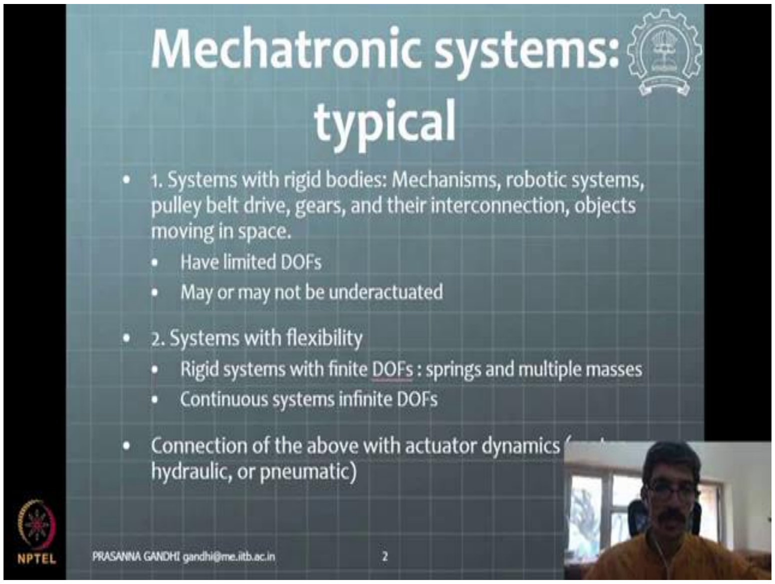
Design of Mechatronic Systems
Professor Prasanna S. Gandhi
Department of Mechanical Engineering
Indian Institute of Technology, Bombay
Lecture 36
Control design for linear systems

Now, we will start with this linear systems control design part. This is important part of any mechatronics system to develop the good controller, which is a brain of mechatronics system. So, we will kind of develop, also we have seen our feedback concept in the previous classes. So, we will now get into a little bit more details about specifically for linear systems how we can go about developing control and give you some kind of initial background in the domain of control design for mechatronic system in general.

(Refer Slide Time: 01:02)



So, let us begin with this part. If we see typical systems that will encounter in the domain of mechatronics systems is of the kind of rigid body systems or physics based flexibility. Now, rigid body systems may have single or multiple rigid body and single or multiple degrees of freedom also. The typical examples could be mechanisms, robotic systems, belt pulley drive and so on and so forth. They typically have a limited number of degrees of freedom. So, many times you have single or two degrees of freedom, although the number of bodies may be different, may be many.

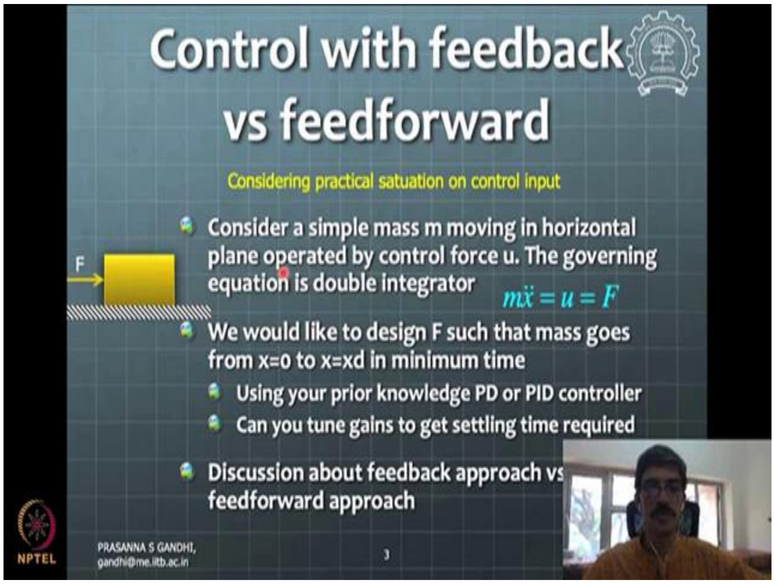
So, if you take an example of four bar mechanism, then it has four links, but actually degree of freedom is only one, because they are constrained to move in certain fashion and we have seen the kinematics of such motion in the past, in the previous courses. So, they may or may not be under actuated. Typically in like industrial commercial mechatronic system they will not be under actuated they will be having full degree of freedom actuation.

Contrary to that systems with flexibility will have by their nature infinite dimensional systems nature. So, what we do for control is some kind of a finite dimensional approximation by using some methods like, there are some method called assumed modes method or simply assuming like all the elements to be co-located at or a lumped kind of a mass and space kind of approximation. So, there are many different ways to do that.

These elements typically make the system, the compliant elements, flexible elements, like the system under actuated. So, one can handle this one or two degrees of under actuation by developing some kind of a non-linear controllers so that under actuated systems control is completely a different kind of issue to discuss and handle, especially in the non-linear domain. Linear domain still you have some kind of tools to deal with them. We will see some part something in the in class.

Then there are tools which are available based on the continuous domain model or infinite dimensional model which is partial differential equation based. This area is still developing. And of course these, both the cases like will have some kind of connection with the electromechanical actuator part or a hydraulic actuator part and things like that. So, these are, this is like the main kinds of system that we have.

(Refer Slide Time: 03:58)



Now, this important concept that we need to understand upfront here I am kind of putting it before we delve into other topics is concept of feedback and feed forward. So, as so that we can see for practical mechanical, mechatronic system implementation situations we may need to consider some advantages and like figure out which one of these or both of them together would be better for your application or not. So, we will see that in a little bit more detail.

So, let us consider our standard like example which we are familiar with simple mass moving on a horizontal platform with this force applied is F and that is our control input. Under this situation, we know the equation governing the system is this $F = m\ddot{x}$. And the question is, again, we have seen already that we have, we are at some position here. From here we want to move to some other position with say some kind of like given amount of time.

So, if you want to do that, we can come up now that we know how to plan trajectory. We can plan trajectory for that much amount of time for this mass going from here to the desired position and say like a representative trajectory or sinusoidal kind of a trajectory. So, you have a choice to like make a choice there and then get $x_{desired}$ value. So, at every position where I want x desire to be and once you have this $x_{desired}$ value available as a trajectory or as a function of time then we can differentiate that and get $\ddot{x}_{desired}$.

Now, when you put that m, put that x double dot desire into this equation multiplying m with $\ddot{x}$ desired then we get this desired force. Now, we can use that force and apply it on this system in time and by the end of like the trajectory whatever this x double dot kind of becomes zero again or block would have moved to the final position under ideal scenario where there is no friction, no other disturbances in the system. That is one of the ways of driving the system to the final position. We are not considering feedback in this case. Remember like we discussed feedback in a lot more detail, but now we are saying we will not, we are not using feedback in this strategy.

So, now think about like if we do this what are the problems. So, if we see the feed forward, so this is called feed forward case, where we are just giving whatever is needed for getting to the final position already like plan trajectory we give that input. Now, if the friction is there, the presence of frication will make this not go to the final position. It goes to some middle position depending upon how, what is the like the level of friction that we are dealing with here.

In contrast to these your feedback strategies a PD control kind of a strategy with somewhat tune gains such that we achieve the result in the same amount of time would work on this effects of friction because we are continuously monitoring the error between the current position and final desired position and depending upon that we are taking some kind of designing, some kind of a control input F.

So, these two cases came, in one case we are doing just feedback, in other case we are doing feed forward, there are some advantages and disadvantages. Can you figure out that? Think about that and then like we will come to combination of these two effects together in one of the control strategies. So, think about it, maybe you can list it down and then we can proceed. Maybe you can pause here and I am continuing further.

So, you see that in feed forward approach you will apply the input only what is required. We are not giving any extra input or whatever. Whatever is required we will give it and it will execute the strategy, territory. And drawback for that is, so input will be not really very high or very low. It is just whatever is the amount of input that is required. In, the flip side of it is that we do not guarantee to be in the final position because we are not taking feedback and kind of monitoring where we are.

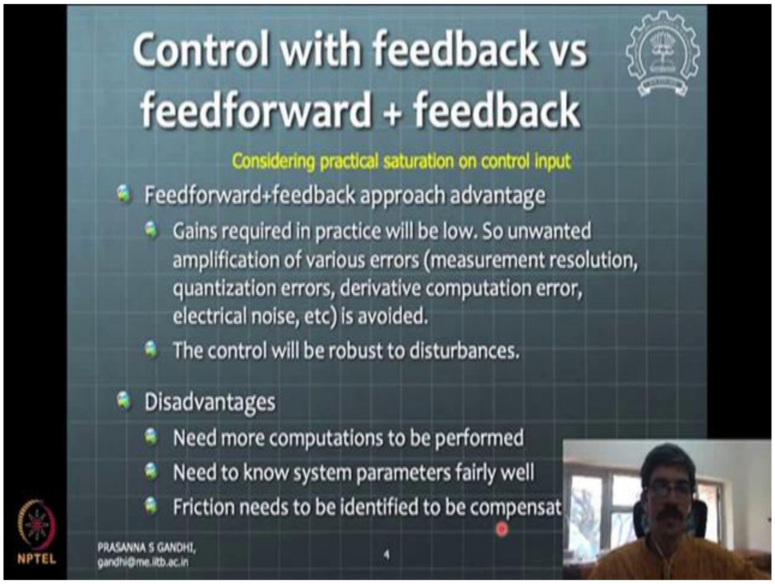
But if you go to feedback case you will find that we take care of like somewhat the error in this case. However, if you see initial kind of position to begin with, there is a maximum error between the desired position somewhere here and the current position somewhere here, so the error is maximum. And if you want this to happen in like a very small amount of time, then you would tune the gains to have very high values.

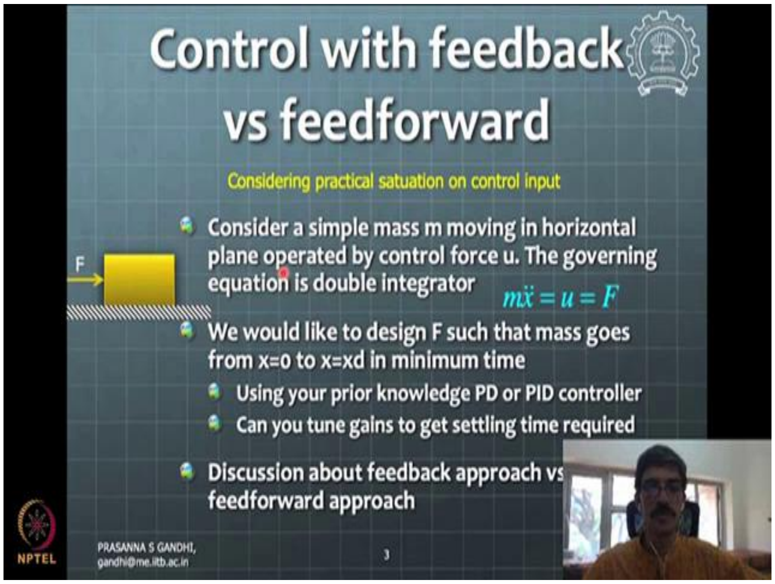
Now, when the games have higher values that is multiplying this highest error, then initial control input that we need to give to the system is really, really very high. And that, many times that input is beyond a level of actuator saturation limit. And that is where like you, although you have planned to go it in certain fashion by tuning your PD control, because actuators will get saturated for some time duration, the job is not really well done. In the sense, we do not kind of guarantee that time, whatever time we had tuned or PD control gains for. So, that is the flip side of feedback kind of a control strategy.

So, in effect what we need to have is some kind of a combination of these two strategies. So, we will, what we will do is we will keep the force which is required to go from here to here according to the plan trajectory and in addition now beyond this, even after we are doing this kind of a feed forward kind of a term, feed forward control application, beyond is whatever we want to control is where we add our feedback control. So, feedback control will be just looking at the error between the desired trajectory that we have found by planning and the actual territory x or actual position x. Only like if that every time point if it is not matching, you will apply the additional control.

So, error will be redefined here or $x_{desired}$ which was final position here is not now considered in PD control. Now, we consider the desired trajectory that we had planned that is considered to be a final finish for the error calculation.

(Refer Slide Time: 12:15)





And then you will find that the error at every point is not so high. It is kind of some small whatever is caused by the disturbances and that would like minimize or that would, like would not allow the control input to go beyond saturation. So, there are many other advantages of feedback plus feed forward control approach. So, naturally now since we are only doing like the error difference between this feed, whatever is achieved by feed forward and our plan trajectory that error is multiplied by gains. We do not need very high gains in practice. We will need, the typically gain values required will be lower gain values also can achieve the same level of performance.

So, because of this gains being low, so gains being low is a good thing for the sake of these other disturbances that unwanted errors are there like measurement resolution may there be some error, quantization error will be there, like number representation error or some kind of derivative computation error. So, these different kinds of errors or noises so to say are there, electrical noise. So, these noises which are undesired kind of effects they are showing up in the sensor measurement or whatever is, what are the terms that are going in the feedback terms.

So, if these terms are multiplied the gains then these errors also will get amplified. So, since our gain requirement in feed forward plus feedback control are low, I mean, the gain requirements means gain values are low, the amplification of this unwanted noise will not happen and that helps a lot in achieving better control action. So, the control in this case will be then because of the lower gains the perform, the response to these errors will not be there, will not be very high. And if the response to these errors is high that will disturb your control objective itself.

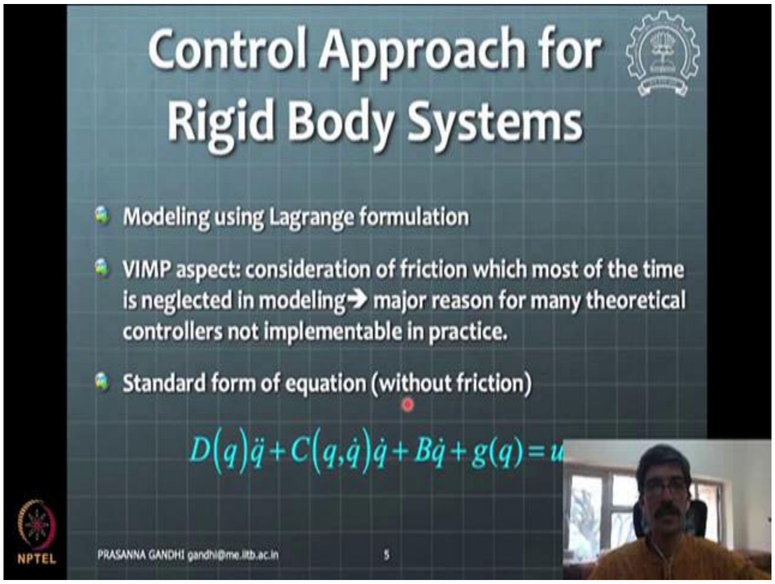
So, you want to kind of settle into final position, but it will kind of be jittery in the final question because these errors are unnecessarily getting amplified by the gains and that much input is going any way to your control. So, this is an important aspect of feed forward plus feedback control approach. So, as long as possible you should try to do this feed forward plus feedback kind of approach. And also with the lower gains the control will be robust to any disturbances that the system sees.

Now, what are the flip or like the cons of this feed forward approach is you will need more control computations to be performed in the microcontroller. So, you will need a microcontroller which is having like more power or more kind of high speed and then you need to know the system parameters very well so that you are able to kind of compensate for most of the things we will be able to compensate.

For example, if in this case the mass is not very well known, mass is known plus there is some data error in the mass. So, $m + \delta m$. Then the delta m error will be treated by the feedback control. So, feed forward will take care of like actual m, but that error delta m will be handled by the feed forward, I mean, the feedback control terms in the control. So, it is good to kind of have the better like better the estimation of the parameters of the system, better will be like lesser will be a load on feedback term in the control. And then you will need to identify a friction to kind of compensate for friction in the feed forward kind of a way.

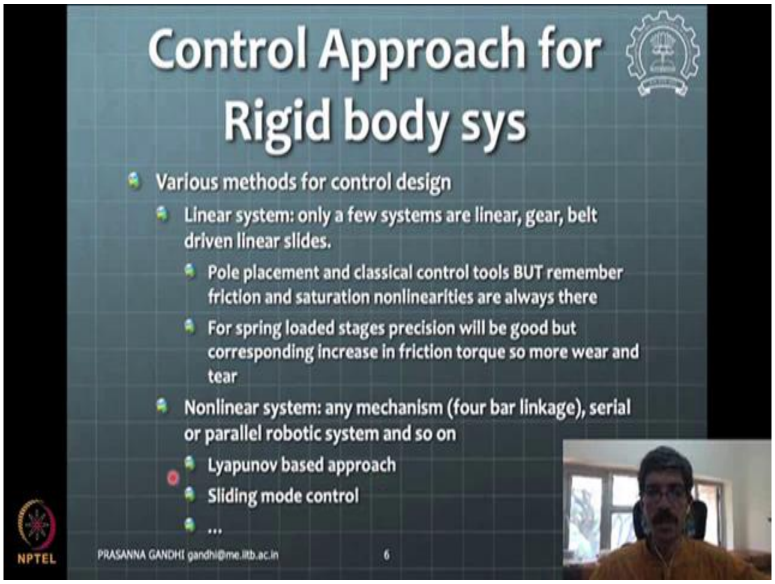
So, this is an important kind of a consideration. So, we will see many kind of controls that we will carry out we try to see whether we are able to apply both of these approaches together to control system design, control design basically.

(Refer Slide Time: 17:08)



Now, there are many approaches for rigid body systems. So, we will consider like mainly the linear kind of control design for now and then we will come to this Lagrange formulation based approaches. So, this form of the equation we will treat little later by adding friction to it.

(Refer Slide Time: 17:36)



Right now we will focus on mainly linear systems approach. And for linear systems you have these approaches based on the pole placement and classical control tools. You just need to remember here this but is there to, that is a friction that will be there in the system and many of the approaches that you are taking may not kind of completely represent system in its full sense without friction. So, friction, actual system has friction, their model will not have fiction.

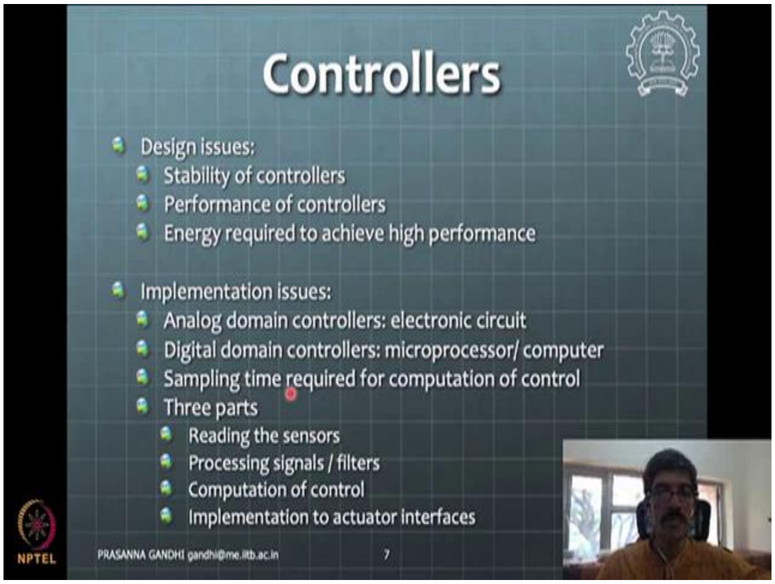
And other thing is saturation will be there in the actuators that we need to always monitor. So, you tune the gains and say my system is working fine, that is not enough, we need to look at with those gains what is a control input that is required and whether that control input that is going to the system is exceeding the actuator saturation limits.

So, these are kind of like these are main approaches that we will take and then we need to kind of see that in some cases the friction will be too high, in some cases it will not be so high, all those kind of nonlinearities need to be handled over and above what we have learned as a classical control techniques and tools.

So, for this typical approach will be first consider system without friction, develop control, and then implement it on a or simulation, implement in simulation on a system considering all the nonlinearities or all the kinds of other things and see whether it is in the presence of other non-linearity still it is giving the same performance or not or like is a performance acceptable or not and tune, retune the gains further to get the final kind of controller is one approach.

Other approach if that is not satisfactory, then we will have to kind of consider this in control development itself. So, our model itself needs to have these nonlinearities incorporated and we then use some ways to compensate for them. And this is a linear system kind of a design approach and non-linear system we will see Lyapunov based approach. Many non-linear systems approaches that are available for control, but we will go through only this Lyapunov based approach that is the most popular one. And many other approaches make use of Lyapunov arguments for its proof, for their proof.

(Refer Slide Time: 20:20)



So, there are two different parts that we need to think about when we do develop controllers. There are, first is like a design issues. What are the design issues that are to be considered when we develop controller? First is a stability. So, the way the system is stable the controller also should be stable. So, computation of the control it should be like from one sampling time to next sampling time as we are doing this control computation it should not run away like kind of in amplitude. So, control amplitude should be always in control.

That means like you will have a system controller poles they should be in the left half of the S plane. That is our stability criteria. So, when we do the controller design we need to make sure that that criteria is valid. The controller by itself should be a stable controller. So, we are developing these control to have the closed loop system to be stable, but in addition we want the controller itself to be having some kind of a stable nature.

Then we talk of performance of a controller. So, it should give the desired performance out of whatever controller strategy you have chosen should yield you settling time or overshoot in specified limits that the application requires. Then other thing is energy required to achieve performance. This is very important criteria. Often people miss this out that you need, like the energy required or control input required to achieve this performance need to be monitored and make sure that we are not exceeding these two about the saturation limit of actuator.

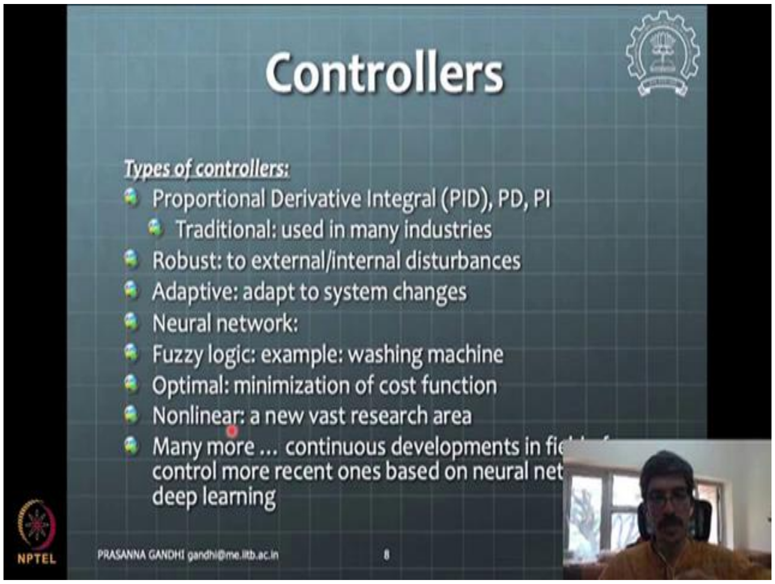
And then there are implementation issues. In the implementation you will have the two ways of implementation analog domain and digital domain. So, of course, here we are talking about digital domain. Analog domain people are not, nowadays these kinds of controllers are getting phased out, because we cannot change their gains once the circuit is made the tuning gains and doing some changes in the control algorithms is difficult.

And also for the cost nowadays because of the microcontrollers available at so less cost, I mean, implementation of this digital domain controller is comparable to the in cost to the analog domain kind of implementation and it provides additional flexibility, additional just change your program and your controller will change or control algorithm will change. So, that kind of flexibility is offered with the digital domain controllers. So, we will go for that.

Then other implementation issue we have to consider is the sampling time. So, sampling time, the main kind of thing is like we need to be able to do the tasks which we are supposed to do in the closed loop continuously. Every, after every sampling time we are doing this task or in during every sampling time we are doing this task. So, this task should get finished within whatever allotted sampling time. If the tasks are not finished then there is something called overflow will happen and some of the operations may not be done and then you will find that your control strategies not working very well.

So, we need to design sampling time in such a way that all these tasks are getting completed. What are the tasks we do? We read the sensors, then process the signals of the sensors in some way if at all, then compute a control and then implement that control on the actuator interfaces that we have chosen to. So, these tasks should get completed in a reasonable amount of time. So, that is one major requirement of sampling time. There are some other requirements. We will discuss them in more detail like as when we talk about signals and signal processing in mechatronics.

(Refer Slide Time: 24:37)



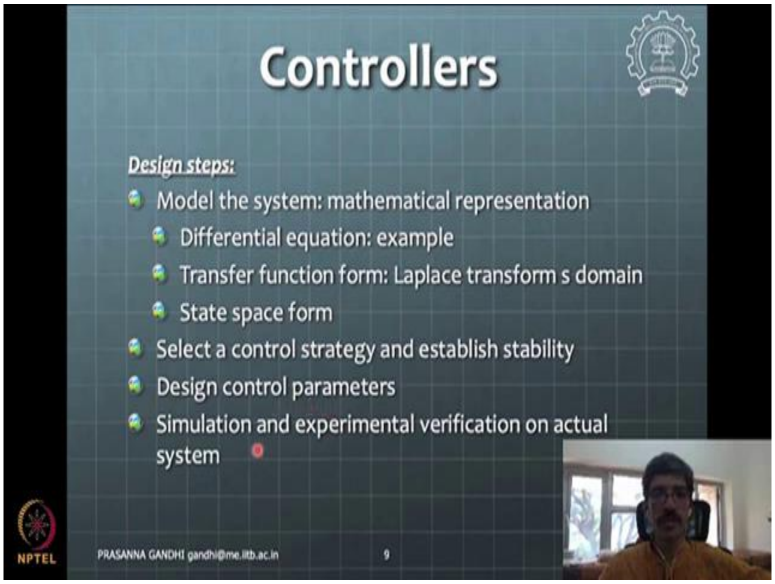
So, we will move from here to first type of controllers and then we will see some more linear system controller in little more detail. So, mainly people use the PID, PD, or PI kind of controllers traditionally and many industries still continue to use them. They work really well in many cases. Then there are this many different types of controllers; robust there are robust to internal external disturbances; then adaptive they adapt to the parameters as the system changes like some properties of lubrication or some other kind of environmental property changes happen slowly. So, the system adapts to those kind of changes in my changing the gains automatically something of that sort happens in the adaptive control.

Neural network control, so there are many, many different kinds of controls, artificial intelligence based control, deep learning based control, like are buzzwords these days to talk about, fuzzy logic is used in washing machine kind of systems. So, so many different kinds of control algorithms are there. Then even the non-linear controller also is a vast research area. There are several, several controllers or control algorithms people have paused. I mean this is ever growing area mathematically in this.

So, there are a lot of developments in this control field and many mathematicians are involved in this development. And not all the controllers that you will find in the literature are amenable for practical implementation. So, one has to have wise or wisdom to see which are relevant for a practical implementation and which are not. So that kind of wisdom they should slowly develop.

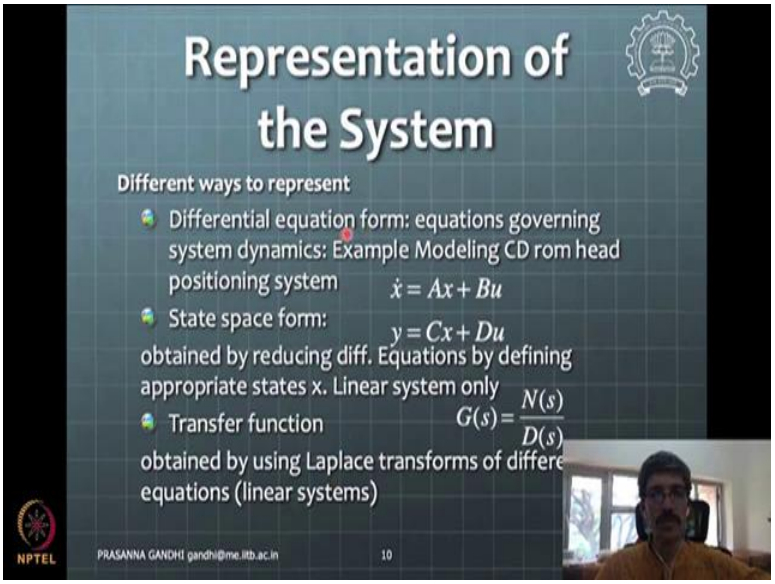
As we see some of these fundamentals of mechatronic systems, slowly that get that kind of understanding.

(Refer Slide Time: 26:46)



Now, the main design steps that you would consider for development of controllers is basically modeling system. This we have already done. And then representing that model into one of these forms either differential equation form or transfer function form or state space kind of a form. And then you will select some kind of a control strategy and do some mathematical analysis to establish that, that strategy gives you a stable controller. Then you put some effort into designing the control parameters to make sure that your desired goals of the applications are achieved. And then simulation and experiments and then reiterate some of these processes to make sure that your final control is good.

(Refer Slide Time: 27:33)

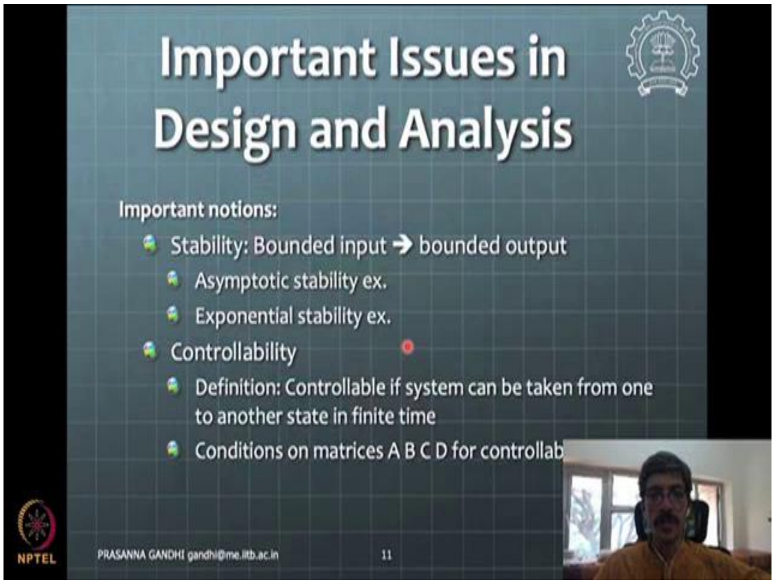


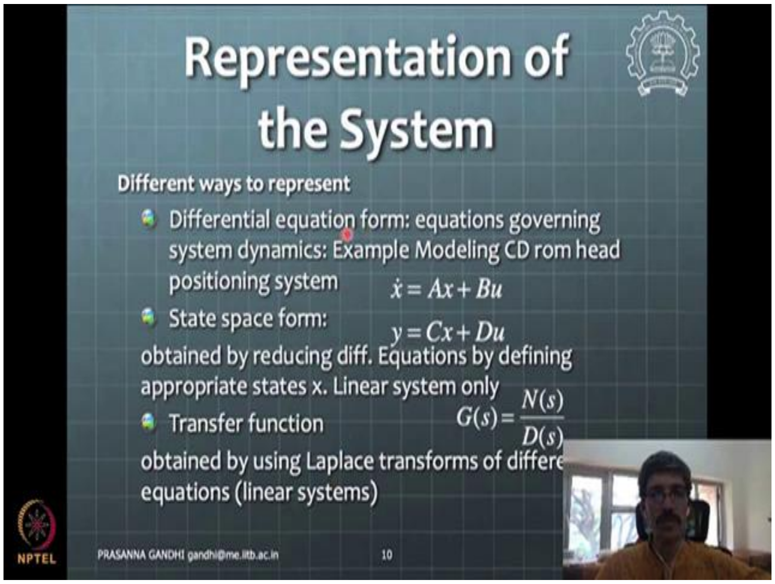
So, representation of system we have talked about already. These are the three forms differential equation form, state space form and then transfer function form. So, in the each of the forms you can go back from differential equation to state space, state space to like transfer functions, transfer function to differential. I mean, you should be able to convert system into one form to another form by appropriate mathematical ways.

These forms have, in each of these forms there are different tools for control available. But we will consider this differential equation to be most fundamental kind of a form of presentation which will give us some physical insight also into the system and we may use that in incorporating interesting ideas for control. So, this is, so although you may develop some controllers in a state space form, you can transfer them back to the differential equation form to see what their physical aspect or physical implementation or physical insight into that controller is.

So, these forms are mainly for the linear systems. They will not be known for non-linear systems. Differential equations can be also non-linear, but they will be able to, we will be able to convert them into state space or transfer function form only if they are linear equations.

(Refer Slide Time: 29:08)

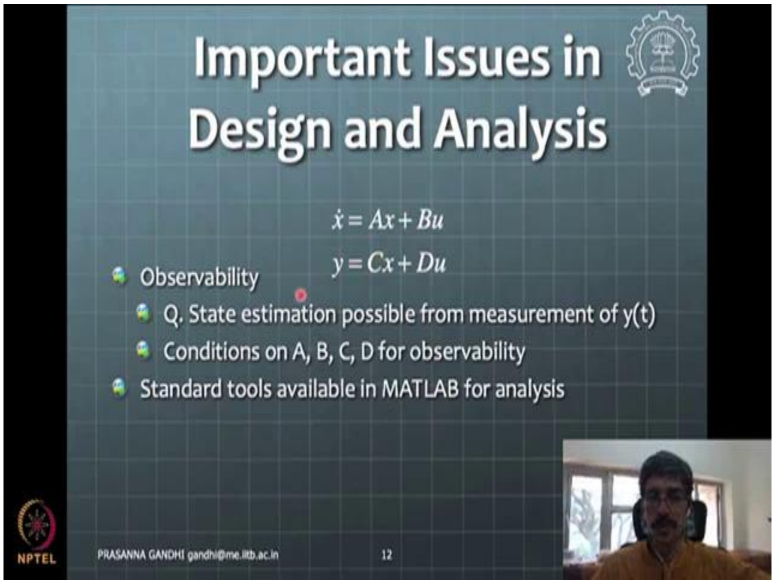




The notions or issues in design and analysis additional to what we have talked already about is, these are like more like a mathematical theoretical aspects. One is like a stability. So, there is a bounded input, bounded output stability notion that we have already seen in my class and maybe in some previous discussions. And other important notion is controllability.

By definition controllable, system is controllable if it can be taken from one state to another state in finite amount of time. So, this controllability and observability are two definitions that are defined in the domain of state space representation of the system. So, there are some conditions for controllability based on these matrices A, B, C, D of this form.

(Refer Slide Time: 30:10)

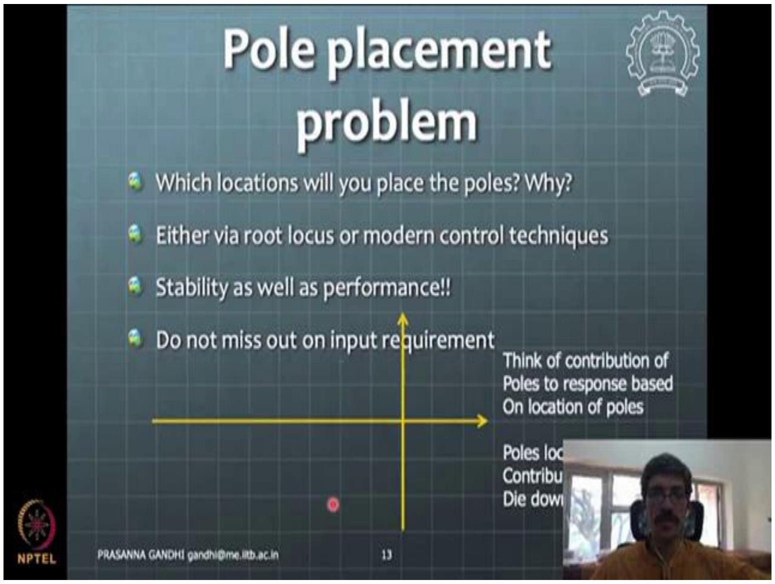


So, we do not get into that right now. We just need to kind of keep it a back of mind that some notion of controllability and observability exists here if we want to use some control techniques from this domain of state space then we will need to worry about these concepts. So, some of you might have already in the classes gone into a good depth of this controllability observability. You may be free to kind of use them to for your predict purposes if you would like to or analyze some of the systems from that perspective and see.

So, in my opinion if you develop the design into the transfer function domain or differential domain you can understand that. That is fair enough to go. If you want to, we can design also in this other domain. But I mean space is actually personal choice I would say. I prefer to kind of like choose to not get into the state space domain too much actually. It just, I think it is just a personal choice. But I mean you can feel free to kind of get into different domains and look at the things.

There are standard tools available in MATLAB for analysis in the state space domain also and in transfer function domain also. One can use those techniques to do designs.

(Refer Slide Time: 31:47)



Now, we come to this important consideration of pole placement. So, I hope you understand this term pole placement. Pole placement refers to like how do you design control so that the closed loop poles will be at certain locations in the left half of S plane. Of course, we want all the closed loop poles in the left half for the system to be stable. So, how do we kind of make a decision of where the poles should be placed given the design requirements or some applications, what are the considerations, how do you kind of like do this choice of these poles.

Once the choice of the poles in the, in this plane is, left half plane is known then we can kind of see currently open loop system poles are at what locations and now if you want the close loop poles with these locations what are, what is a control, how do we develop control that kind of a control problem can be posed. Then that can be achieved by like either using root locus or any modern control or state space based kind of control techniques. For example, if the system is controllable you can place the poles at any locations in the left half plane. That is a very strong result from the modern control theory you get.

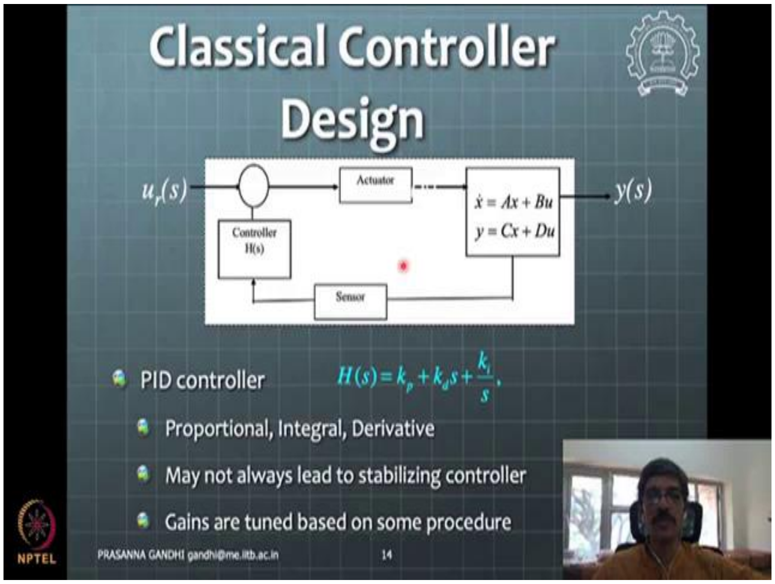
Of course that requires state feedback, but still I mean by using the state feedback you can actually design the controller such that you get the closed loop poles at desired location in the left half. So, there are how, what are the criteria is to think about, one is a stability criteria which already satisfied, but now other criteria is performance and then other criteria is input requirement. So, if we start like performance, from performance perspective, we know that if the poles are far away in the left half then their contribution to the control output is very low in the sense the contribution dies down fast. So, if all the poles are placed far away here, then the response will be very, very fast for the system.

So, we ideally would, we say look I want all the poles to be far away as possible that will give me a best response, that will give me a very nice kind of evolution of the system. However, it is not free. Things come at a cost. What is the cost that you need to pay? As you push the poles away from the left, away in the left half plane, away from this imaginary axis, then you start increasing the control input that is required to drive the system. And that is not desirable because our actuators have a limit.

So, that is why you do not want to push them too far away, but you need to have them like far away so that it is, your performance is good. So, these are fighting against each other these two things performance requirements and then input requirement. So, that is where like we, if you have a strategies which can give you better performance with a little lower like control input they will be accepted or they will be used in the application much better. So, also we, that is why from that perspective we may choose to have two poles retained close to the imaginary axis, relatively close, and other can be pushed little apart.

So, these two poles will be a dominant poles for the system and output response or contribution to the output from the poles which are far away here that will be anyway dying down. So, we can assume the closed system, closed loop system to have only these two poles like a second order system and then we can carry out some control parameter design based on the formula that we already know for standard second order system response, which formulas are also listed here for our utility later. So, that is one of this kind of strategies. Instead of pushing all the poles far away here, we keep two poles retain, two poles close to the imaginary axis and all other poles are a little farther away so that these poles are dominating. That is the kind of a way one can think about.

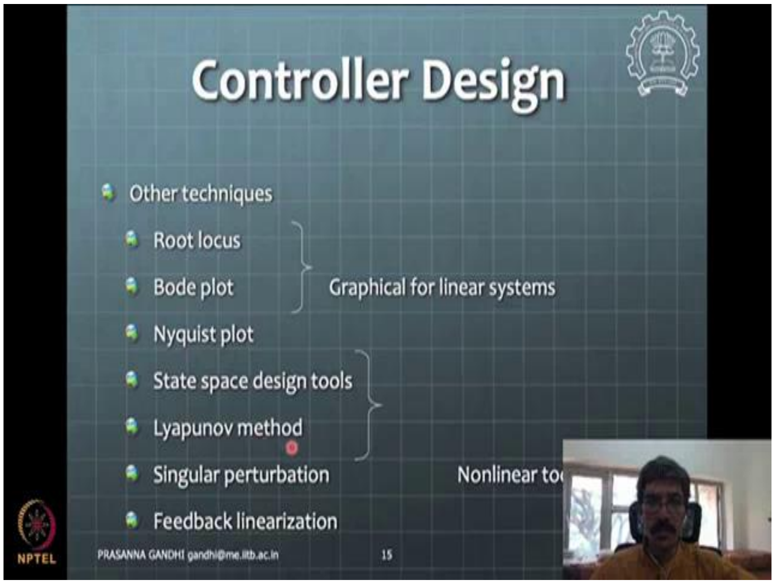
(Refer Slide Time: 36:50)



So, you can do that pole placement or like by using this PD, PID kind of controllers and see now with this controller introduced how the closed loop system poles are and things like that. So, one can like start constructing this kind of block diagram with the flow and then put controllers at appropriate locations. Your controller may be here or controller may be here. At some location controller is there. Then you can analyze the different, different proposals and see which one is best suited to give you the desired kind of response.

And typically, you will avoid to carry out pole zero cancellations by using controller. So, as we, some part we know or like we if we cancel the poles especially in the left half of the plane, they, this cancellation is not really going to be exact, because actual system and like the its model will have always some small differences here and there. And then only that small part which is not compensated completely that will kind of have some undesired effect on the system and it may not work well. So, therefore, from that perspective we do not do the cancellations here.

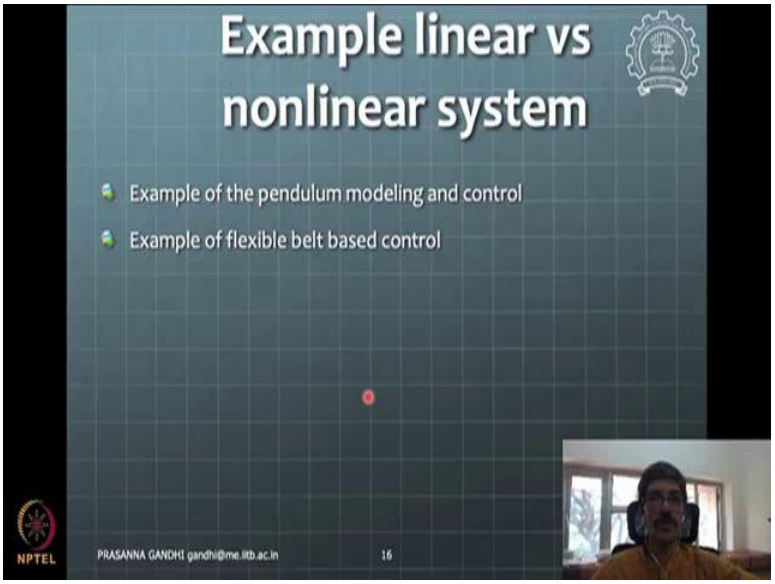
(Refer Slide Time: 38:23)



So, there are a lot of techniques that you might have studied earlier. Root locus, bode plot, Nyquist plot, these are like classical Laplace domain control techniques. Then you have the state space based designs which is based on the linear systems theory. Then in the non-linear domain you have this Lyapunov control, singular perturbation and feedback linearization or some many, many different approaches. I am listing only few of them here.

Out of that also we are going, not going to consider a lot of things for discussion. I will give you some kind of a flavor of how do we go about designing and analyzing and maybe you can use one or more of these techniques to kind of do this final goal for whatever system that is under your consideration. You can make use of this in your project for example. We will go through little more detail in Lyapunov method and like what is the best controller that can be possible for a rigid body a non-linear system. So, up to that we will do. And that is a good controller for practical implementation aspects as, practical implementation as well. So, that will be nice for us to look at.

(Refer Slide Time: 39:41)



Then we will consider now example of linear or non-linear system. So, the system which is non-linear to begin with is pendulum system. We all know it is non-linear if it will, if you consider this Θ for the pendulum to be not small and then like see what kind of a ways we think of its control. And then another example flexible belt drive system that you modeled in the assignment problem. So, I will, I would like you to kind of do this analysis. For that model is already existing with you.

So, you think about, suppose I want to position my load at some specific position and I give input as motor angle or motor torque or whatever you want to consider and I want the vibrations in the belt to be minimum or I want to position the output Θ_l at some position such as there are not much of a vibrations when I go to the final position. So, how do you kind of develop such a control or flexible belt kind of a system by using some of the concepts of this linear systems theory.

So, you, I would like you to kind of give a thought to that before going further into the discussions. So, I will pause here for the first video for this part of the lecture and then we will continue in the other video for the more discussions on these examples. So, let me stop here.