# Theory of Mechanisms

## Lecture 5

## Linkage Synthesis Classification

## 2-position Motion Generation

So, when we talk about synthesis.

(Refer Slide Time: 00:14)

## Mechanism synthesis

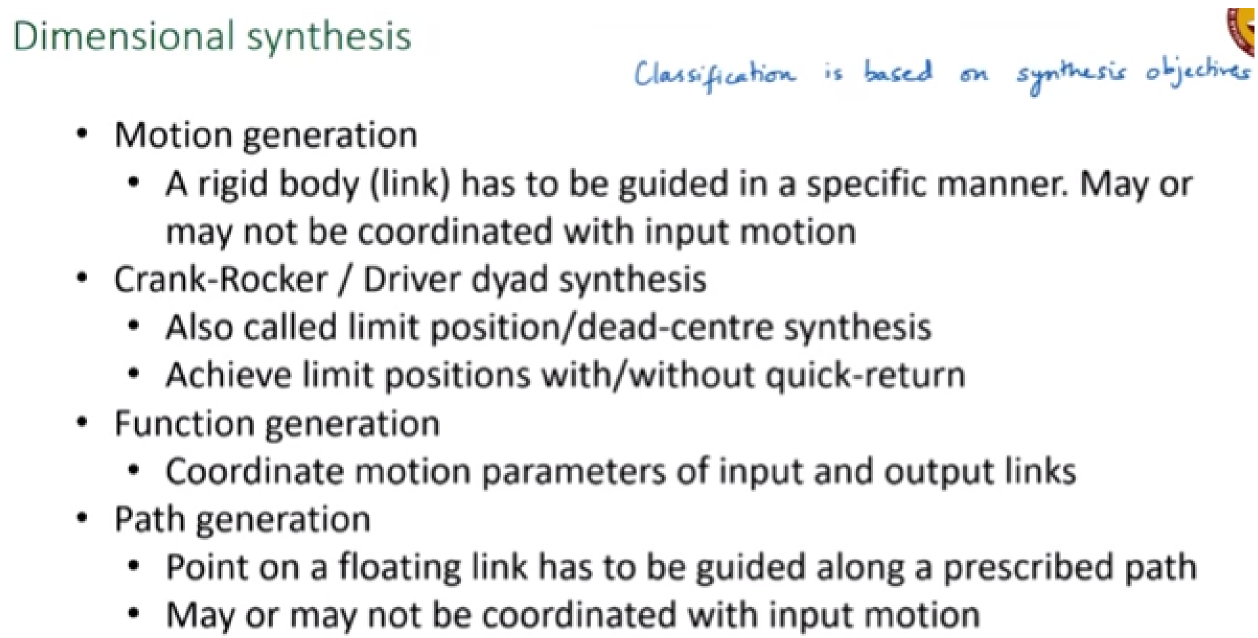
- Type synthesis
  - Determine mechanism best suited to the problem
- Number synthesis
  - Once a linkage has been chosen as the optimal type of mechanism
  - Determination of the number and order of links and joints to achieve the motion with required mobility
- Dimensional synthesis

- Dimensional synthesis
  - Determine link lengths for the specific motion characteristics required

Okay, we start looking at it from the you know the bird's eye view you look at from the top you know for a particular problem and this is something if you happen to end up in a core engineering job you might, you might have a task for which you need to design, you know some task of motion transformation, for which you may need to design a mechanism. So the first thing you're going to look at is, what kind of a mechanism do I want for this, okay it could be a linkage, it could be something that uses robots, it could be something that uses like a conveyor, it could be something that uses like a chain, a pulley rope Makin, you have a wide variety of machine elements to put together to form mechanisms, okay so that forms the basis for type synthesis and it's something that comes through experience, so you would have seen similar applications or you know you would, so what you would do at that point is basically research similar applications see what kinds of mechanisms are used for that and then try to narrow down to the type of mechanism you want to use for the application.

There may be cost considerations, space considerations, you know maintenance considerations, whole bunch of factors that will play into what you determine as the type of mechanism you want to use for your application and a lot of that basically comes from experience, so here we are not really going to deal with that aspect of it, in this particular course our understanding is that okay, I want a linkage for my I think a linkage would be best suited for the applications that I'm looking at and so the first thing that we do when you look at ,if you want to look at a linkage as your solution is to do the number synthesis which is what we looked at earlier ,which is basically determining the number and order of links and joints, which you wanted used to achieve the mechanism of required mobility, so again that is something we looked at in the last couple of classes. Now we come to the crux of this course which would be the dimensional synthesis, so once you've determined that a linkage is what you want you know what sort of job, you know what mobility you want for your mechanism, you know in many cases it may be a four bar or a slider crank or some combination of that now what you want is to determine the dimensions of the links in order to solve a particular problem, in order to achieve the particular task that you are looking at ,so that is dimensional synthesis, so you have specific motion characteristics that is what you are designing for and you want to determine what should be the link lengths for that particular linkage that you're going to design.

(Refer Slide Time: 04:19)




So dimensional synthesis can be further classified into the following you have what is known as motion generation. So, in motion generation you want to guide a rigid body, through a certain number of specified positions okay, so here we are talking about both the position and orientation of that rigid body, you want the rigid body in the plane located at a certain point and oriented in a certain way and he wanted to go through a series of those positions and you may or may not want to coordinate it with the input motion, so these types of problems are called motion generation problems ,then you have what are known as the crank-rocker diet synthesis driver diet synthesis problems, where you have the mechanism, so in many cases here we are talking about will first look at a four bar okay, we will first look at a four bar, so four bar can be designed for motion generation a four Bar can be designed for you know as a Crank-Crocker obviously we have seen the Grashoff criterion, so this is also called the limit position or dead center Synthesis. Where you want the output to attain to limit positions, to move between two limit, limiting positions okay. So you want the motion to be restricted, the input motion may be continuous okay, in the case of the crank-rocker synthesis but for that input motion you want the output to be limited to two extreme positions and this could be attained with or without quick return ,so you could say that when it goes from position a to position B it takes a certain amount of time and then from position B to position a maybe faster, so in many cases, if you have a stroke where you are doing some work, say you're pushing something, or you are cutting something, then you want a slower speed ,but for it to again do that job again you want it to return quickly, so you may actually have a quick return or a non quick return kind of mechanism in these types.

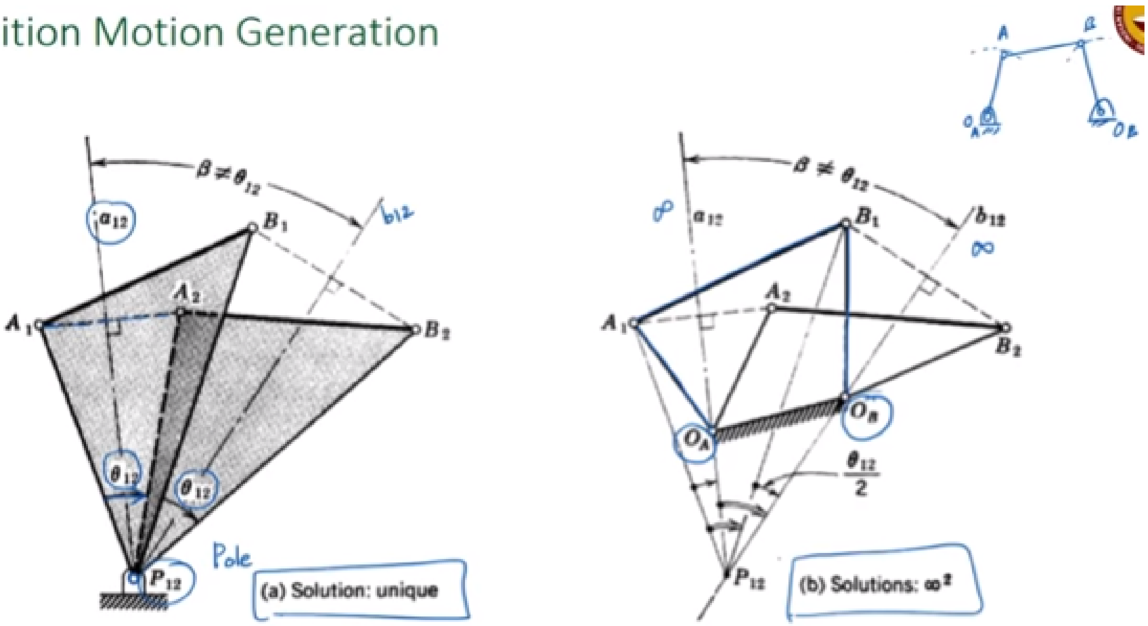
The third type of synthesis we'll look at the problems that we look at will be the function generation problems, where you are actually coordinating again the motion, motion parameters of the input and output link it may be angular displacement of input linked with that of the output link ,essentially the previous one is also an example of function generation your crank-Crocker, you're coordinating the motion of the crank to the angle of the rocket or the displacement of the, the stroke of the slider crack so those are also function generation problems, but a special type of function generation problem, so in this you're essentially the motion parameters could be apart from displacement it could also be velocity and acceleration, so you may want the in for a certain velocity of the input link you want a certain velocity of the output link similarly with the accelerations, so we will see some methods of synthesis that will allow you to do that.

So that category of problems are called function generation problems, the fourth category is where you have a point on the floating link which is the coupler, which follows a certain prescribed path okay ,you want it to follow a certain prescribed path you may recall the example I showed you of that filmstrip mechanism, where you had a point on the coupler which followed a certain path and you may want to design a mechanic to do something like that okay, so there the objective was the path of that coupler point, so if that is your design objective then we classify the problem as a path generation problem, now this classification yes motion generation you're looking at both the position and the orientation of the link in the plane, in the path generation you're looking at only the path of a point on the coupler okay, so that's the difference between the two.

Now a four ball doesn't really care what kind of problem you are solving okay ,you may have all of these you can design afford bat to achieve any of these tasks, the four bars behavior is in you know it's, it's going to behave as a four bar it has a certain relationship between the links, its coupler points move in certain ways all that okay, the classification is only based on the problem that you are trying to solve okay, so in most cases we are designing a four bar and the four bar behaves the way it does, we are trying, we are trying to narrow down which characteristic of the four bar we want for our particular problem okay, so that is the classification, so it may be the same linkage that we'll learn so any four bar irrespective of what you design it for will have dead centers, will have limit positions right, there will be some relationship between the input and the output motion, the coupler will attain certain positions in the positions and orientations as the four bar moves okay, and the coupler points will trace certain paths okay ,what you design the four buffer will depend on your problem classification so this classification is actually based on what you want for your application so that is the synthesis objective. So this classification is based on your synthesis objective, this classification is based on objectives okay.

Refer slide time: (11:31)

## 2-position Motion Generation




Okay, so let's look at, we'll start off with to position motion generation okay, so I have this, I have a rigid body a link which I want to move from a 1 B 1 to a 2 B 2 ok, that's my design objective, I wanted to attain this position first and then move it to this position, so essentially, I want to so if I look at these two positions of a rigid body okay, I can actually identify a point on the plane for which that position does not change for the two positions of the rigid body okay, so if I extend these bodies okay, then I find that so how do I do that okay. Say I want to move this in a, do you remember castle's theorem any planar motion can be decomposed into a pure rotation right.

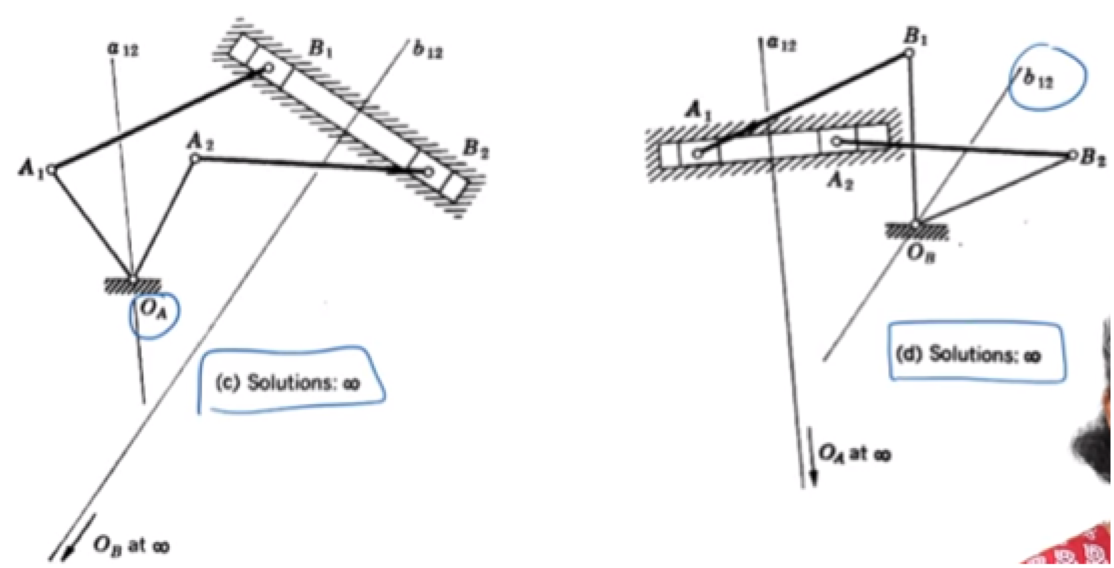
So in this case if I want to try to do that, say I draw thee let me do it here itself I think I have another figure so if I look at a 1 to a 2 I draw the perpendicular bisector, so this is the perpendicular bisector a 1 to a 2, so I know that if I pick a point on that that's going to be equidistant from a 1 and a 2 similarly I do that for B 1 and B 2 and the point where these two perpendicular bisectors enter, intersect so this is let's make all this B 1 to the perpendicular bisector of B 1 and B 2, the point at which these two intersect is called the pole for these two positions okay and what that tells me is suppose I extend this body okay ,I make this a triangle a 1 P 1 2 B 1 then this P 1 2 does not move when the body moves from the first position to the second position ,so in other words if I pivot the body at P 1 2 and rotate it about that then I achieve the orientation a 2 B 2 by means of a pure rotation ok, I achieve it by means of your pure rotation, so if I look at this, then the rotation so oh a 1 sorry not, not oh a 1 P 1 to a 1 has rotated to P 1 to a 2 by a certain angle and any line on this rigid body is going to rotate about, rotate by the same angle okay, I'm rotating about a fixed point any point any line on this legit body is going to rotate about that same so P 1 to B 1 also rotates by theta 1 to here theta 1, so actually for two position motion generation I don't really need a four bar, I don't really need a linkage.

If I can just find the point P 1 2 ok, in a reasonable location I can then pivot that body about that point and achieve this motion by a pure rotation about the fool okay, but in many cases it might happen that the pole is in a location that is not really accessible okay, so it may not be possible to locate the pole and pivot it at the pole in order to achieve these two positions okay, so in that case what do we do.

You see here so if you look at a four bar, if you look at the fourth bar, this link Oh AAA and OBB are pivoted about fixed pivots, so these 8 traits as an arc, b traits as an arc, centered at the respective fixed pivots okay. So, when you pivot it raises an arc okay, so that means the two positions of a lie, that so the center of rotation will lie somewhere on the perpendicular bisector of the of a1 and a2 okay, so if I pick my Oh a somewhere on the perpendicular bisector of a1 a2 ,then a1 will move to a 2 along an arc right ,similarly again for b1 b2, I pick a point on the perpendicular bisector, so if I locate my fixed pivot there then I get my four bar linkage Oh a a1 a 1 b1 b 1 Oh B okay, so when oh a a1 moves to Oh a a2 b1 Oh B b1 will move to OB b okay, so here now I have got an ax because my pole may not be accessible I can do the same to position motion generation by means of a four bar linkage okay and I'm using the fact that a and B the locus of a and the locus of B both lie on arts that will be centered around the fixed pivot, so I can look at my I locate my fixed pivots for the four bar on the perpendicular bisectors of the positions of the two points. So this gives me so when I pick the pole as my solution for achieving this movement, then I only have a unique solution, because there's only one pole for the two positions, so it's very limiting in that sense, instead if I want to design a four bar for the same thing, I can pick for a Oh a I can pick any point on the perpendicular bisector, I can locate my o a anywhere on the perpendicular bisector which gives me an infinity of choices, because it all has to lie on that line okay, so I can pick a point that lies on that line similarly for OB I have an infinity of choices okay, so I say that I get infinity square solutions for the four bar to achieve this two position synthesis, of course other constraints will come in which will limit the number of solutions that you eventually come up with, but theoretically I could have infinity square solutions for this problem okay.

Refer slide time: (20:32)

## 2-position Motion Generation

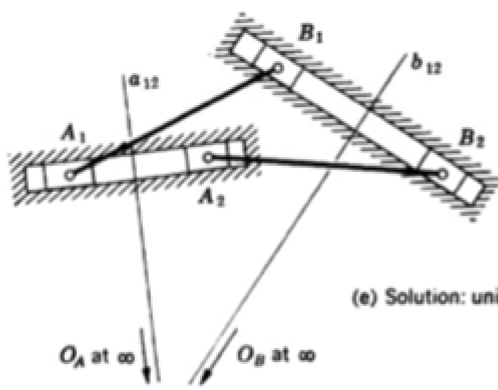



I could also choose to do this with a slider crack, remember the slider crank is only a limiting case of a four bar, so that means in this case I say b1 b2 is my path my straight path, in which the slider block will move okay, so here I will just I have b1 and b2. I draw the perpendicular bisector so essentially I draw the path which is perpendicular to that okay, so in this case only two positions so it's a straight line doesn't matter, you know I don't really need the perpendicular bisector, then for a1 a2 I locate o a somewhere on the perpendicular bisector of a 1 a2 okay. Which is this one, so I get a slider-crank now what happens I still have an infinity of solutions based on where I pick o a, however my slider path now becomes fixed ok, b1 if I choose to do this with a slider crank then my slider path is fixed because the pivot for that is that infinitely, earlier I had an infinity of solutions, when I could take if I wanted a finite location for the pivot now I have only one solution which is at infinity so that becomes a straight line, so I lose the choices in terms of that, so the number of solutions that I have is an infinity of solutions based on where I locate the pivot o a ok, I could also do the other thing I could instead make a1 a2.

Depending on what I want as my input okay, I could make a 1 a 2 as the slider path and then pick OB anywhere on the perpendicular bisector b12 okay, again infinity of solutions are possible because of the flexibility in choosing OB. What do you think is the next option?

Refer slide time :( 23:15)

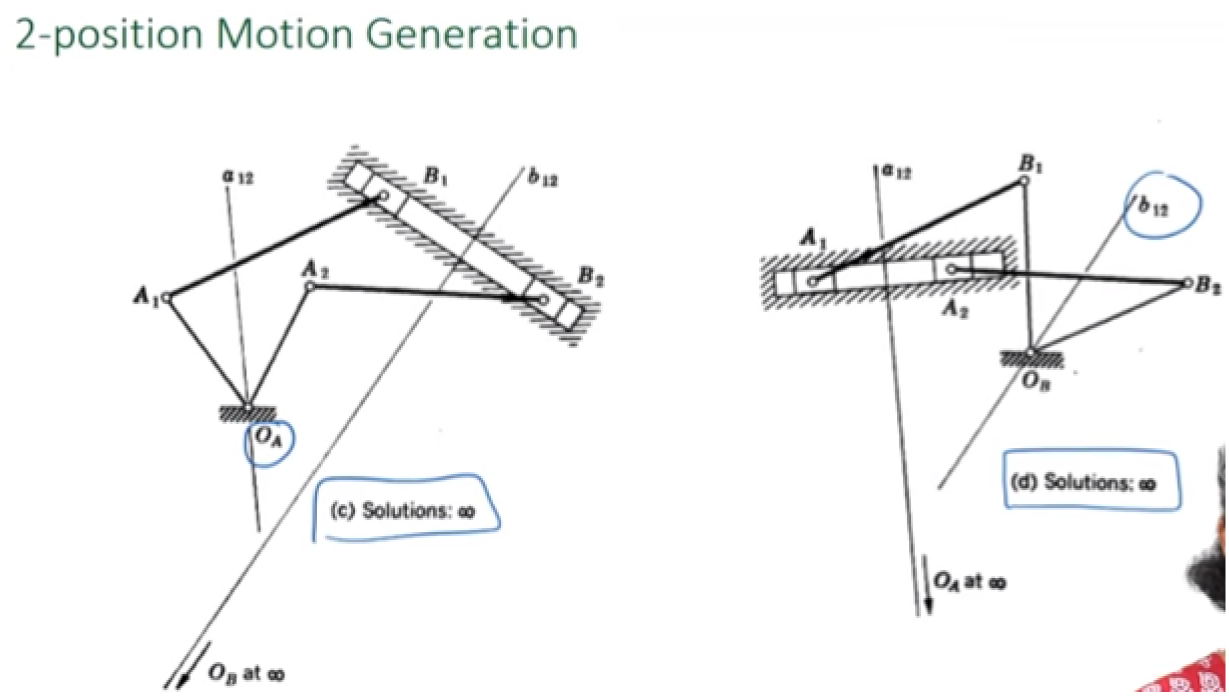
## 2-position Motion Generation




(e) Solution: unique

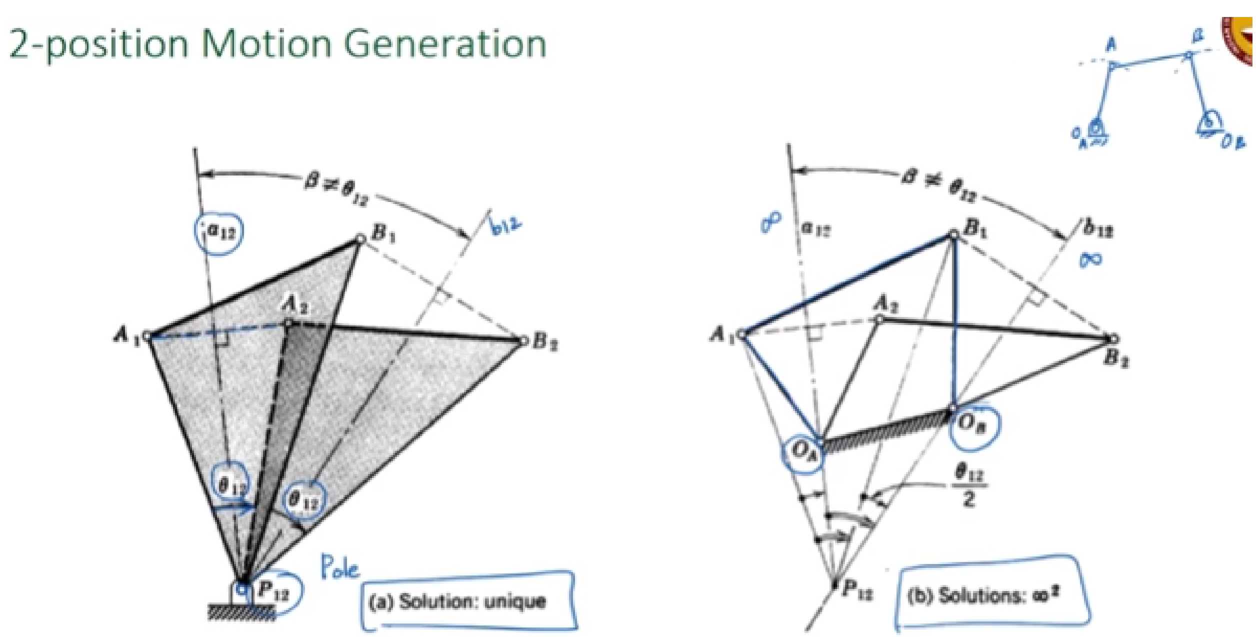
Both sliders only one solution right becomes a unique solution, because this is also at infinity that is also at infinity okay, so if these are guided then a1 b1 will move to a2 b2 I can, one thing that we have not looked at here okay.

Refer slide time: (23:47)



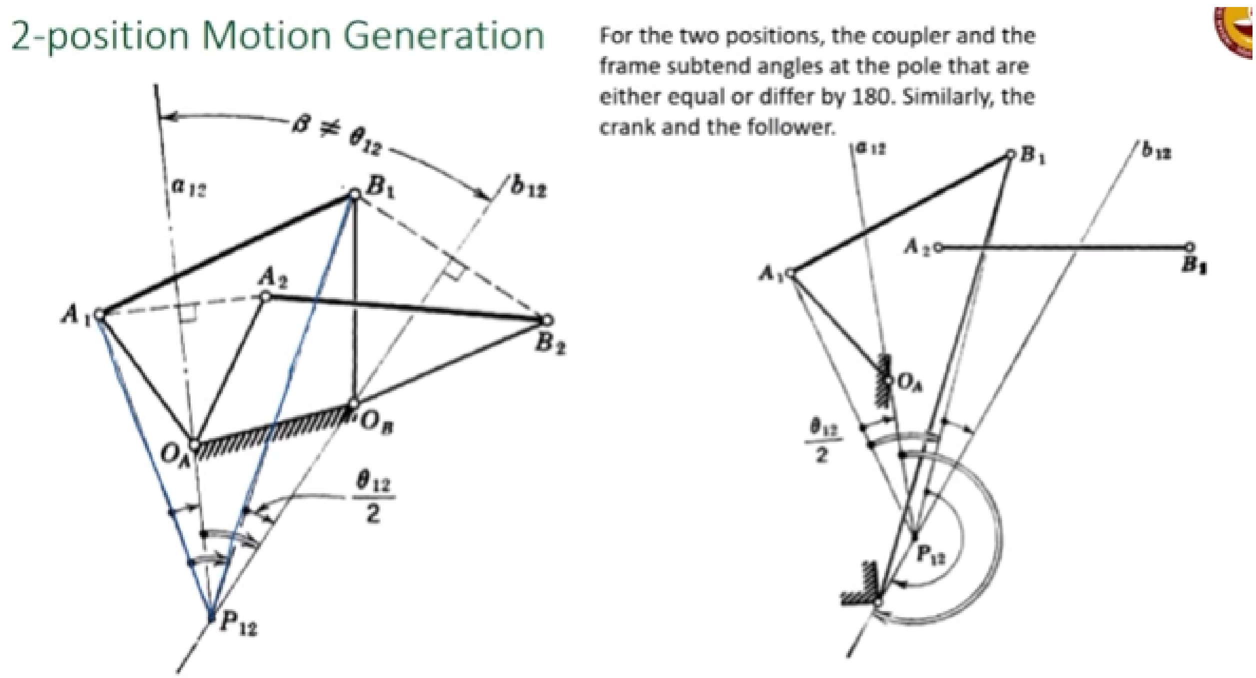

Is with any of these mechanisms, it will achieve a 1 B 1 2 a 2 B 2, but we don't really know what is happening in between that's one thing. And we don't know that these two are limiting positions, so it may go past on the side or this side, you know you you're not really looking at how the linkage behaves.

Refer slide time: (24:10)



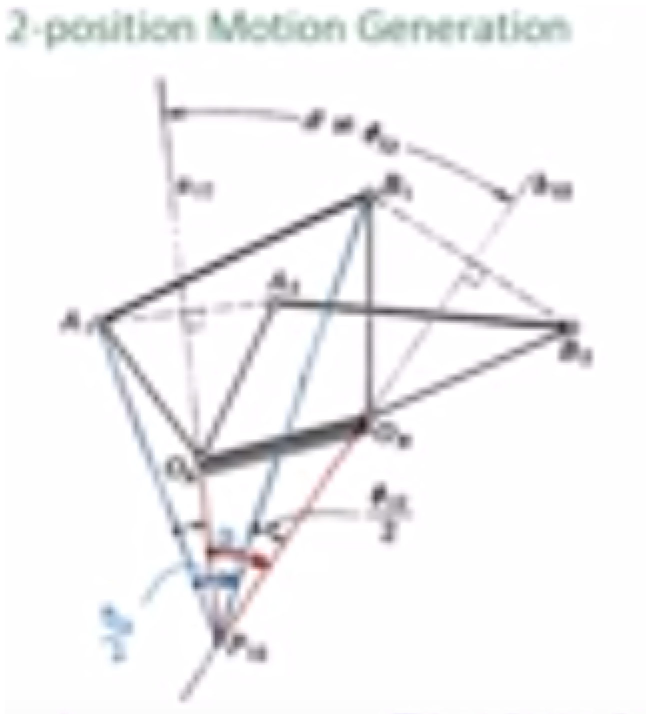
Behind here also, I'm saying it'll achieve a 1 B 1 2 a 2 B 2 if I use this for bar, but I'm there's not there's nothing to say that it'll only move between these two points. These two are not limiting positions that I'm looking at, in many real-life applications you know. You want your design positions to be limit positions. You want to pick something up place it there, you know orient it do something here so.

Refer slide time: (24:48)



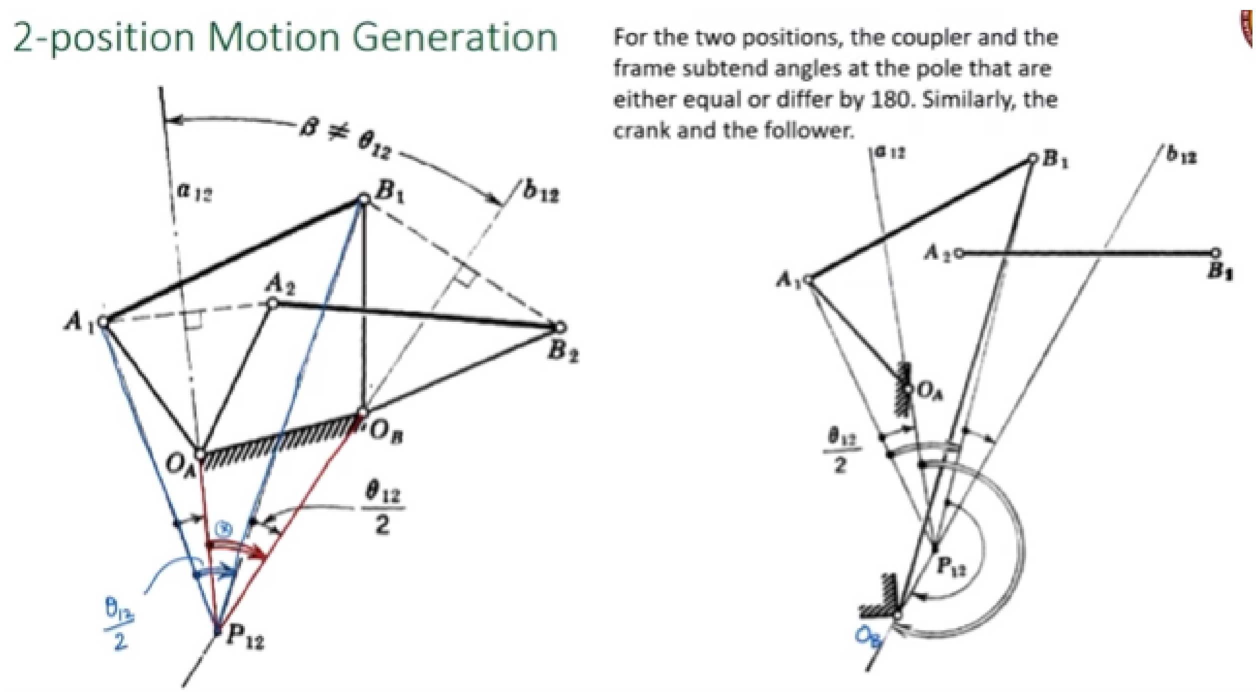

And you don't want the linkage to then. Move beyond that interfere with something else, or you know do other things, you want these two typically to be limit positions. So one interesting thing that we will look at, so here depending on so here I chose OA and OB like this with respect to the pole and if I look at the angle subtended by the coupler a1, b1 and the frame OA, OB at the pole okay ,so I draw these triangles okay.

Refer slide time: (26:03)



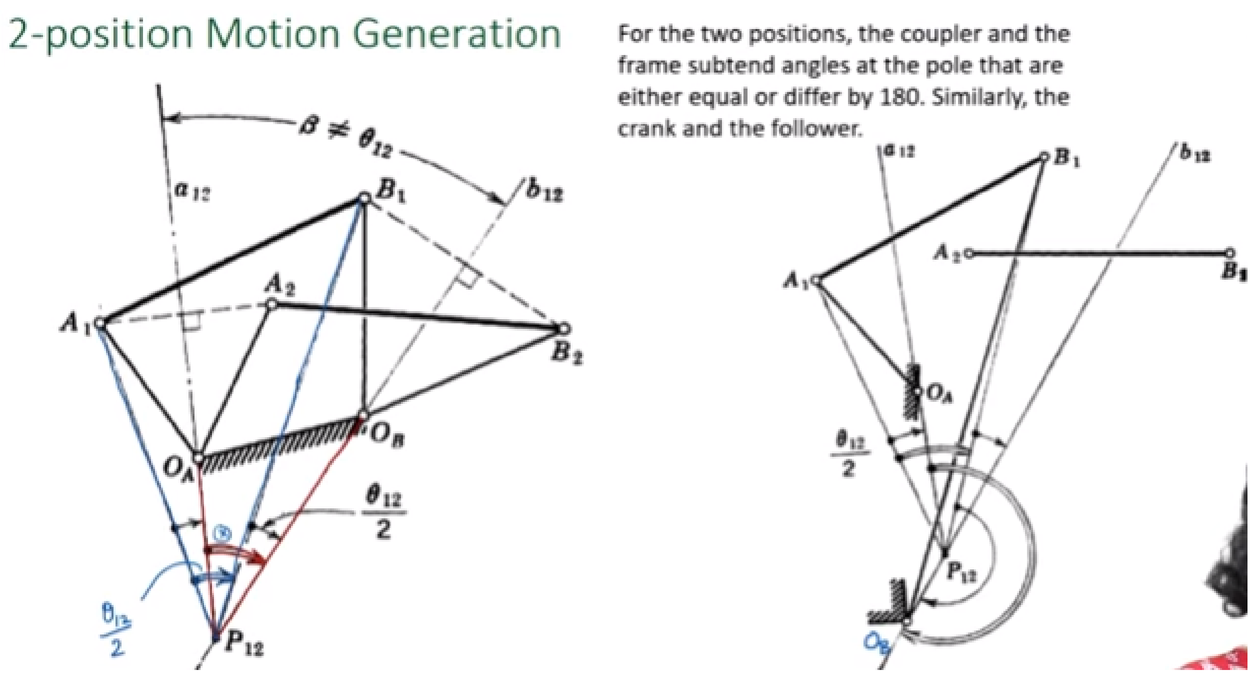

I have a1, P1,2,  B1, I have O a p12 O b okay, if I look so I can say this here is the angle subtended by the coupler at the pole okay and then I have O a ob subtending this angle at the pole okay. Now tell me if I look at this is the perpendicular bisector of a1 a2 okay, so this angle here, this angle what is it equal to? Theta 1 2 by 2 okay, then this angle similarly here OB p1 to b one again is Theta 1, 2 by 2 okay, this angle is also theta and this angle is common to both this part, so this angle the blue arrow equals the red, equals that angle that is at the pole the angle subtended by the coupler, is equal to the angle subtended by the frame okay.

Refer slide time: (28:04)



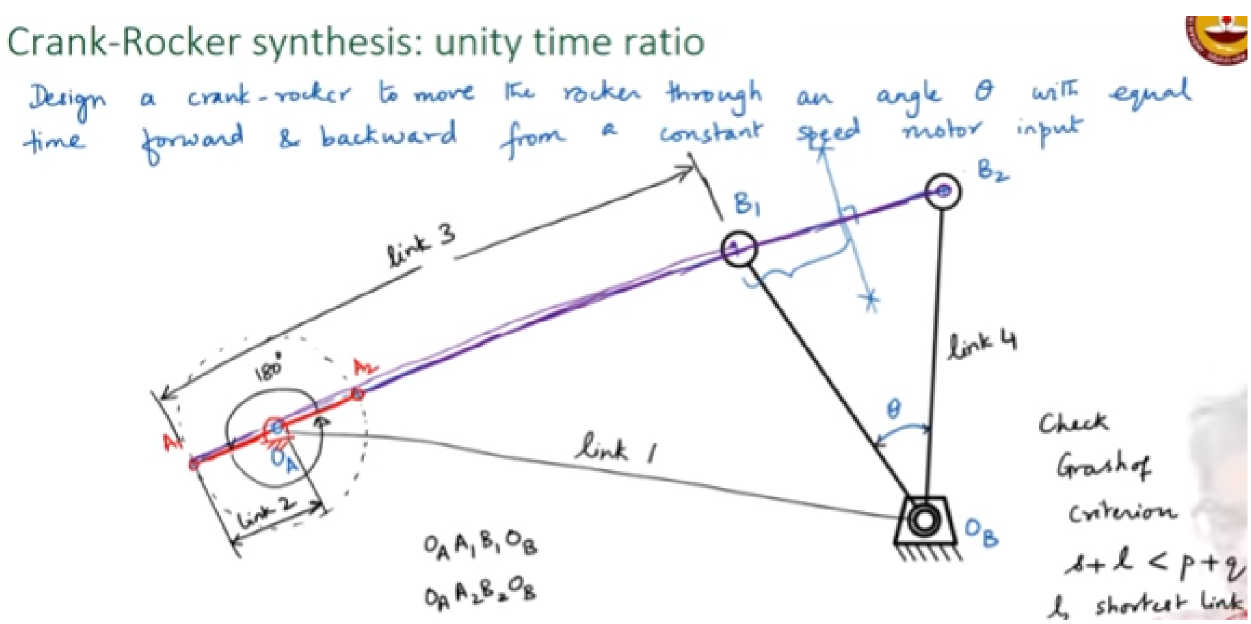

This is something that we will use later or they will differ by 180 degree, so if you happen to choose OB on this side then you will find that they differ by 180 degrees the two angles okay. And you can also show that the crank and the follower subtend, so here you have obviously you can see the crank subtends an angle of theta one two by two okay, Oh a p12 a one p12 that angle is Theta 1, 2 by 2, which is equal to the angle subtended by the follower the rocker okay, so the opposite links subtend the same angle at the pole the frame and the coupler subtend one angle at the pole that are equal or differ by 180 degrees similarly for the crank and the follower, this is something we will come back to later and we will use this for function generation synthesis okay ,the polls give you a very elegant way of synthesizing for function generation, this fact about the poll ,so we will come back to that later but I just wanted to mention it at this point.

Refer slide time: (29:45)




Do you see here, look at this figure okay, do you see why this angle is Theta one two by two, you see that? Okay, similarly this angle is theta 1 2 by 2 you see that from this one, now this I just showed you theta 1 2 by 2 this is again theta 1 2 by 2, this part of the angle O a p12 B 1 this part is common to both right? So this angle is Theta 1 2 by 2 plus this X which is equal to this plus this, so the two angles are equal okay. So coupler and frame crank and follower opposite links, they subtend equal angles at the pole or angles that differed by 180 degrees. We will come back to that later okay.

Refer slide time: (31:13)



So now two position motion generation you can solve it without a linkage by just pivoting at the pole okay, or you can use a four bar or a slider crank or double slider whatever mostly we look at the four bar, so the other practical problem that I use that we usually encounter is where you want the rocker to move through a certain angle, move between two positions subtending a certain angle okay ,as and you want the input to be continuous, so I want this to rock back and forth between these two limit positions okay, so I want to design so let me state the problem I want to design a crank-rocker, so basic II design a four bar linkage or I'll just design a crank-rocker which means it should satisfy the grashoff criterion and the input should be the shortest link to move the rocker through an angle theta okay, will first do this and then we will look at and I want to do this with equal time forward and backward from a constant speed motor input, very Pro common problem that you would encounter. Okay,

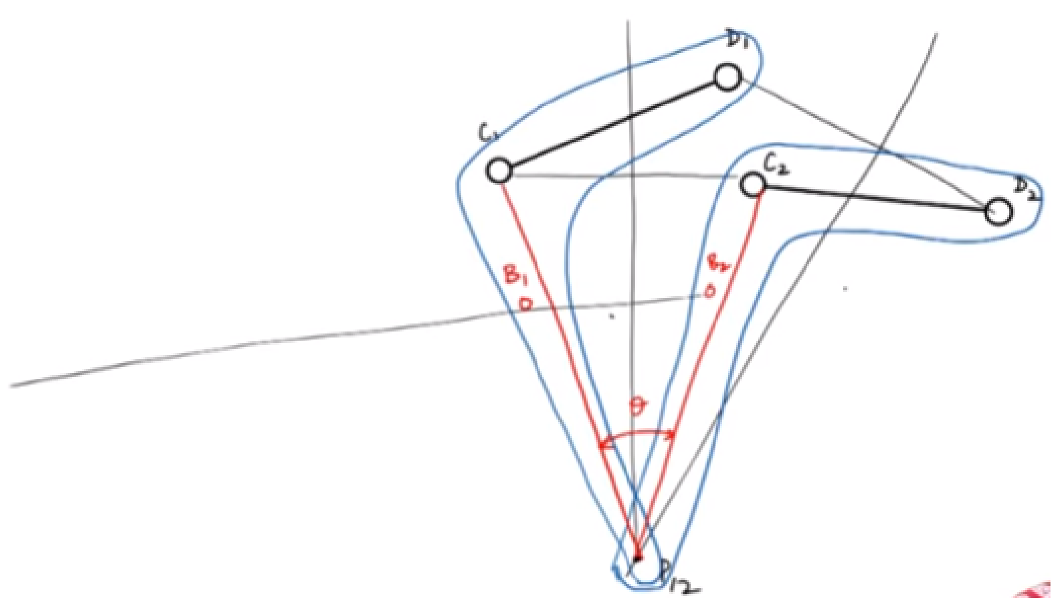
So I give a continuous input using a motor or could be a hand crank although it's far more difficult to crank with a continuous speed hand crank, so let's say a continue and I want it with equal time forward and backward ok. So let's say, this is my first position, this is my second position, I want is to move from and I will this will be then OB, so I pick my pivot OB someplace, I pick a certain rocker length which may be determined by other considerations of your application and I say okay now my problem becomes I want to move OB b 1 to OB b2 ok through this angle. Sorry, yes I'm limiting those positions, I wanted to move between these two and no not beyond them okay, these are limiting positions. So I'm only moving through this angle theta, I don't want the rocker to move beyond that okay, so that's what it okay, so now let me do this, so I have I have chosen this then now I'll first do the construction then it will become clear to you why we do that, so I take I join b1 b2 and I extend it. okay, then I draw the perpendicular bisector of b1 b2 okay ,then I pick some point on this line and I designate it as O a okay it will become clear why I am doing that and then I take this distance. Which is half of B1, B2 and I mark that on either side okay and I say that, this is a 2 and this is a 1 okay, so now. Now I say my design is complete show you why, so I keep that as one fixed pivot, this is my link Oa, a2, a really long link a to b2 , then in the second position this is my link O a a1 and now a1 b1 is overlapping from here to this point, so you can see that as the link O a , a moves by 180 degrees okay, I have moved from OB b2 to OB b1 the rocker has moved from OB b2 to OB b1 and then now when this moves back again I move from OB b1 to OB b2 and because I’m moving an equal angle and I’m running it through a constant speed motor, the time taken to move from a1 to a2 is the same as the timed and ,and therefore B1 to B2 is the same as the time taken to move from a2 to a 1 or B2 to b1. So essentially I have constructed it through simple geometry that two extreme positions of the linkage, how do I know it's a crank-rocker? Right now I don't know, I have to check, what do I have to check, I have to check the, so my linkage here is O a, a, a1, b1 O b in one position or O a, a2, b2, o B those are the two extreme positions, a B will be equal, Y is a b equal? Because, this distance okay this is the perpendicular bisector right, so that that is what I am taking here so O a, a 1 equals O a, a 2 so if I look at this I can easily show that this distance is the same as a 2 b 2 – okay, so this is link 3 a1 b1 is the this is link 3 ,this is link 2 that's the length of the crank and then link four this is link four link 1 will be the distance between o a and OB.

So, now I have my four link lengths, so I have to check grashoff criteria, I have to ensure that l2 is the shortest link and s plus L is less than P plus Q. If these two conditions are satisfied, then my just looking at the two extreme positions I know my linkage will behave the way I wanted to that is I can give it a continuous input and have the rocker move between these two limit positions okay, so this angle again is specified, theta is specified, so I can take some length for o four, so that was a free choice, so you can see that I made a lot of decisions here okay, I had the freedom to choose several things here in this design ,I chose where to locate OB, then I chose my rocker length it may be given to you, you may have to choose it the angle theta was given to me, sometimes they'll tell you what the initial position should be so this the orientation of link 4 may have been given okay, so I might have said no you have to start with this at a particular angle okay, or that may be a free choice.

Then b1 b2, you know where I locate the pivot that also is a so basically the length of link 4 was a free choice, then it had to be located on the line joining b1 b2, so that was not a free choice there, I had to look at oh is somewhere on that line so I had an infinity of choices on that line, so I located it at some point O a  okay, if I'm if I can locate something freely in a plane how many choices do I have I have infinity square for a point on the plane. Because I have the x coordinate and the y coordinate if I am locating on a line, I have an infinity of choices so I have this ,so this was also a free choice was the length of oh a, a - a free choice no it was 1/2 B 1 B so that was no longer a free choice okay, so once I know this and I know where I am locating B b1 and b2 the length of the shortest link becomes determined, okay and you can see here that if I oriented if I take this entire linkage I have these link lengths assemble like this, if I orient it anywhere in the plane, in any which way I'll still get the same input-output relationship okay, that's something that you will see with all function generation problems okay, because the angular relationships between the input and output link are what you are concerned with and that does not change when you change the so I could design this and then finally orient at orient my linkage such that my O a OB is horizontal, which is fine okay ,it will change the orientation of the other process so this is the case for them.

Refer slide time: (45:08)

## 2-position synthesis: rocker output



Now, how can I use this for the original problem that I had, which was I have this let me just call it C 1 D 1 C 2 D 2. I have this rigid body that I want moved from C 1 D 1 to C - D - can I limit it to those two positions using what I just did, how would you do that? Not enjoy you know, how will you consider this as a follower, poll so remember that this if I can find the poll for this motion then essentially this becomes a rotation about the pole. I don't know how well this will work out, but let me try ok not bad okay, so then essentially this problem becomes.

If I this problem becomes moving this rocker okay ,p1 p1 2 c1 2 b1 2 C 2 right because if this line moves here that it's all part of the rigid body so everything will rotate by the same amount, so if I am able to rotate this through this angle and limit that rotation to that then I can move this rigid body so essentially I will move c1 d1 - C - D - and back and now I have also limited the motion, so I would, so I can pick my b anywhere on ,on this rigid body ok I could pick it here for example I don't know it doesn't necessarily have to be on the line p1 2 c1, so I can pick b1 here now because this is a rigid body I have to maintain the same relationship here ok, I have to locate it the same way with respect to. Sorry which I have not done here somewhere here okay, so then I can use this again do my construction similar to what I did earlier okay, find the midpoint of b1 b2 or find half of b1 b2 use that as my crank length and complete my crank-rocker synthesis, so I converted the motion generation problem into the dead center rocker output dead center rocker. Because I had only two positions and I wanted to limit it to those two positions and I got I can now run it with a continuous rotation okay.