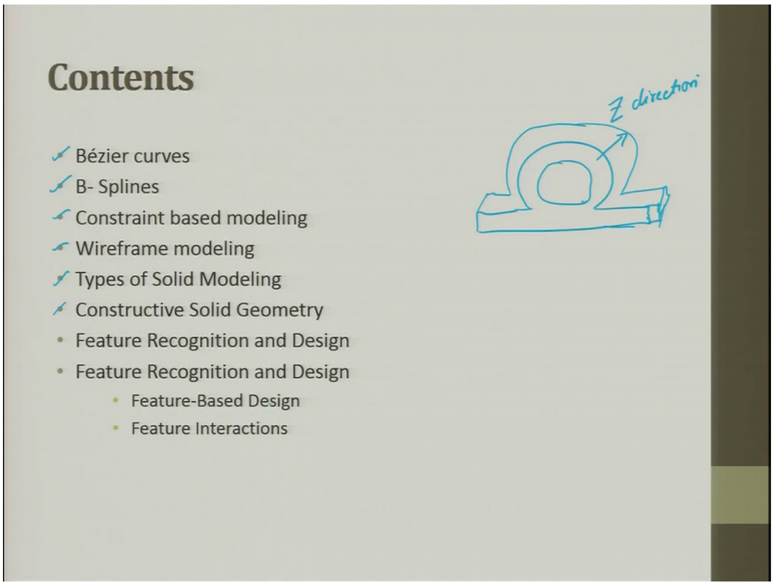
Rapid Manufacturing
Prof. J. Ramkumar
Dr. Amandeep Singh Oberoi
Department of Mechanical Engineering & Design Program
Department of Mechanical Engineering
Indian Institute of Technology, Kanpur

Lecture – 38
Rapid Product Development, CAD (Part 2 of 3)

Welcome back to the lecture on Computer Aided Design and manufacturing. We are more focus towards CAD in this lecture.

(Refer Slide Time: 00:25)



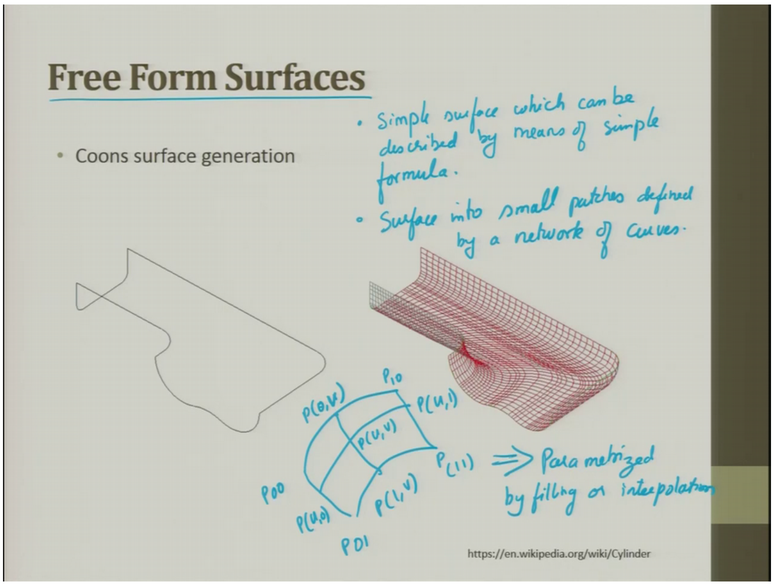
Last lecture we saw more towards generic simple 2-D representation in CAD and then we slightly extended how from 2-D to 3-D has to happen. In that case, we started talking about translation and sweeping, two mechanisms or two algorithms which are in built in 2-D such that you can create 3-D structures. We also saw a simple parametric form where in which you draw a 2-D structure and then you sweep it.

For example, if you can think of your plummer block so those people who are not aware of a plummer block the schematic diagram look something like this ok. So, you put a bearing inside. So, here if you draw the 2-D structure and if you just extrude because it is going to be along the Z direction the same. So, you go here to whatever distance it is and correspondingly you also make the circular also there and then you try to get it.

So, this one along the Z direction whatever you pull, from the 2-D structure we get a 3-D object. So, such simple things we were discussing, but life is not as easy or as simple as we think. We moved into 3, we have to now moved into 3-D structures. In 3-D also we will have free form which is the need of the r because today we are looking at multipurpose or multi application or multiple functional doing components which are used in the products.

So, keeping that in mind we will move through this lecture. So, in this lecture we will try to see Coons surfaces then move to Bezier curve, B-splines, constraint based modeling then we will see Wireframe modeling then different types of solid model then constructive solid geometry in the 3-D form, then feature recognition and designing in that we will try to see feature based design and feature interaction.

(Refer Slide Time: 02:43)

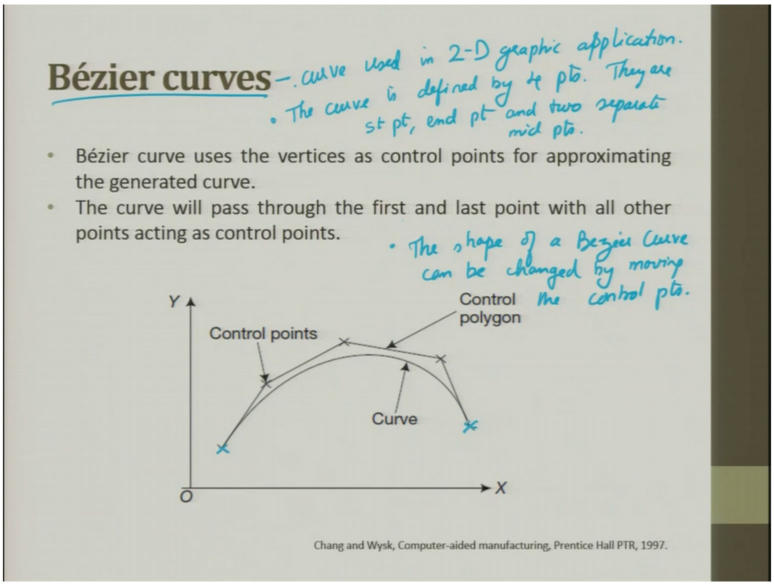


So, when we talk about free form surfaces first thing comes Coons surfaces. So, Coons surfaces are relatively simple next step from whatever we have studied 2-D and then we saw translation rotation and next step moving towards 3-D is going to be with free form and then 2-D also free form surfaces is very important. So, we will see how to draw a curve and other things.

So, Coons surface is basically a simple surface which can be represented, which can be described by means of, by means of simple formula. So, in reality you will have many surfaces which these surfaces has to be drawn. So, what we do is we try to divide the surface the easiest thing is we will try to divide the surface into small patches ok. And which is defined by a network of curves. What we get is a Coons surface. For example, so, this can be P naught 1, this can be P 0 0, this can be P 1 0, this can be P 1 1 and you can define the functions and you also define the directions with which the curve has to (Refer Time: 04:51) for example, here you can put it as P u, 0, P u, v and then you will have P u comma 1.

So, you have seen, now you have you have discretize this curve a parts into several small points and then you are trying to represent those points. So, this is here Coons surface. So, generally in Coons surface each patch, the entire patch is parametrized. So, parametrized by interpolation by filling or by interpolation by interpolation, we try to get those boundaries and then we try to solve.

(Refer Slide Time: 05:55)

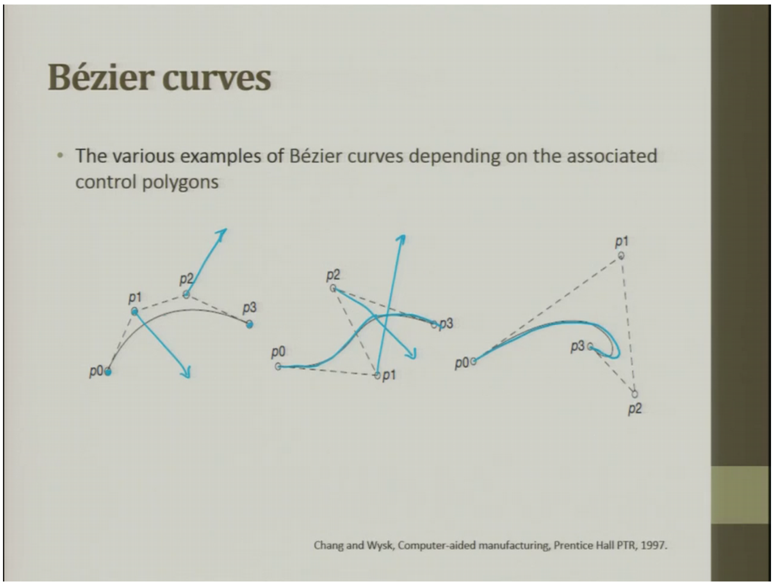


Next we will see the Bezier Curves, those curves which we saw earlier also had some had some technical problems. So, where in which you define certain points and you define the complete patch. So, all you have to do is wherever you wanted to attach it or fix it up to as free form surface the entire patch which been already defined so that patch comes you fix it up and then you start getting the smoothing of the curve. So, when we look into the patch also we had limitations that we were not able to control and change those patches. So, then came the need for going into Bezier curve ok.

As far as Bezier curve is concerned, if we wanted to draw a curve where in which we know the starting point, ending point we define some more control points and try to draw the curve which is close to those control points. So, those options are given by this Bezier curve. So, currently we are discussing 2-D, we also have a similar one for 3-D. So, Bezier curve is nothing but a curve that uses that is used in 2-D graphical applications ok. The curve is defined by 4 points.

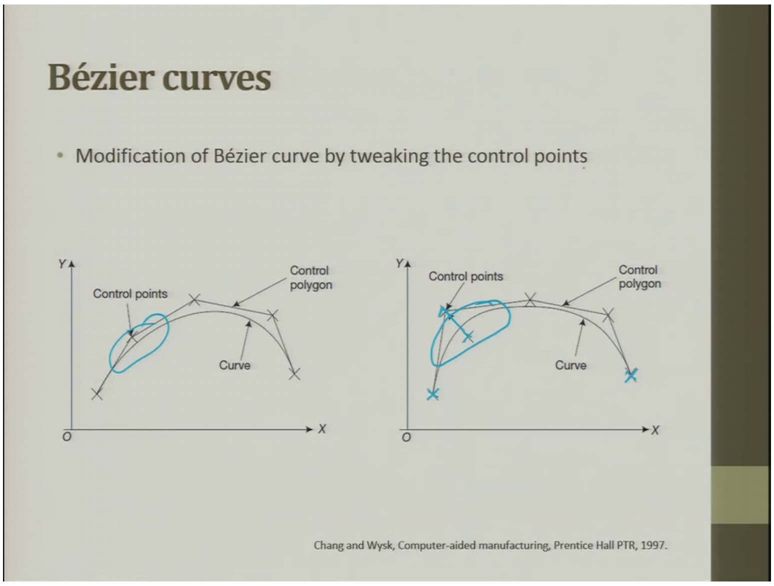
They are start point, start point, end point and two separate mid points ok. The third thing is the shape of a Bezier curve can be changed by moving the control points. So, Bezier curve uses a vertices as a control point for approximating the generating curve. The curve must pass through first and last point that is compulsory, with all the other point acting as a control points. So, you do not have to pass through it, but these control points the curve has to go close to those control points.

(Refer Slide Time: 08:59)



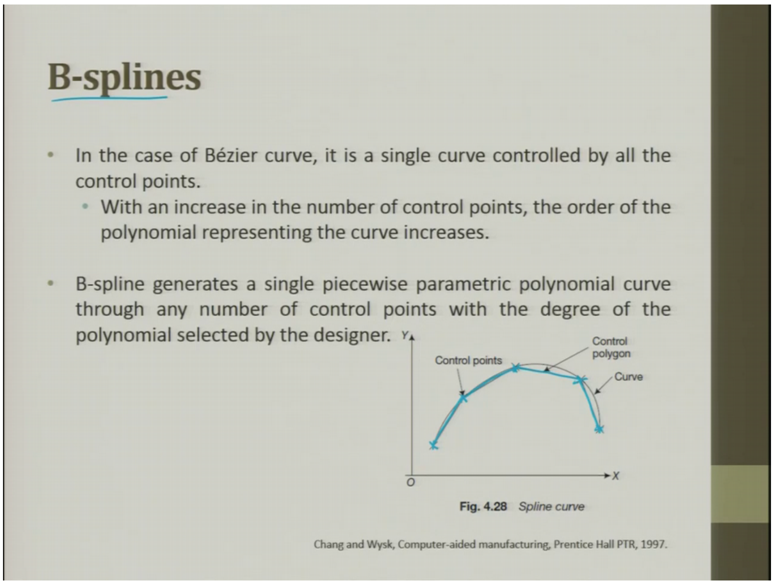
Now, you see various examples of Bezier curve depending on the associated control polygons. So, we had a start point, you had an end point then you had 2 points which we have designed which we have defined and then the curve moves along this, along this path. So, suppose I define p 0, p 3, but change the location of p 1 and p 2, then I get a Bezier curve of this shape. I have just change the control points. But the start point and end point are approximately the same. The next one is I have used the same starting point and ending point, but still moved the p 1 to this side and p 2 somewhere here. So, you see a completely different curve profile coming. So, what did as per as application point of view is concern, the control points decides the curve profile.

(Refer Slide Time: 10:05)



So, there is a modification of the Bezier curve by tweaking the control points you can change the, you can change the curve ok. And here what happens is you can see here I have just moved a control point from here to slightly here. So, this portion of the curve is tweaked, but it is not only this small patch, but corresponding to those patch all other points are also moved. So, it will have a starting point and it will have an ending point ok. So, the modification of Bezier curve by tweaking the control points you can try to tweak the complete curve.

(Refer Slide Time: 10:53)



The next advancement. So, I do not wanted to control I wanted to control at as closely as possible. So, what limitation was there in Bezier was removed by using B-spline. In the case of Bezier curve it is a single curve controlled by all the control points; this only a single curve. With an increase in number of control points the order of the polynomial representing the curve increases. So, if you want have more and more and more control.

So, what you have do is, you have to keep increasing the number of control points. B-spline generates a single piecewise parametric polynomial curve through any number of control points with the degree of the polynomial selected by the designer. So, you look at it you this is a B-spline curve. So, you see here it is now divided into several small small small patches. So, these are the control points, these are the control polygons.

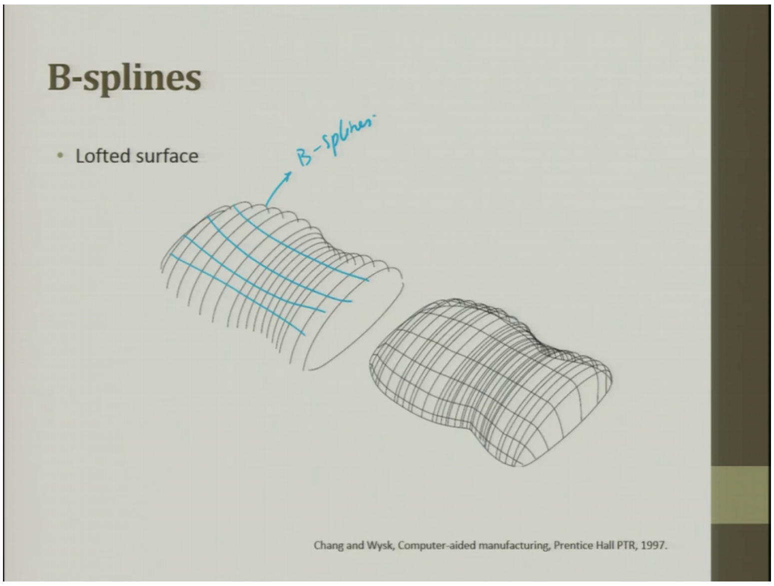
So, what was missing in Bezier could come out easily with B-spline. Why is this all important for this course is, in when you want to do anything in digital manufacturing you are suppose to draw. When you are suppose to draw you can have simple geometries free form surfaces. In free form surfaces the major thing is going to be the curve.

So, in curve if I give you the freedom of tweaking the curve to your own requirement without putting any constrain then that is what will try to be a major advantage for the designer. Why is this require? Because when you are looking for the mass customization point of view, mass customization point of view you are suppose to understand the customers requirement in terms of design and make changes in the design such that it can suit the customers need. For example, you can have a standard shoe and in that standard shoe you can have modules which are added to the shoe such that it can suit to the customer.

Currently you can see lot of people who have issues in their leg or we have movement for them what we do is, we currently may customized shoes. But these customized shoes are also not truly customized to the customer requirement. So, what they do for those people is either if there is a height variation the modules are added, these modules are then curve to the requirement.

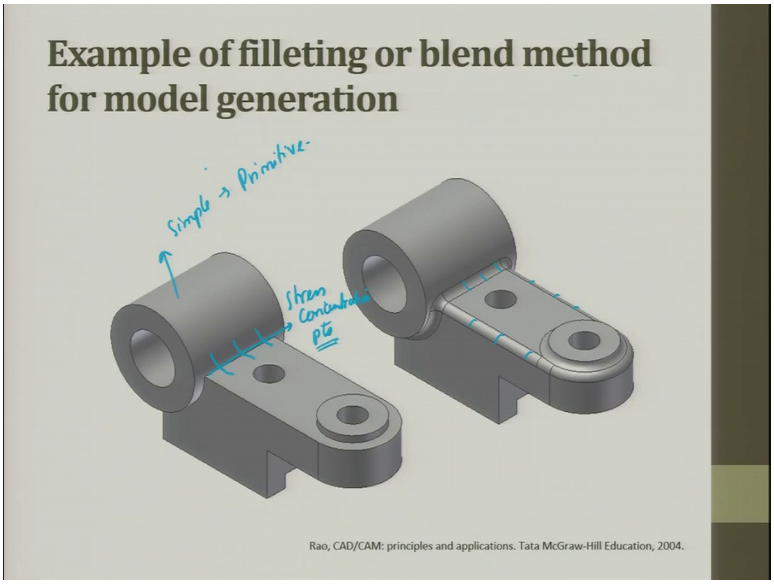
So, when you have to curve it and get this curve in done automatically then we are suppose to get this get to this freedom of using this Bezier, B-spline curves. So, B-spline generates a single piecewise parametric polynomial curve through any number of control points these are the control points with the degree of the polynomial selected by the designer.

(Refer Slide Time: 14:11)



Here is a lofted surface. So, this lofted surface these are the curves, these curves are joined together and they form a surface. So, all these curves can be drawn by using B-spline. So, where in which you had these control points what we saw in the previous figure. So, you see the control points and these control points are used to decide the curve.

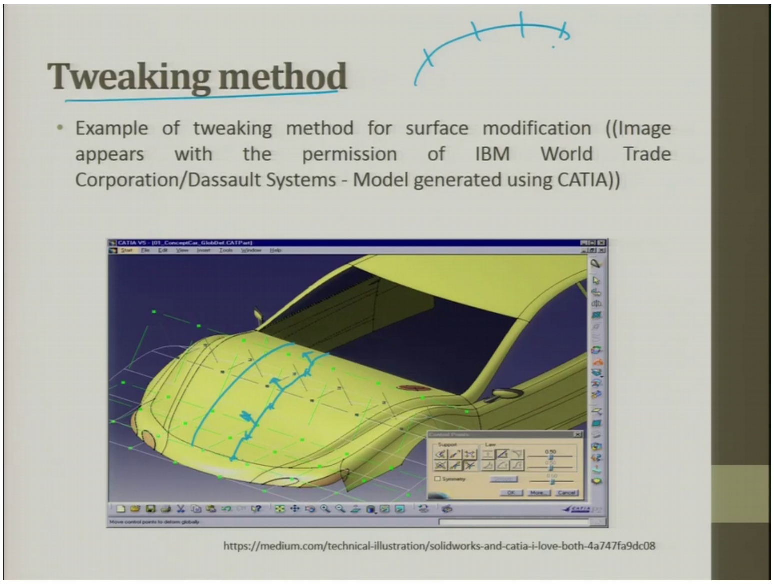
(Refer Slide Time: 14:47)



So, when we real time draw these objects, we draw this is a simple object or this you can have use it as a you can have a simple primitive is there, these primitives are chosen and these primitives are placed at locations. But when these primitives are placed and when this buts against a flat one it is always necessary to have a filleting done such that it these points are not the stress concentration points, and you can keep continuing this.

If this is a flat one with just buts against the surface then this will lead to a failure very faster. Here what we do is we have a option of filleting, you can see here wherever there is a sharp edge there is an option of filleting. Filleting is giving this will try to enhance the safety as well as the life of the component. So, these are some of the examples where in which filleting or blend methods for model generation is possible.

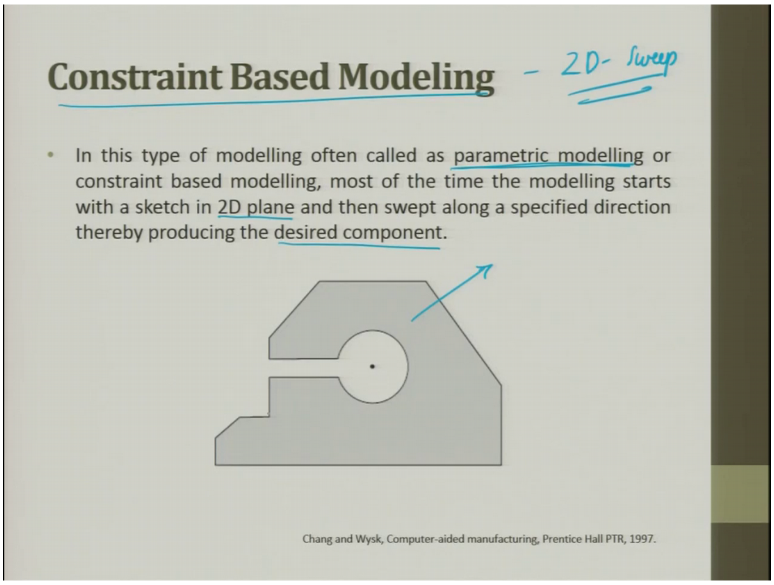
(Refer Slide Time: 15:59)



When we start talking about a real time application of a car so, the example of a tweaking method for surface modification is very much used. So, the radius need not be uniform. You can have some radius change at several points. So, if you want to do that, now what you do is you discretize them at several points and you are given the freedom of changing the or placing the control points and changing these patches ok.

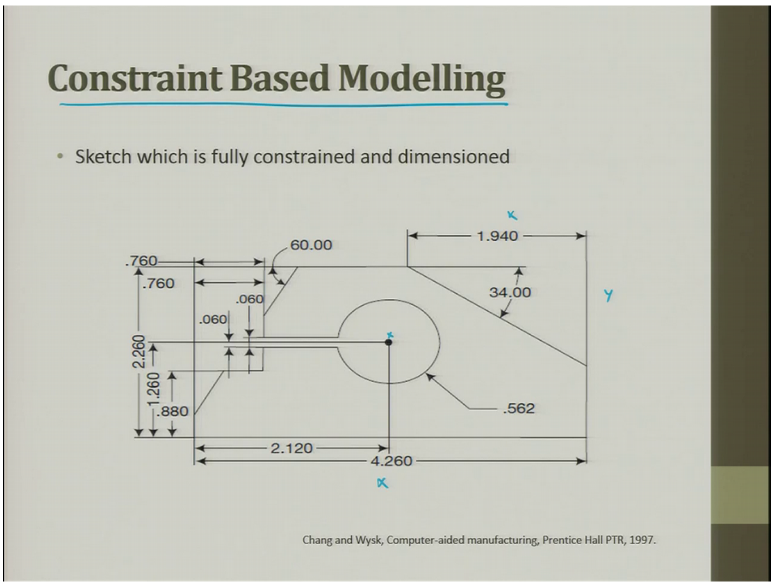
So, it will look as though it is smooth, but when you watch very closely you can see several small faces are there or curve structures because they have discretize the curve into several small control point. So, this is what is talked about in this B-spline. So, you have control points and you have control polygons.

(Refer Slide Time: 16:51)



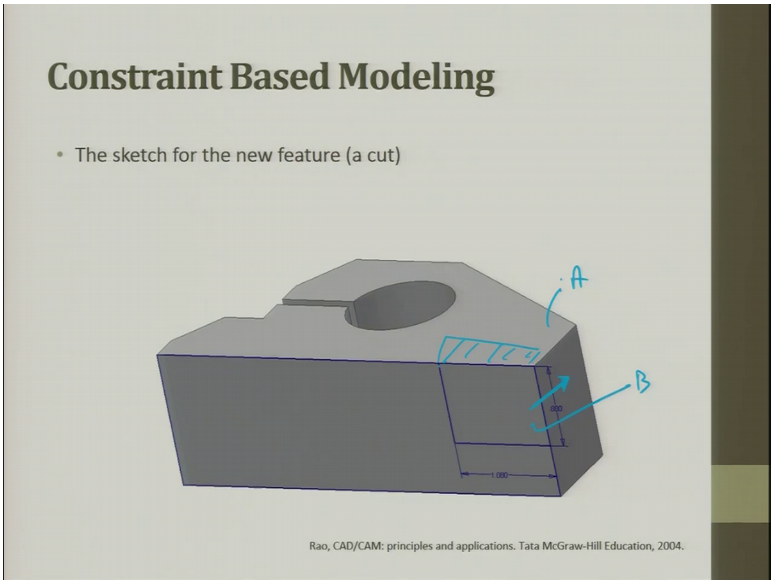
When we talk about this constraint based modeling. So, in this type of modeling often called as parametric modeling or constraint based modeling most of the time the model starts with the 2-D sketch and then swept along the direction so that you get a 3-D object. This is the same thing what we studied in the 2-D with sweep command. So, there it was a simple object so here it is a completely real time object. So, what you have done is you have constructed this 2-D and then you say desired depth which we make 3-D and then you get the component.

(Refer Slide Time: 17:27)



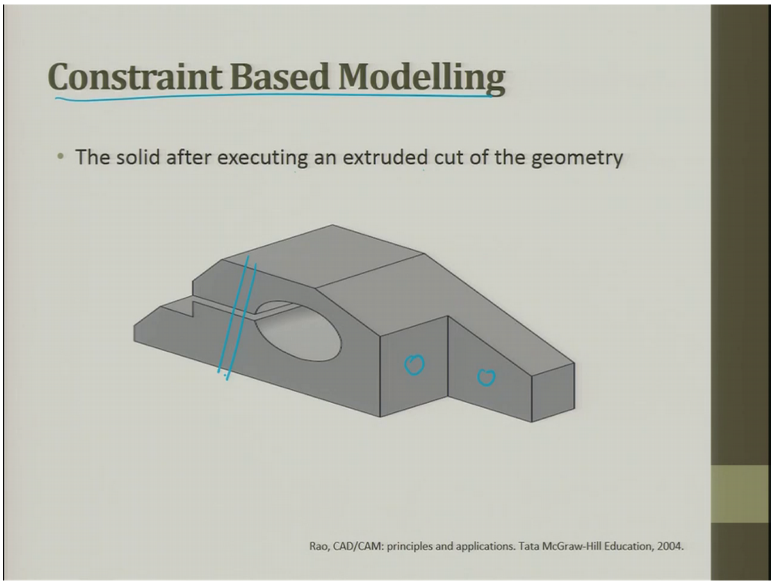
So, the constraint based modeling, the sketch which is fully constrained and dimensions are given here. So, these are the dimensions which are given. So, you can also try to establish a relationship between all these angles as well as the side edge length and you can define it as a you can library function and all you have to do is define some 3, 4 points and then moment you define this 3, 4 points the rest of the objectives getting constructed. So, in constraint based you do a 2-D and then you stretch it either in the positive direction or in the negative direction you get a complete object.

(Refer Slide Time: 18:07)



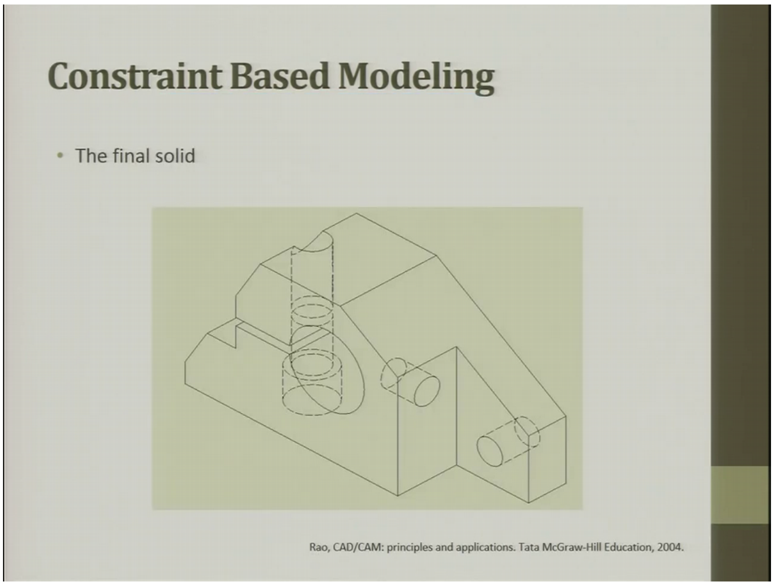
So, when sweeping along a linear path is produced the solid is made. So, in this constraint also if you want to cut and remove material so, that is also possible; for example, I do not want this portion I want this portion to be cut down, I give the 2-D profile and then I say along this direction stretch it and then do this is A this is B so, then I say a subtract B from A. So, what you get will be this profile.

(Refer Slide Time: 18:39)



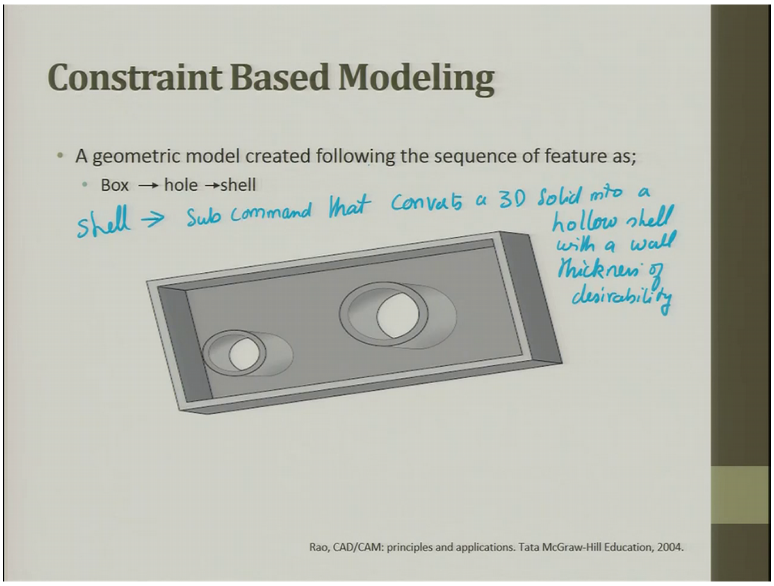
So, a solid after execution and extruded cut of the geometry you will get this. So, these all are part of constraint based modeling which is defined from 2-D and then, you start doing a 3-D operations.

(Refer Slide Time: 18:55)



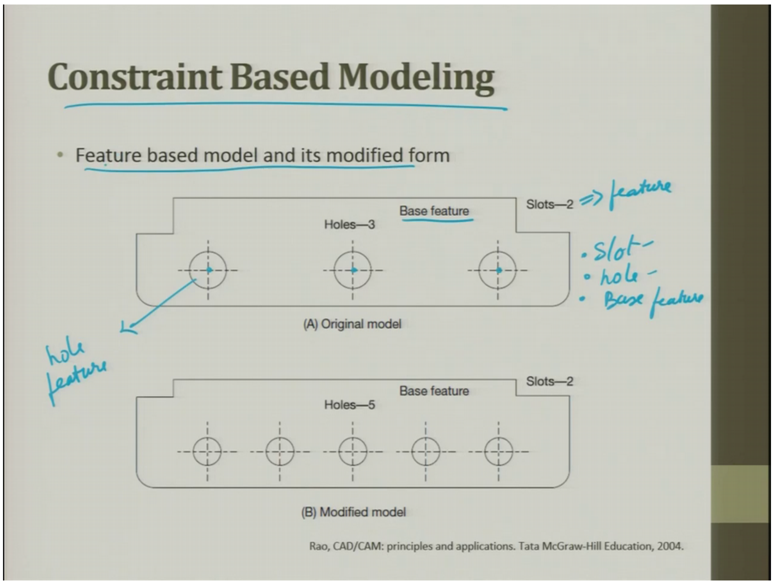
If you take an object like this and for the same object what we were we were discussing prior. So, now so here I want to make a hole, here I want to make a hole and then here I wanted to fasten. So, this object I draw all these fellows and then I say subtract these things you get to see the final object ok. The final solid is this, what you get. So, this you can see it can be quickly made through constraint based modeling.

(Refer Slide Time: 19:25)



A geometric modeling created following the sequence of features as shown here. First we create a box, then we create a hole and then we create a shell. What is shell here? Shell is a, is a sub command that converts a 3-D solid model, solid into a hollow shell with a wall thickness of desirability. Or he draws this is the shell whatever we talk about shell box then we have a hollow then we have a shell, shell is nothing but a sub command that converts a 3-D solid into a hollow shell with a wall thickness of a known wall thickness.

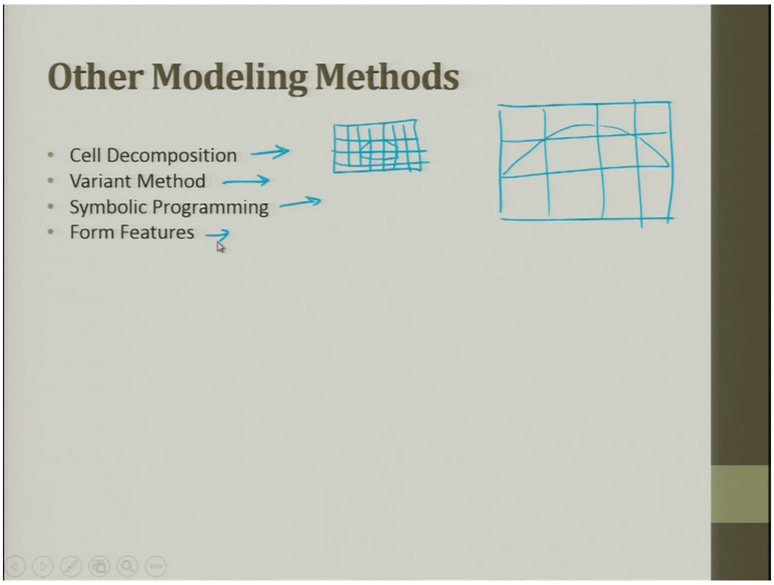
(Refer Slide Time: 20:39)



So, feature based model and it is modification feature. So, this is the base feature and these hole is another feature. All you do is you take the base feature take the hole feature and defined a locating points then you get whatever you wants. So, if you want to make a slot, slot is a third feature. You define the location for the slot you get the slots on both the ends. So, you again take the base feature and then if you want to make holes you try to take 5 holes, you put one after the other after the other and 2 slots on the both sides you try to get whatever you want.

So now, the feature of slot is known to me, the feature of hole is known to me, the base feature is known to me, I only put the location for these features in the drawing such that I get whatever I want ok. This is again a constraint based modeling and here it is feature based modeling and it is modified form. Why is this important? Because I know to create a slot, I know to create a hole right. So, I, when I say I know to create a hole the I know the list of process parameters, I know what is the machine to use, what is a process parameter to use, what are the toolings to be use such that I get the output.

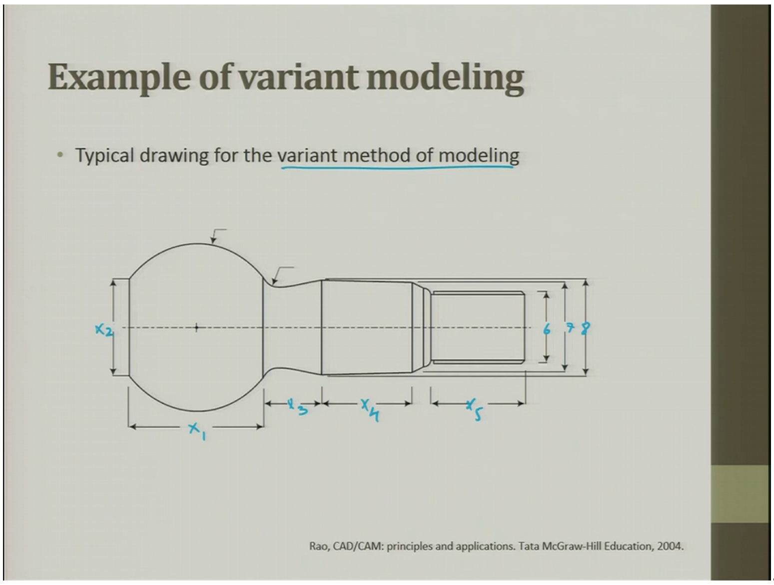
(Refer Slide Time: 22:31)



So, now when I say hole as a feature; so, now immediately when I draw the CAD for the CAM it is easy for bringing in all the manufacturing processes attached to it and align to it ok. So, we have here in this particular case, we have distinguish different features and linked with these features to the manufacturing process. The other modeling methods are cell decomposition, we will see little later.

But however, when we say cell composition what we do is we draw the entire cell or it is like a graph paper we draw entire graph paper and on the graph paper we start drawing an object ok. And here the object is decomposed into several cells and then we know what are all the cells to draw. So, if you look at very good artist what they do is they always draw a grid pattern in their in their artistic paper and then they start linking it. In the generating the feature whatever it is. Then we will also see what is variant type symbolic forming and symbolic programming and form features.

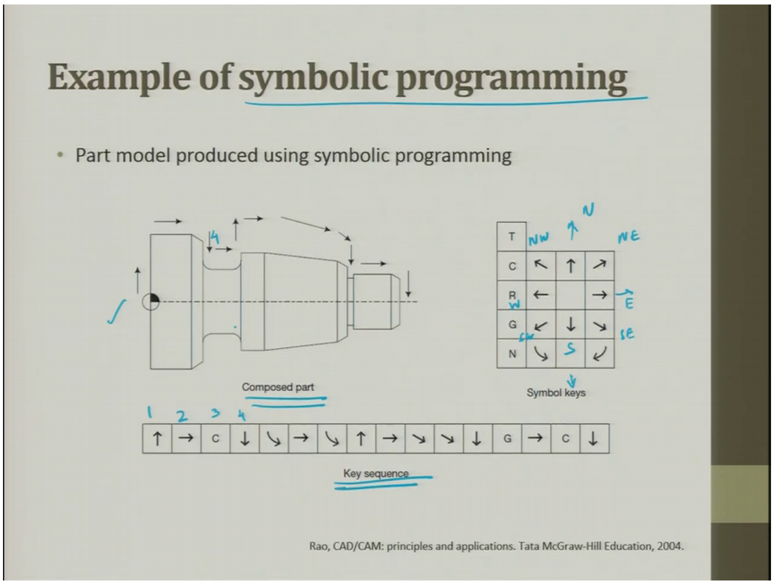
(Refer Slide Time: 23:19)



So, this is a variant method of modeling. So, here in which we define x 1, x 2, x 3, x 4, x 5, 6, 7 and 8 and now this x 1, x 2, x 3, x 3 all these things are having some relationship and already this relationship is established. So, all you can do is just drop in these values and try to get different different shapes and size of this same components. So, that is a variant method so we just put these things to the requirement

And again as in manufacturing in cooking also we can when we scale up the processes also change. In manufacturing also when we as and when we scale up from 10 part to 20 parts, 100 parts, 1000 parts the manufacturing process also changes. So, that is a separate domain, but we are not getting into the domain, we are talking only about CAD. So, CAD there is a relationship clearly established between these variables and when you call this product or component comes in front of you just define it and then get the part. So, this is a variant method of modeling.

(Refer Slide Time: 24:37)

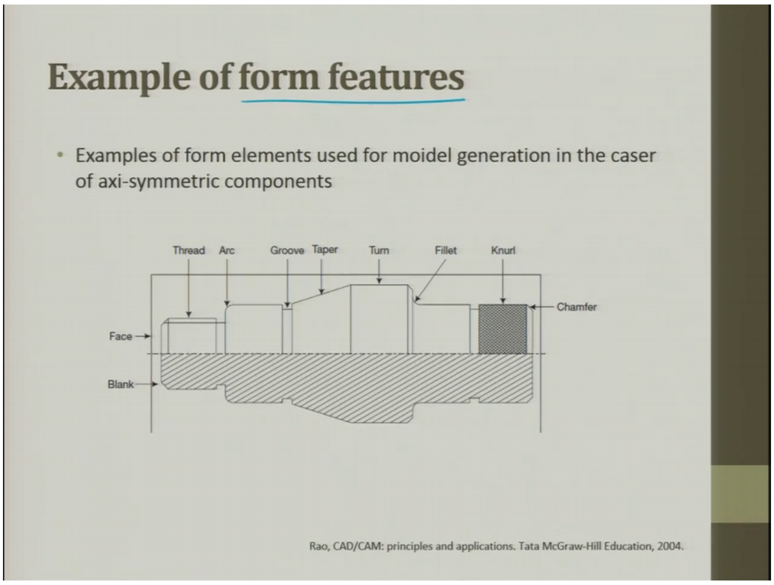


The other one is symbolic programming, we use symbols it is like trying to tell a blind man how do you walk and then walk and then generate a path for him so that he regularly follows. So, you look at the component, the component is this for which the component is divided is divided into several lines, arcs and arcs curves. So, now, these curves had to be generated one after the other after the other.

So, that is what is done using the symbols here. So, we use top the bottom, left, right ok. And then between top and or you can say north south and then you call it as east and west. So, this is northwest, northeast, southeast, southwest right and now what you do is you try to use those buttons and then start defining a sequence. So, here the sequence is define, the component is made the dimensions are not talked about here. The dimensions will be talk next ok.

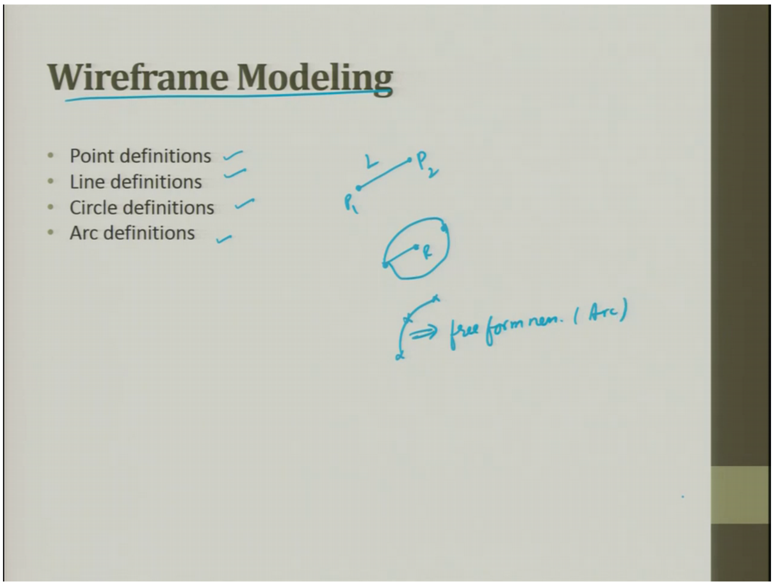
So, here if you see first path it most from here this is first then this goes and then it goes for chamfer 3 and then it goes for a chamfer then it goes down. So, this is 4 and then it is 5 so you can keep on adding the sequence. This is the key sequence to generate the process these are the symbols with these symbols you can try to draw this component and this component you can start feeding the dimensions and get the data.

(Refer Slide Time: 26:21)



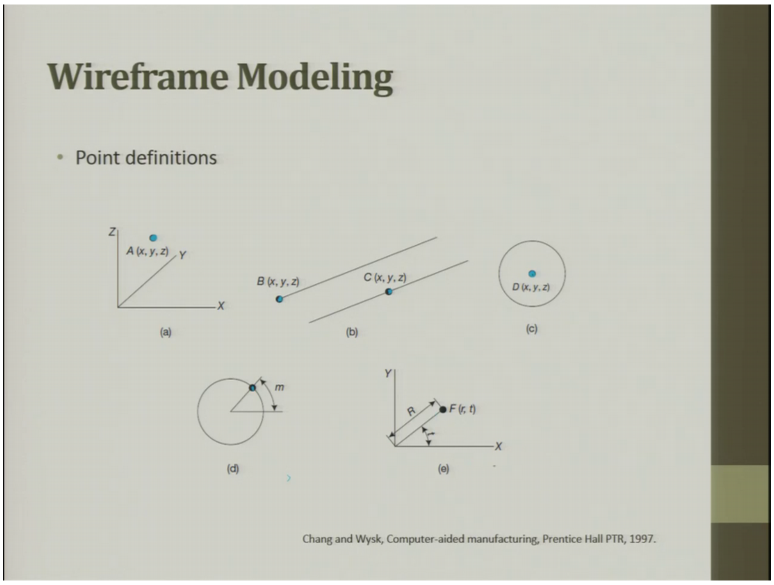
The examples of form features, this is an examples, so this is a feature from this feature what are all the features from the blank to what are all the features you want to make you can try to draw it and then try to store it.

(Refer Slide Time: 26:41)



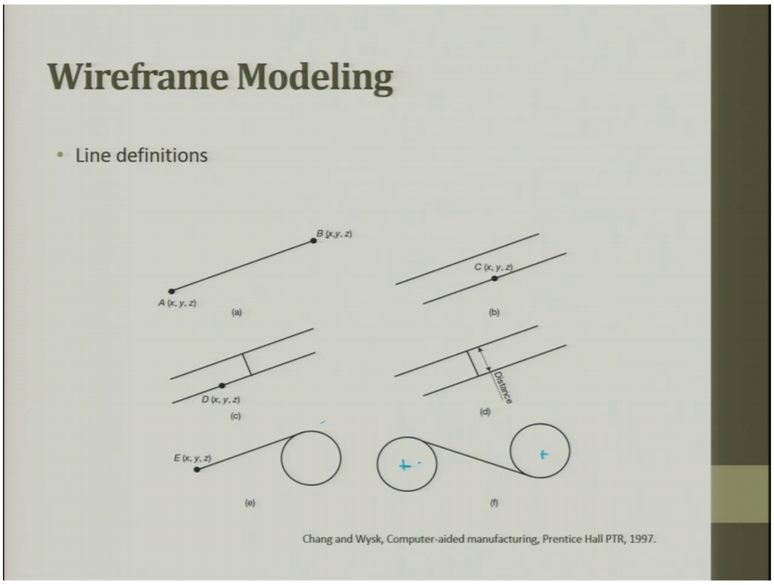
So, when you move to wireframe, though we know wireframe does not give you unique representation. But wireframe is exhaustively used because it occupies a small space. And when we talk about wireframe we will first always start defining a point, point to a line, line to a circle and then arc. So, if I know to draw a point good, if I know to draw 2 points it is excellent, then I can joint these 2 points by a line, so point 1, 2 then it becomes a line, then if I want to make a little more difficult. So, point, point and point so I make a circle ok. This is the radius or the centre point ok, and then the next one is the arc. So, arc is important because here I am given you the free form, free formness. So, if I know to define a point, a line, a circle and a free form arc ok. So, then I can draw any complex objects. So, in wireframe we will see how do their represent these to these parameters or these entities drawing entities.

(Refer Slide Time: 27:49)



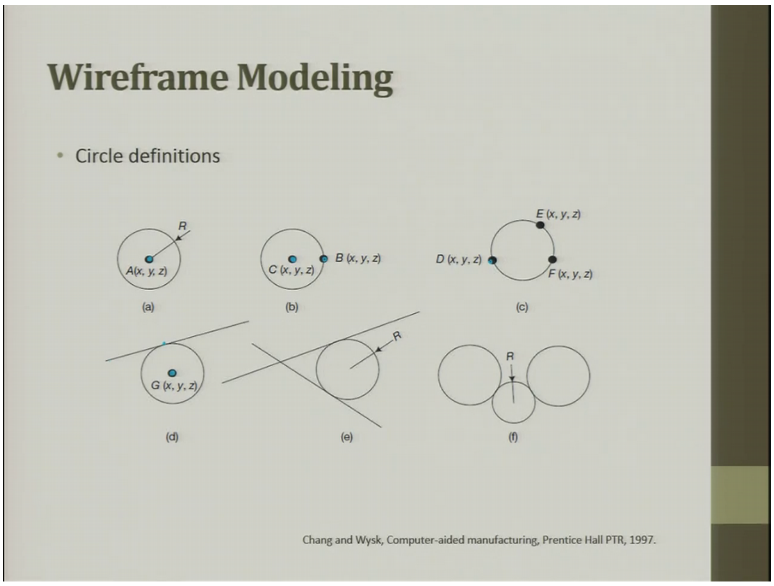
So, point you can define a point a x comma, y comma, z you can define line and then you can say start point and to mid point this and this is for a circle, this is for sector you want to do and point drawn at an angle to x axis So, these are different ways you define a point.

(Refer Slide Time: 28:17)



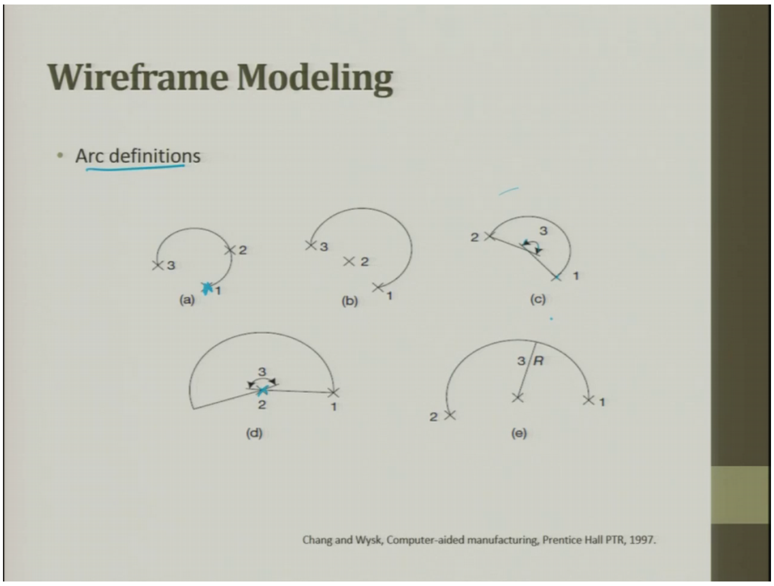
And these are different ways you define a line point and point joint a A and B, you get a line. So, you draw a parallel to this line with an off set of this also you draw a line and save a perpendicular to this you can draw you just defined a distance and then you can also try to draw a line which is a tangent to the circle. So, for a single circle at tangent between 2 circles ok; so, if you look at delts fully mechanism you have crisscross were the tension is maintained.

(Refer Slide Time: 28:51)



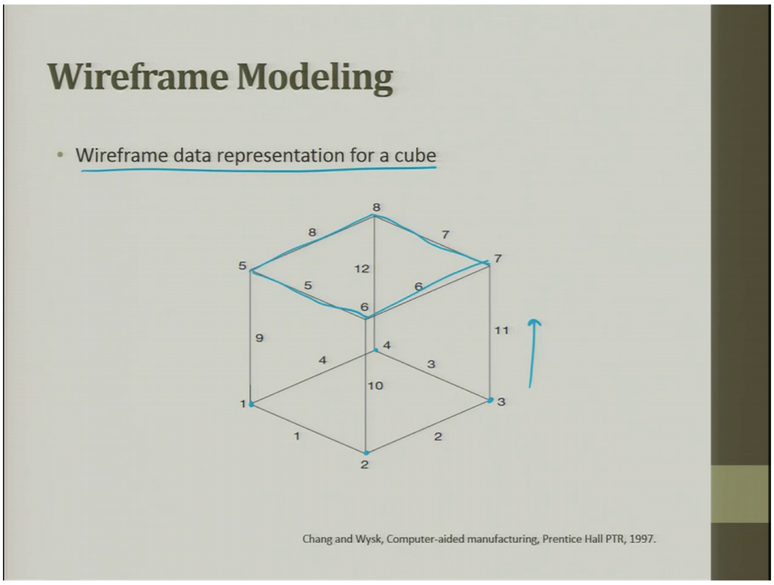
The other one is, you can define circles by giving centre point and radius. So, centre point and a point here and then you can define 3 centre points, you can define a centre point and the tangent 2, you can define 2 lines and then tangent 2 which is intersecting so, you can also define that arc and then you can also define with the other circles and then 2 circles joining these 2 circles here radius. So, you can define a circle. These are all different ways of defining a circle.

(Refer Slide Time: 29:19)



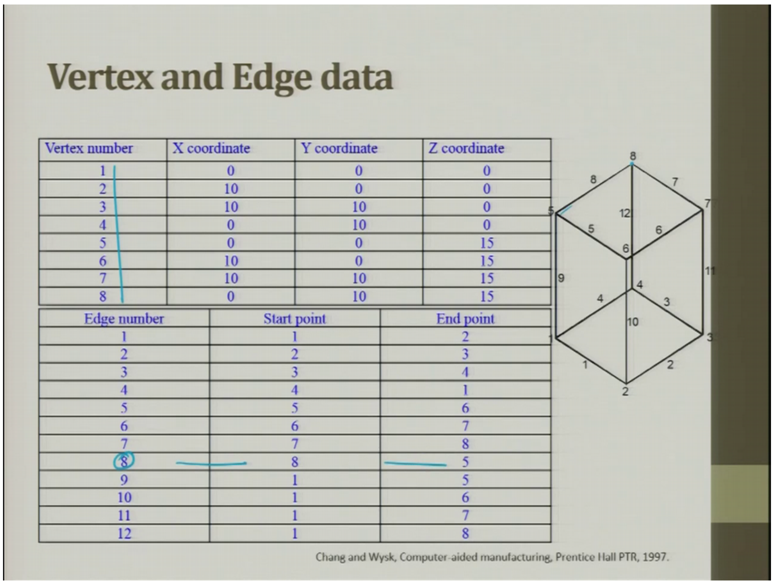
So, then let us define an arc. So, arc I have already told you start point, end point, mid points, start point, end point and then you can fix the centre, you can also try to say start point, end point and the angle of including and this is start point, end point where I have said the start point, centre point and the angle included or I can say start point and end point and the radius. So, these are different ways of drawing an arc.

(Refer Slide Time: 29:49)



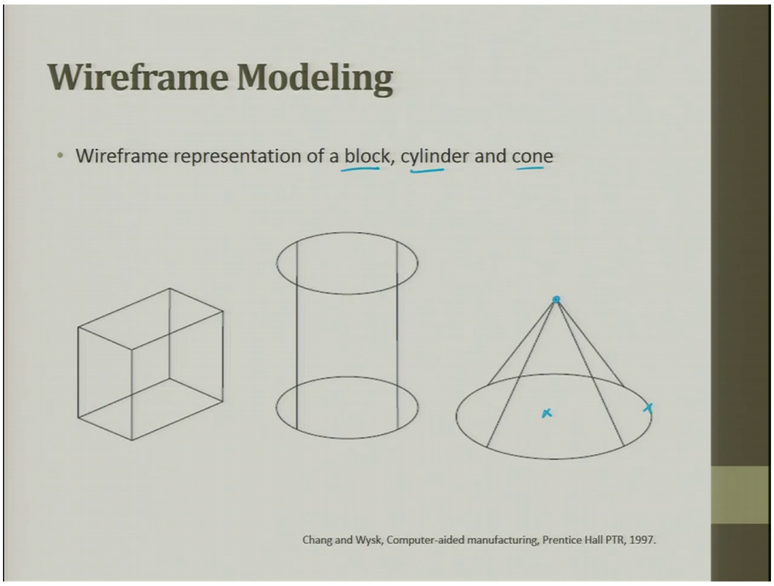
In a wireframe representation if we try to draw a cube you would like to see how does this data getting represented? So, here the edges are these are the vertices 1, 2, 3, 4 and you just off set this plain you get 4, 6, 7, 8 ok. Now you this is 5, 6, 7, 8 you have drawn this. These are the edges and now if you want to draw a line connecting that, so 1 and 2 is connected by 1, 2 and 3 by 2, 3 and 4 by 3, 4 and 1 by 4 and in the same way 5 and 6 by 5 you are drawing a line. So, this when you look from 3-D prospective it becomes an edge. 6 and 7 it is 6, 7 and 8 it is 7, 8, 5 and 5 it is 8 ok. Now, these 2 planes are hanging in free air. You wanted a cube; cube has to be linked between these two fellows. So, then what you do you draw start dropping a line from each vertices so I get 9, 10, 11 and 12 ok.

(Refer Slide Time: 31:01)



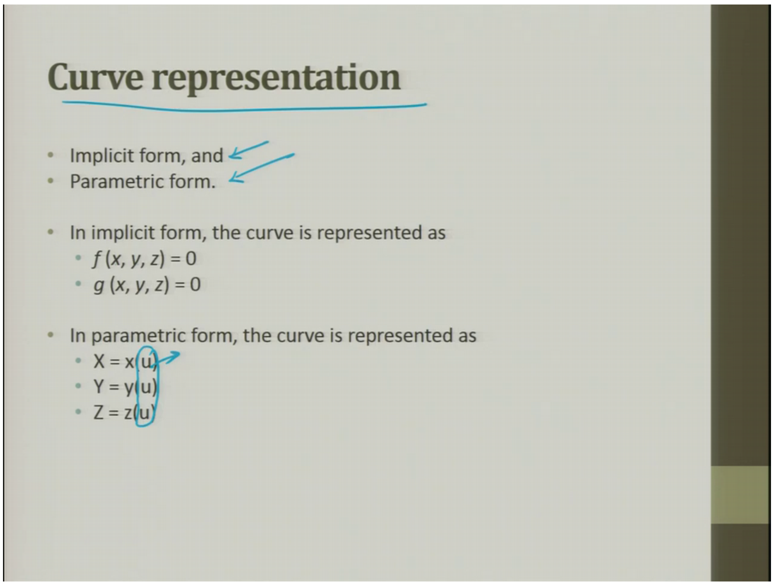
If you want to represent this data in computer this is how it is represented. So, vertices 1, 2, 3, 4 what are all the X, Y , Z. Then edge numbers, edge numbers edge number 1, what is the starting point is 1 and the ending point is 2. So, here because you have defined all the vertices you do not have to define once again the vertices here. So, what you do is you just say edge is drawn connecting 1 and 2. For example, let us take 8 edge number 8, edge number 8 is this. So, it is trying to connect vertices 8 with vertices 5. Bertices 8 with vertices 5 ok. This is how a wireframe data is stored in a computer.

(Refer Slide Time: 31:43)



So, all these things cube versus is a simple 3-D object, but what is occurs lines and all those things are small basic entities which can be use to develop 3-D model. In wireframe representation you have block which is a cube example we saw, we can also have a cylinder and a cone. Suppose you want to define a cone we do this centre point radius and then we also tell the height. So, you can try to get a cone.

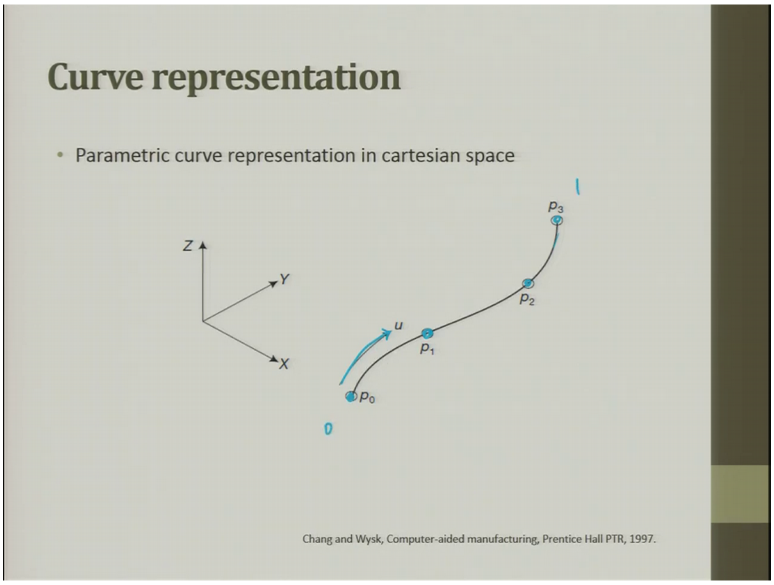
(Refer Slide Time: 32:13)



So, when we talk about curves there are two ways of representing the curves. So, one is the implicit form the other one is the parametric form. In implicit form if the curve is represented as f is a function of x, y, z which is equal to 0. g is a function x, y, z is equal to 0, but we do not have a linking thing between these 2.

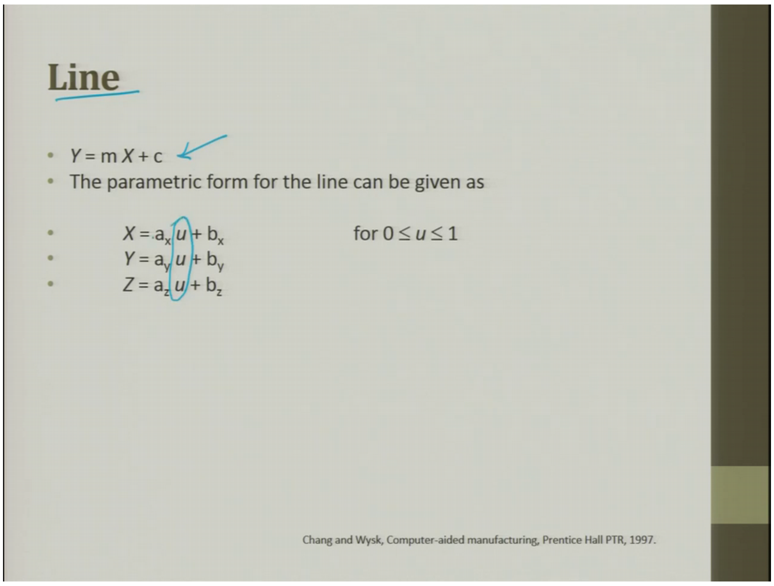
So, we felt that if we can define if we can define f and g with minimum number of points it will be good and if I can also connect f and g through one more connect these two intro connect with a variable then that is much more easier for me to operate so I do not have to define so, many points. So, in that case the parametric form gave me an advantage. In parametric form the curve is represented as x equal to x of u, y equal to y of u, z equal to so, you see here all these things are linked by a terminology or by a factor called u.

(Refer Slide Time: 33:13)



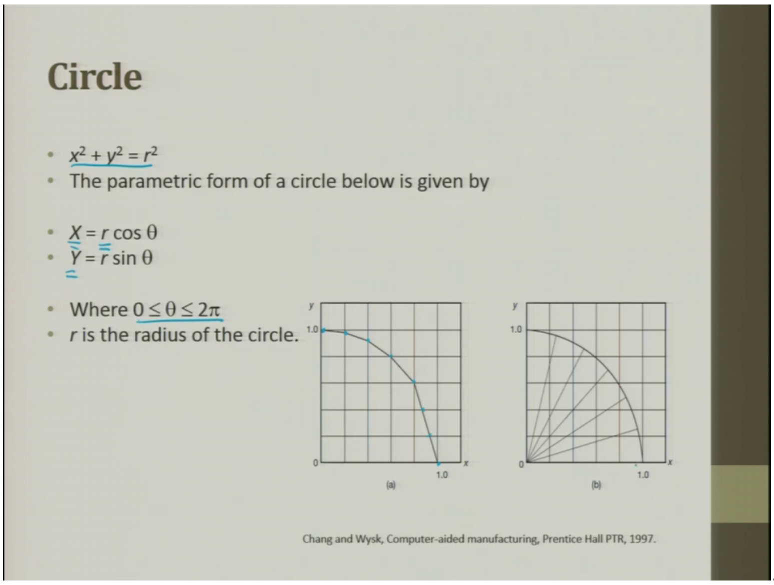
All I have to do is define u which takes a value 0 to 1 I can get whatever output I want. So, here is a curve. So, I have define point start point, end point, point P 1 and P 2. Here is the u vector which is going. So, I can take a value from 0 to 1 anywhere and I can start drawing the curve.

(Refer Slide Time: 33:33)



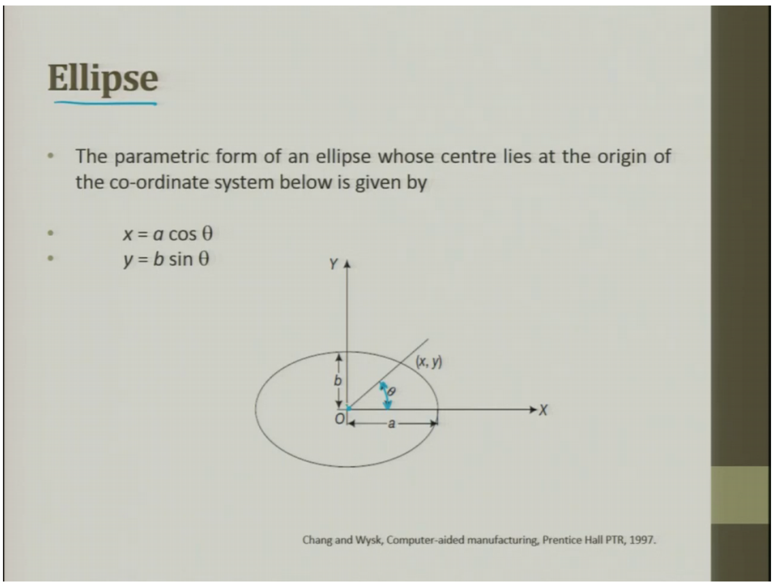
So, if you want to draw represent line, which is this is a simple equation in a parametric form you represented as X, Y and Z. So now you see u takes a value between 0 to 1.

(Refer Slide Time: 33:47)



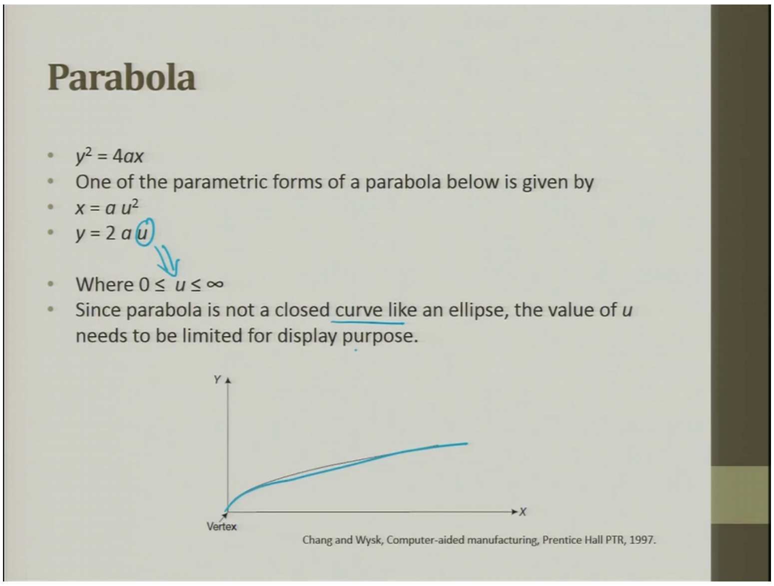
So, when we have to represent a simple circle it is x square plus y square is equal to r square. In the parametric form I represent it as x equal to r cos theta y equal to r sin theta. Where theta takes a value between 0 to 2 pi, r is a radius. So, now, you see and the parametric form I am representing a circle. So, this is what it is. So, 1 and 1 and these are what it is and if you want to see these are facets which I have drawn. So, these are all facets know the arc is divided into several facets. So, this is a curve which is getting done.

(Refer Slide Time: 34:25)



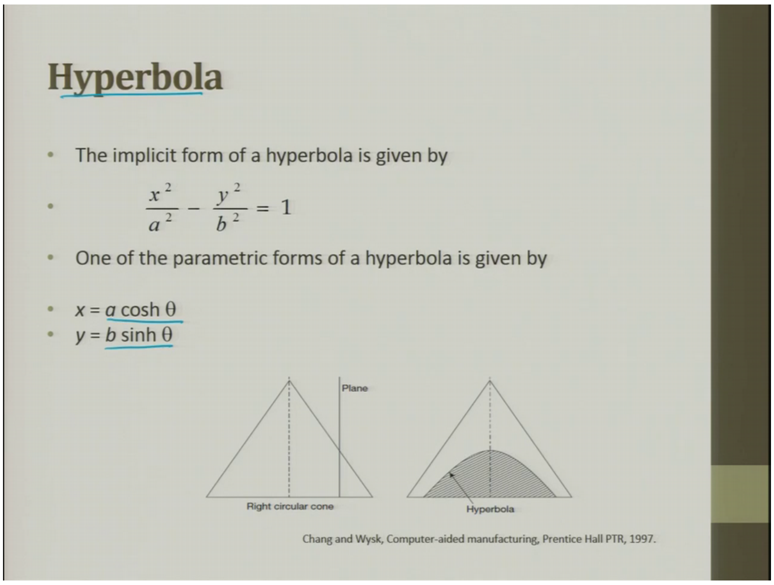
So, Ellipse the parametric form of an ellipse whose centre lies at the origin of the coordinate system below is given by x equal to a cos alpha phi and y equal to b sin phi. So, this is phi a and this is b. So, this is a typical ellipse. So, you have x and y.

(Refer Slide Time: 34:53)



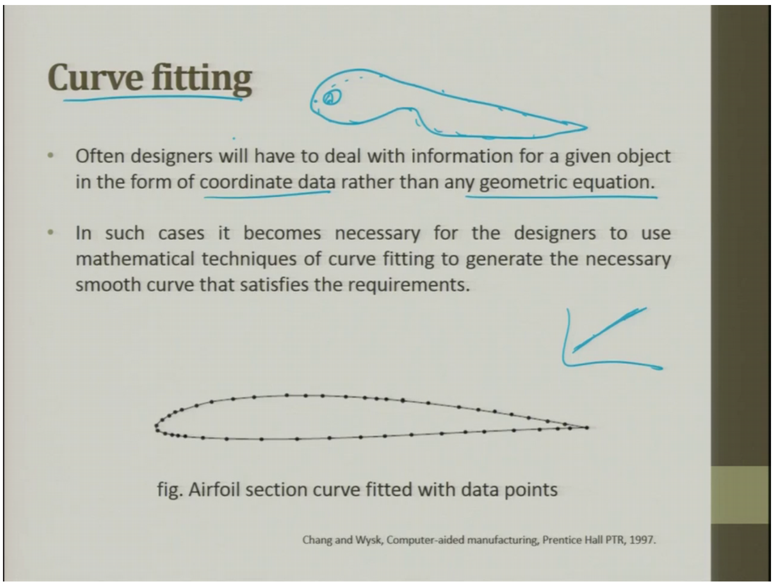
So, Parabola; parabola is y square equal to 4 a x, one of the parametric form of the parabola is given by x equal to a u square and y equal to 2 a u where u takes a value this is very important 0 to infinity. Since the parabola is not closed so that is why we go to infinity since the parabola is not a closed curve like an ellipse the value of u needs to be limited for the display purpose.

(Refer Slide Time: 35:23)



Then we want to do a Hyperbola, the implicit equation for a hyperbola is x square by a square minus y square by b square equal to 1. I am representing it into a parametric form, I represented as a cosh theta and b sin h theta ok. And so here theta also tries to take the value like value like this. Whatever is use to an ellipse but all these things are simple curves, but what we really want in our real time usages free form curves. So, now, the apart from all these curves there is a need for curve fitting also to happen.

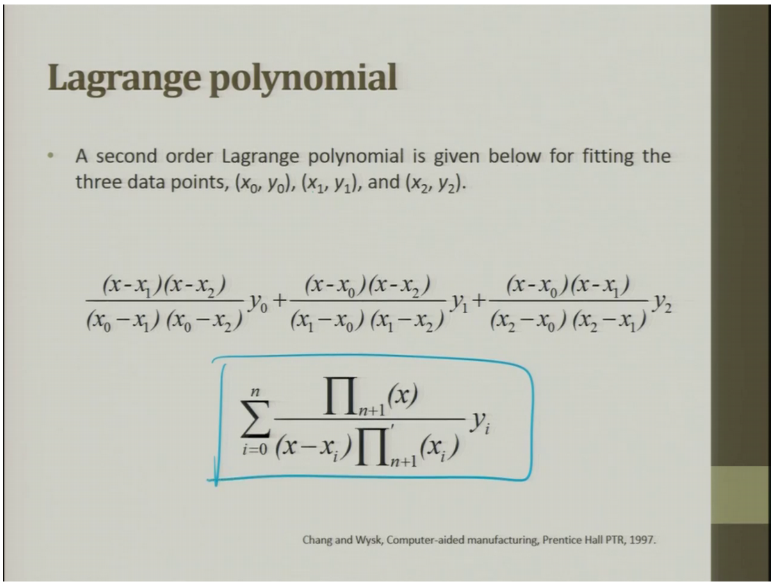
(Refer Slide Time: 35:49)



So, next we will see what is curve fitting. Often designers will have to deal with information's of a given object in the form of coordinate data rather than geometric equation. Because we touch surfaces and we get the data. Something as so, you see here this gets developed ok. So, here what we do is we cannot represent this form which is a bio mimicking form, in a geometric form we always try to take several of these coordinate points convert this coordinates and then get the value.

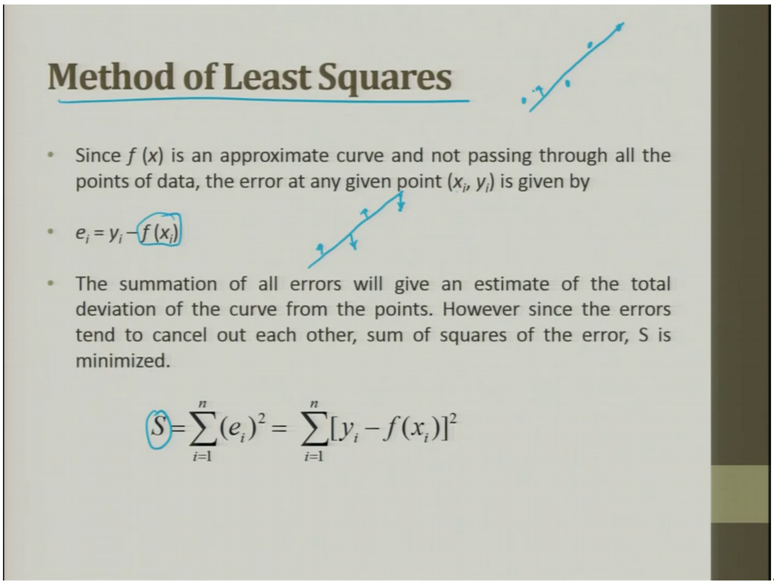
So, in such cases it is necessary for the designer to use a mathematical technique of curve fitting. What is curve fitting? You have several points pass the curve through as sets we are covering maximum number of points are closer to maximum number of points. So, the curve fitting to generate the necessary smooth curve that that satisfies the requirement ok. So, curve fitting is something which is very important. So, Airfoil curve fits with the data point.

(Refer Slide Time: 37:15)



So, here we use Lagrangian polymer a second order Lagrangian polymer is given is given below for fitting the 3 data points x, x naught x 1, x 2. So, this can be represented by this on fine tuning this, this can be represented in this form.

(Refer Slide Time: 37:33)

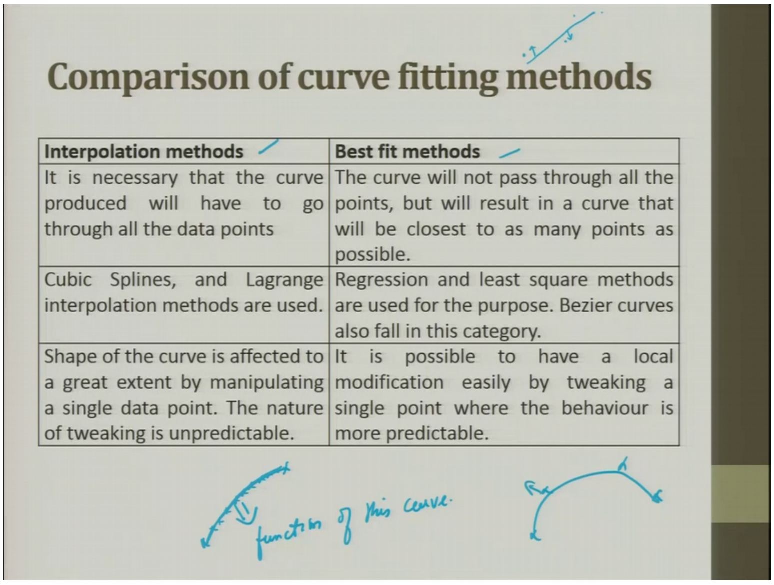


So, the method of least square is also used in order to have a best curve fit. So, since f x is a is an approximate curve and it is not passing through all the points of data the error at each at any given point x i, y i is given by error is nothing but y i minus a function of x I. So for example, you had you have several of these points and then you have a line passing through now you wanted to measure this deviation which is the error.

Ok since f x is an approximate curve and not passing through all the points of data the error at any given point x i y i is given by e i is equal to y i minus a function of x i. the summation of all the errors will give an estimate of the total deviation of the curve from the points. However, since the error tends to cancel out each other sum of these squares of errors S is should be minimum. So, this error deviation should be minimum.

Suppose you have a set of data points like this and you are trying to draw a line so now, what I am trying to a line or a curve I am trying to shift this curve such that I try to have minimum error between the data points.

(Refer Slide Time: 39:01)



So, in curve fitting method we have 2 methods; 1 is interpolation another one is the best fit method. Interpolation is I know start point, I know end point, I know a curve and I know the function of this curve ok. So, in interpolation what we do is we discretize into several small points and these points are the curve, these points are discretizing the curve for which you have written an equation.

So, it is necessary that the curve produced will have to go through all the data points. Cubic spline and Lagrangian interpolation methods are used in interpolation. Shape of the curve is affected to a great extent by manipulating a single data point the nature of tweaking is unpredictable. So, these are the points for interpolation.

When you go to best fit curves the best fill will not pass through all the points which I told you, data point curve. So, this fellow will get mode up or down to get the best in the curves will not pass through all the points but will result in a curve that will be closest to as many point that is best fit method. Regression and a least square are 2 different techniques which are different different techniques which are used to get the best fit curve.

Bezier curve, B-spline we studied Bezier, Bezier curve falls in this category. So, what is Bezier curves? Start point, end point, control points there is a best fit curve. So, falls it is possible to have local modification easily by tweaking a single point where the behaviour is more predictable. That is the advantage of best fit. So, we would stop here we will continue in the next class.

Thank you very much.