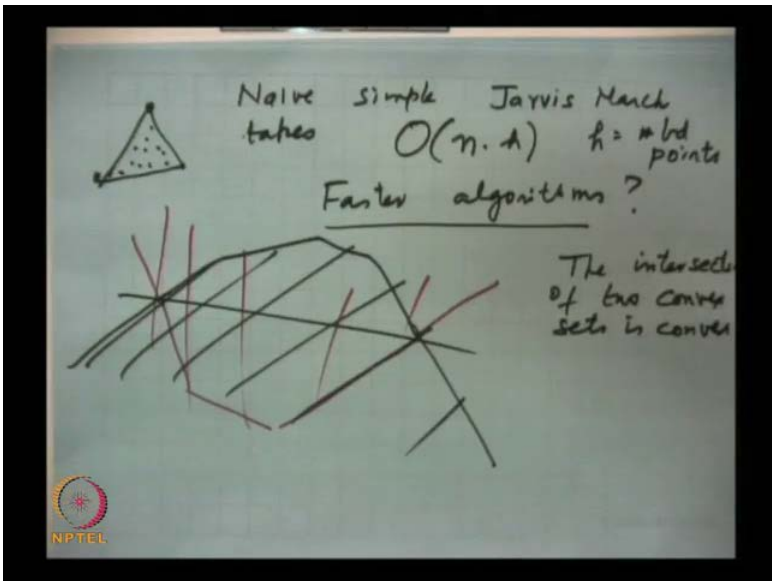
You just paste them that are not strictly merging; it is just the part of the hull that is above the line the part of the hull below line. So, if I can construct the upper hull and I can construct the lower hull, there is no real merging it is just pasting.

What the whole thing will be the pasting of those hull

Yes, so is that is that clear or not clear.

No, there is a point, so you should be able to argue it on the basis of, see you can think about the upper hull. So, this is my separating line you know this is my upper hull and there is some lower hull.

(Refer Slide Time: 51:06)



The way to think about this which is clean is that you should think about the intersection of this unbounded thing, and the intersection of this and that will be this, and there is one thing I did not mention that the intersection of 2 convex sets is always convex, by definition if you can think, because the entire segment should be completely contained in the set, so if you look at the intersection it lies in both, and therefore the intersection is also convex. In fact, half plane is a very good example of a convex set. Actually, let me stop it I think I will resume tomorrow.