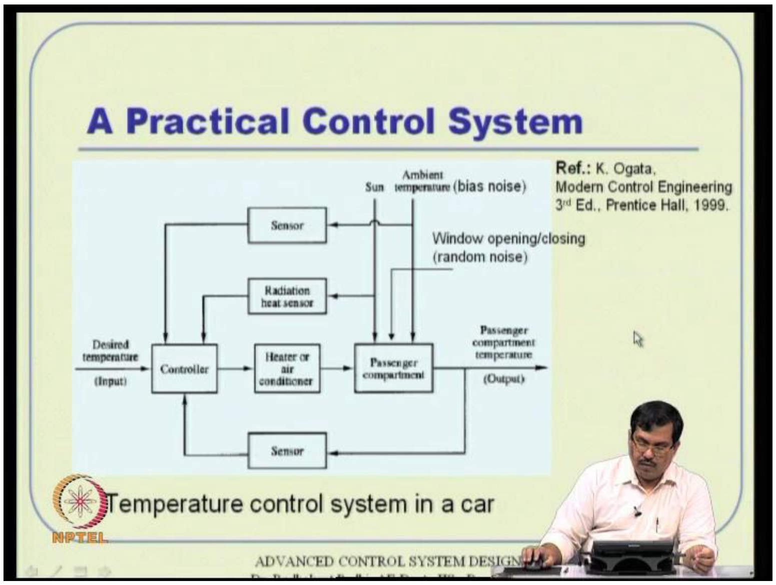
Advanced Control System Design
Prof. Radhakant Padhi
Department of Aerospace Engineering
Indian Institute of Science, Bangalore

Lecture No. # 09
Representation of Dynamical Systems - I

Previously, we have seen on the last class, the this some concepts of flight dynamics, I mean the basic principles and all. We will see later how to derive some equations of motion, and then take advantage of this control system design somewhat later. We will continue about our genetic theoretical development here; and I mean as you know this course is all about modern control design, which heavily I mean depends on this state space representation. So we will study in one class some ideas about state space representation, and then proceed further.

(Refer Slide Time: 00:56)



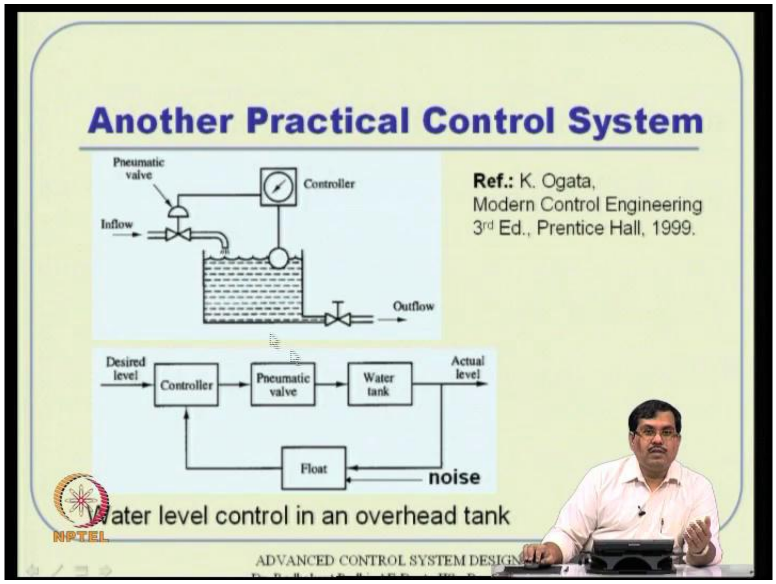
Before doing I mean formal analysis, let us study one or two practical control system example. When you see a practical control system you see many components here; this is a typical temperature control system in a car, where this is this is you your system you can imagine. And then I mean this is your controller primarily either a heater or a conditioner or a cooler essentially. And then this is the passenger compartment that you are interested in actually.

Now, you can visualize the system from the variety of sub system point of view, you are interested in temperature control, so primarily you need to know what is the temperature inside the passenger compartment; so obviously you need a sensor actually for doing that. And you really require some desired temperature depends on primarily the person who are inside the car, they sometimes you may want to set a different temperature domain actually I mean different temperature levels, but that comes in primarily as a reference input, and that is what you want to maintain some sort of a designed temperature.

And then there are ambient temperatures and sun, primarily you can think them as some sort of a bias noise or something, but some sensors are also available for finding out what is the outside temperature, so that information is also available for you, and then you can also talk about a sun thing and you can you can even have a radiation sensor actually, so you have a ambient temperature sensor, you have a radiation sensor, you have a temperature of the compartment, that also is measured through a sensor actually, so various sensors are there, and you can think of as a designed input the that is for your I mean that is what you said actually, there is a control algorithm, there is a heater or conditioner that is like an actuator finally, it will excite something for that for the system.

And then you have in various noise input in addition to control input, you can think of that as sun radiation ambient temperature, and sometimes window opening and closing that you can think of is some sort of a random noise actually, so all these things they go into the system for a proper practical control system design actually so this is one example.

(Refer Slide Time: 03:14)

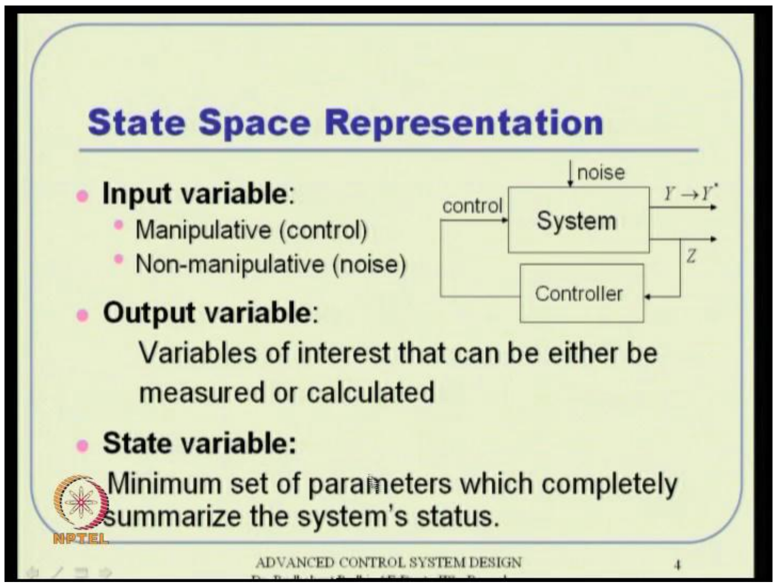


Second example if you think of as the simple basic flow control example, where you really want to control the water level control in an overhead tank, most of our big apartment buildings and all they do the simple control system mechanism realization actually, because many times we cannot observer the water system in whether water level has gone down we cannot manually fix it on, and think like that it is a not desirable it is not needed either actually, where a very simple control mechanism can do the job.

Now, how does it how is it mechanized you really want some sort of a desired level in the tank, and then you say there is obviously an outflow depending on your usage and all that, when there is outflow the level drops when there is a inflow the level rises obviously, so you can have a floating device here which can simply measure how much is the height actually, once you calibrate this device it will give you what level of the water is available here, and then there is pneumatic valve which will either open and close we always assume that the moment pneumatic valve is open then the water inflow is there great it is therefore, how they are actually.

So, then this particular mechanism you can put that in a flow diagram something like that, you have a water tank the pneumatic valve is your actuator, there is a controlled algorithm mechanism where desired input comes actual level is there, and where you measure this the moment this is falling you are something remember this will have some oscillation vibration and all that, so that will come as a noise to the input actually I mean noise to the sensor actually, anyway so all these things will go you constitute a control system really, so far we have been bothered about input and output only, but then there are much more details they are go inside the inside the system actually.

(Refer Slide Time: 04:58)



So, what are the such things that goes on in a typical system you can see there is a input there is a output, what input can be categorized into two sort of input, one is manipulative input what is which is a control variable, and if you see that this is a manipulative input for you this is also like a desired control and all I mean whatever control algorithm we are doing in all that that is manipulative actually.

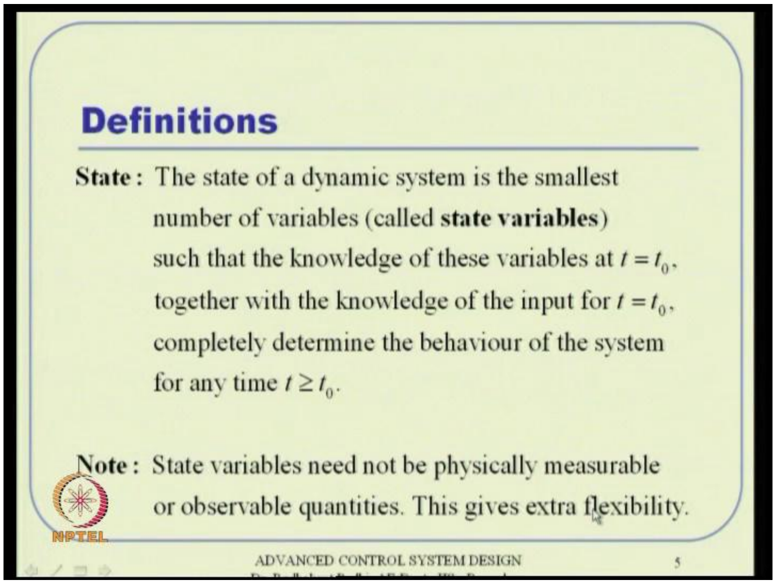
So, desired level or desired temperature here these are manipulative variables to the system, that is what you want to do something actually so those are control inputs, and then probably like what you see here these are like there are some non manipulative inputs also, you do not have a control on that for example, the vibration of this water level, or this I mean ambient temperatures on an window opening and all that, those are whatever you consider is non manipulative variables, those are also input to the system remember that there are physical input, input is going to I mean the system is going to respond to those inputs, and those are nothing but noise input actually, so you have a control input and you have a noise input.

As far as outputs are concerned there are they can be two types of outputs, one is the performance output or this may not be sensed by your sensors, and there is a sensor output actually, so this is something that you would you measure by sensor then take it to the control algorithm then your control input is computed then it is fight back to the system, so this Z what you see here need not be y it can be same as y, but it need not be y actually in all of on applications.

So, what you see here that two types of inputs noise input and control input, there are two types of outputs as well, but whatever dynamics goes inside the system that need not be represented fully by this set of variables, the control noise different outputs if you take there may be still something left out actually, so that is where the straight variable comes and that is how we define, minimum set of parameters which completely summarize the system status, that is the definition actually, we will see little more detail in next couple of slides actually.

So, we want some sort of minimum set of parameters using which we will be able to describe the system dynamics in a fairly complete manner actually.

(Refer Slide Time: 07:20)

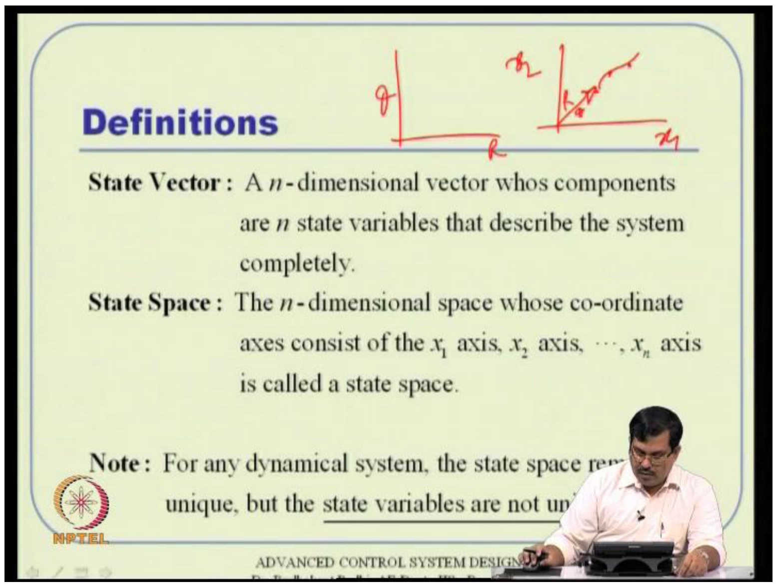


As a definition formal definition is like this, state means like this, the state the state of a dynamic system is the smallest number of variables, these variables are called state variables, such that the knowledge of these variables at some point of time t equal to t 0, together with the input variables for I think this is t is greater than equal to t 0, a small mistake here.

So, for together with the knowledge of the input variables for t greater than equal to t 0, that means you'll you know the knowledge of the input variable for all time after that, then it will completely determine the behavior of the system for anytime after that t greater than equal to 0, all that you need to know is initial condition of the state variables and knowledge of the control variables for all time, then we will be able to know the state variables for any time t greater that equal to t 0, after that you know everything about the system.

Remember typically the outputs are sub set out off state variables in a way, because if you know the state variables you can compute that compute the output variables as a function of some state variables, they may be directly selected from the set of state variables or they can be passing through some sort of algebraic function actually, and also remember that in this definition the state variables need not be physically measurable or observable quantities, that means the state variables need not be part directly part of these output variables that we are talking actually, we need not be measurable it need not it need not be part of your performance output either actually, so what that gives us extra degree of flexibility actually, how do you define this state variable something like that so that we can describe the system dynamics completely.

(Refer Slide Time: 09:16)



And then the further definitions we talk about state vector, so after the state variables if you put them this state variables together 1 by I mean 1 over the other x 1, x 2, x 3 like that in a column vector, typically it is defined as a column vector by default, then you will constitute a n dimensional vector assuming that you have n such variables x 1, x 2, x 3 up to x n, if you just put them 1 below the other it will form some sort of a x n dimensional vector that vector is called as state vector.

And then what is state space, see this if you consider the coordinates of a of a system n dimensional space which coordinates are nothing but x 1 axis, x 2 axis, x n axis, then it will constitute some sort of a n dimensional space, and that is called a state space, that is why all these things we will see state space analysis state space representation like that actually. So, when you talk about let us say R 2 space then you have x 1 and x 2 in 1, 2 coordinates that gives a R 3 space actually, assuming that these are all real variables by the way, and then if a three dimensional system x 1, x 2, and x 3 you can constitute some sort of a three dimensional vector, and remember these all that is required are the state variables are independent linearly independent of each other, that also satisfies a requirement of state space, the coordinates needs to be in linearly independent of each other.

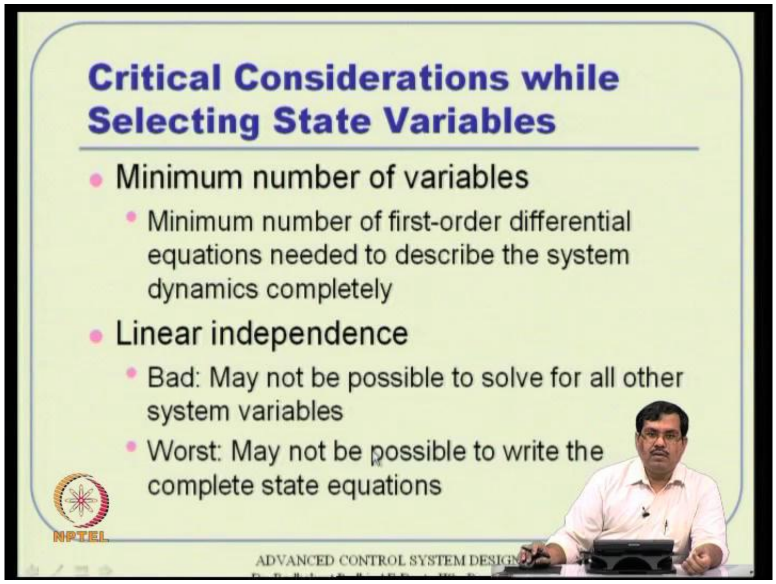
So, that is but, also remember that coordinates need not be orthogonal actually, that means if you take inner product of x 1 and x 2 that need not give gives 0, these axis need not be orthogonal, or they are certainly linearly dependent actually, and also remember that for any dynamical system the state space remains unique, that means if the problem demands n dimensional space then it is a n dimensional space, we do not talk about n minus 1 or n plus 1 that remains unique.

However the state variables are not unique, so state you can select a different state variable I can select a different state variable that that is not unique actually for example, if you constitutes some sort of a R 2 space let us say talk as let us say this is something like a R 2 space, I can constitute let us say x 1, x 2 here and represent some sort of a prime object by some x 1, x 2 coordinate actually, depending on where I am flying I will always find out, what is my x 1 and what is my x 2 here actually.

However I can also represent these points by some sort of a R theta coordinate, this is R and this is theta so, if I represent this thing something like R and theta that is also possible, remember this is still a 2 dimensional problem, so whether you talk about this I mean this system whether you talk about this R theta system or x 1, x 2 system the state space is 2, but the state variables are not unique actually.

But there will be conversion formulas available and we know that actually, R equal to the I mean R square equal to x 1 square plus x 2 square all sort of relationship they are algebraically related actually, and the relationship need not be linear that is also there, they are they are uniquely related that means R equal to square root of x 1 square plus x 2 square that is there that is certainly not a linear expression actually. But anyway so coming back to that what it means is the state variables need not be unique actually.

(Refer Slide Time: 12:52)



And some critical considerations while selecting state variables first of all they have to be minimum number of variables, so in other words the minimum number of first-order differential equations needed to describe the system dynamics completely, you can think of that way. So, if you talk about number of differential equation all represented by in a first order sense, then number of such differential equations that we needed difference of number of variables that are needed through describe the system completely, and that is nothing but the states actually, that is a primary requirement you need to collect minimum number of variables, the moment it is more than that there'll be problems actually, because if it is less than the minimum number of variables the thing is you will not be able to describe the system dynamics completely anyway.

Suppose you really require let us say you really require R and theta, by selecting only R theta information cannot be extracted from there, so that is a drawback from here if you select lesser than what is required you will not be able to describe the system completely, but if you select more than the number of say I mean variables required there are there can be two issues actually. One issue is computational complexity, that means if you really require 3 and you have selected 5 then instead of dealing with a 3 by 3 matrix you have to deal with a 5 by 5 matrix, that in the sense of a matrix and all that in a linear system we will see we will talk little while later actually.

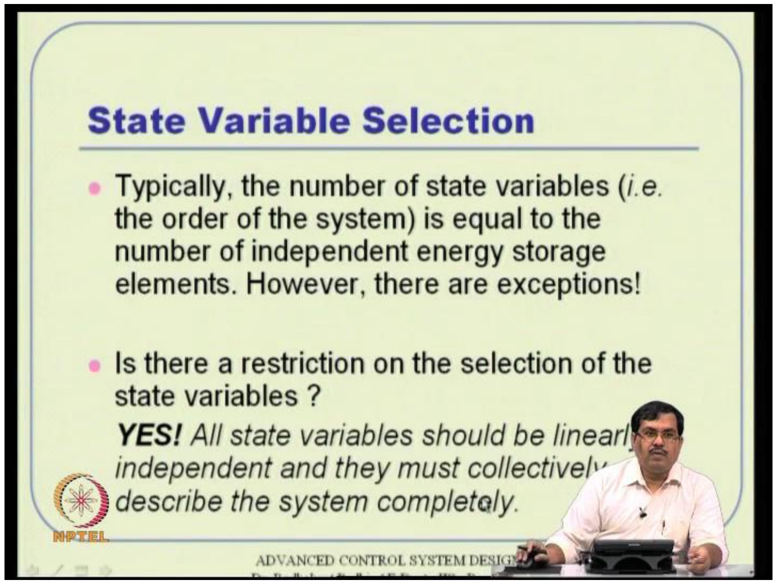
But that is a small problem, the even bigger problem happens to be like if you select more number of variables then necessary you will lose control ability and observed ability property, which is even more dangerous actually, what is control ability and observed ability we will see it in a in a later class actually, if the system is not controllable then you cannot design a control system at all, if the system is not observable you cannot design an observable at all actually, and hence a filter also, in that sense it is more dangerous not only in computational complexity point of view, but with the danger that you will be able to I mean you will be leading to losing either control ability or observability or both, that that more dangerous actually.

So, that minimum number of variable so is a necessary requirement actually, then linear independence is also required, because if you lose linear independence that the variables that you select are linearly dependent then two things can happen, one thing obviously can be slightly but still it is bad, that is it may not be possible to solve for all other system variables, right if it is linearly not independent then you cannot really solve for all of the variables actually.

But even worse that what will happen is it may not be possible to write the complete state equations actually, so these are all requirements because if you see this x 1 and x 2 if the moment I select x 2 along x 1, then theta information is just not possible actually, I mean this I will not be able to convert it in a proper way, because see x 2 is contains almost the same information as x 1, x 2 does not contain an independent expression actually information what I mean, so those problems will start coming actually.

So, what you need to remember is minimum number of variables you have to select, if there is a more than that then there are computational complexity as well as danger of losing control ability observability, if it is less than that we will not be able to describe the system completely, and linear independence is a requirement, because if it is the system variables that we select a state variables are not linearly independent, then there can be a bad issue that means it may not be possible to solve for all of the system variables, there can be even more dangerous issue that it may not be possible to write the complete state equations actually, so these are all things to remember actually.

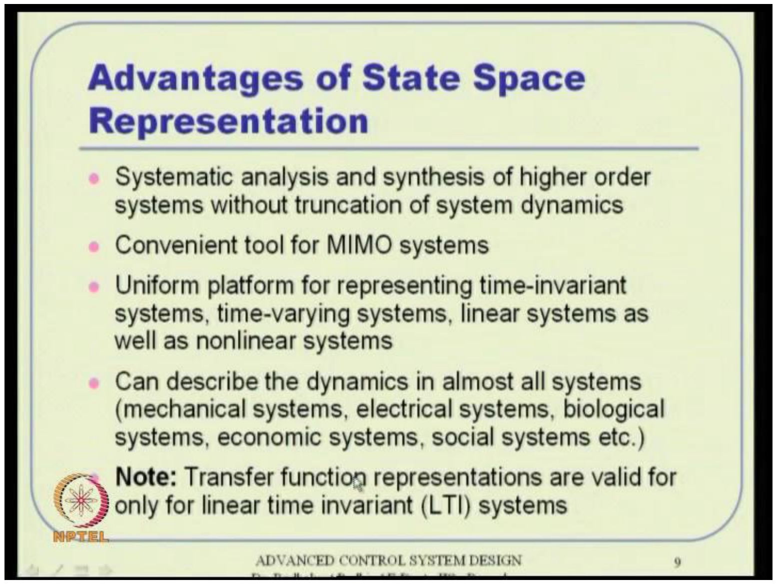
(Refer Slide Time: 16:49)



Now, how do you select like state variables, typically e the number of state variables that is the order of the system is defined by that as well is equal to the number of independent energy storage elements, that means if you talk about a I mean there are concepts like which of the elements can store energy, and which of the elements they are passive element they cannot store energy like that actually, can study more details so that in no one has probably that is that is written there.

But you can but the unfortunate thing is there are exceptions actually, that means this need not be a necessary condition basically, that can simply give you a hint if you have that many energy storing elements then that means instead are there actually. So, is there a restriction on selection of state variables yes we have already discussed about that, this should be linearly independent, and they must collectively describe the system dynamics completely that is we have all ready discussed about that actually, so more details on that you can find in the text book as well.

(Refer Slide Time: 17:56)



Now, before proceeding further let us see what are the advantages of state space representation, there are several advantages over classical things actually transfer function what I mean, so what state space representation gives us is a systematic analysis and synthesis tool for higher order systems, and without any truncation or approximation of system dynamics, whatever may be the system dynamics complexity either from the order of systems or from the from the point of view, the non-linearity is involved or saturation thing is involved whatever it is, it is possible to describe the system dynamics completely using the state space representation without any approximation without any truncation of the system dynamic, that is why it is a very powerful frame work actually.

So, most of the things most of the modern control books and modern control philosophies are based on this state space representation, they normally do not talk even transfer function analysis, and it is obviously a convenient tool for analyzing and synthesizing MIMO system multiple input and multiple output system, remember there is a transfer function means we always talk about C s by r s sort of thing, which is like single input and single output representation actually, there are there are extensions to that where you talk a transfer function matrices that we will see is next class next class also.

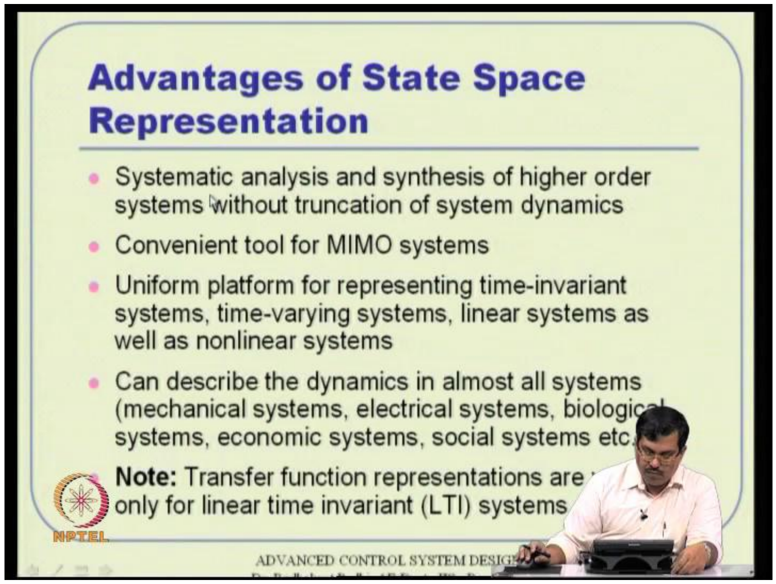
So, as long as the system is linear then in input output sense, I will still be able to I mean take advantage of transfer function and I would be still talking about transfer function matrices that is still possible, but if it the moment it is a non-linear system I will not be able to do that, no matter what this a very convenient tool for MIMO systems actually, the state space representation.

Third point it is a uniform platform for representing whatever systems you talk about actually, time-invariant, time-varying, linear, non-linear whatever systems we talk about this state space representation can do the job for you actually, and that is extending that from application point of view it can describe the system dynamics in all most all systems, you can take example through mechanical system, electrical system, biological system, economic system, social system no matter what actually.

Whether the whether the system dynamics comes from some basic physical laws, like Newton laws or law and thing like that, or it is simply it I mean simply found from experimental data analysis which is typically done in biological systems actually, that you carry out several experiments and try to fit models for that, there is no physical explanation but, still the model is capable of representing the dynamics actually.

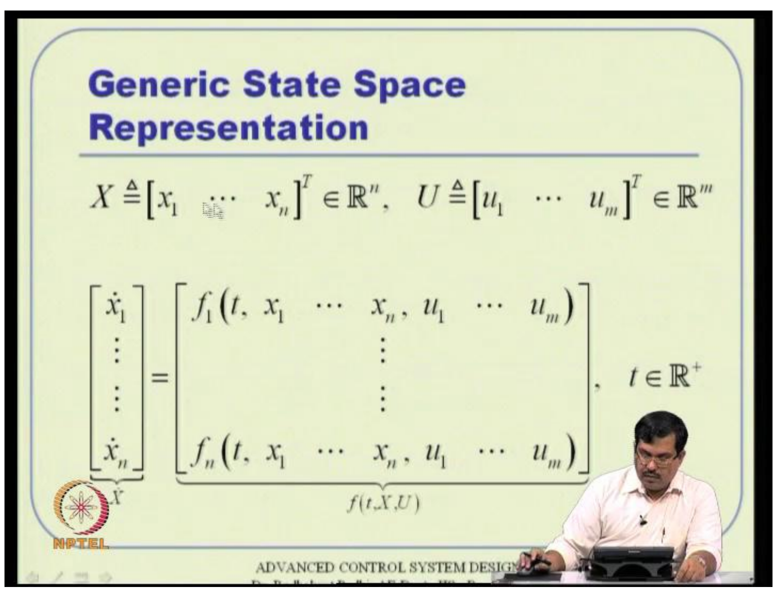
So, all sort of things is it possible to describe in the frame work of non-liners, I mean this state space representation actually, and just remember the transfer function representations are valid only for linear time invariant systems, I am not purposefully put s I s o single input single output, because of that reason we can still talk about transfer function matrices. However you can always remember that it is always valid only valid for linear time invariant system, the system has to be linear system and the parameters of must not be time varying, this a time invariant system then only you talk about transfer function, otherwise the transfer function representations are not that attractive actually.

(Refer Slide Time: 21:40)



Now, generic state space representation let us see, I mean most of the physical systems by the way all the example that are going to follow will primarily confined ourselves to mechanical system, that is where aerospace engineering falls actually, whenever the examples will fall pro mechanical system however similar analysis can be done for all these electrical, biological, economic, social all that actually.

(Refer Slide Time: 21:56)



So, generic state space we will talk about let us say state vector which consists of it is a column vector consists of x 1, x 2 up to x n elements, and all these elements will assume their real numbers actually, so this state vector is typically belongs to some sort of R n dimension R n space, and then similarly, the control input that we I mean that we account for is u 1, u 2 up to u m that is a m dimensional space, and these are also real numbers for physical systems, so each of them have to be real for physical system, each of them has to be real for physical systems for states also.

So, without further like assumptions of telling all these by default throughout these course will may will assume that state variables I mean state vector belongs to R m space it is n dimensional vector, and control vector belongs to R m space that means it is a m dimensional vector, that is implicit whether we tell all the time or not actually, straight vector is a n dimensional vectors, control vector is m dimensional vector, and all of these elements are actually scale or values as real numbers actually.

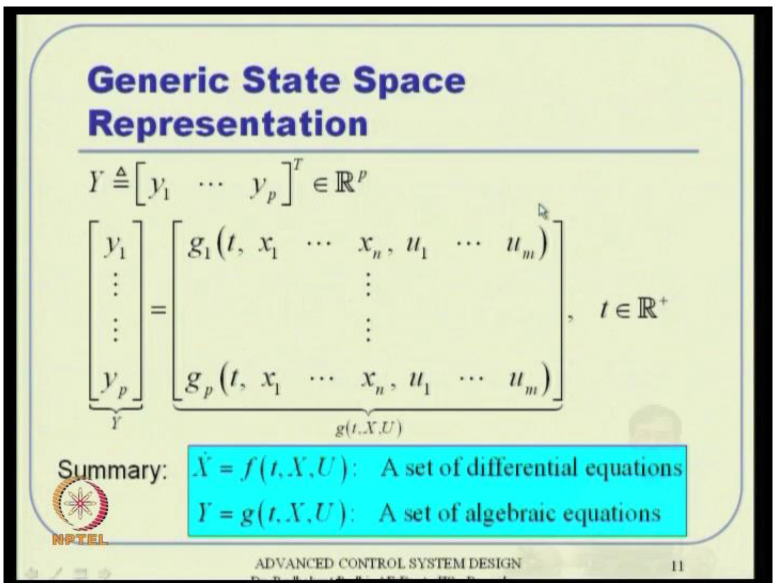
And also 1 more I mean 1 more point that if it is written in a small variable small letters that means it is a scalar component, if it is a big letter it is a vector component, that is that is also a some sort of a notation will follow throughout this course actually, if it is a big u that means by default it is a vector the small u with a substitute u 1 and u 2 all that these are all scalar components actually.

Now, if you define like that the state space representation of the system dynamics of a system in general can be represented something like this, x 1 dot is a function some function of time t, remember t is an independent variable which belongs to R plus only it is a positive quantity by default again, no restriction issues but we will to take time as a positive quantity, and specially some applications we require backward integration we require some negative information as well actually, but we by default we will take it as a positive number actually, so we are assuming this system dynamics are always propagated for adding time basically.

So, x 1 dot is a sum non-linear function of t, and then all these variables can come in can be any function consisting of x 1 to x n, and it can also be a function which is this u 1 to u m, some genetic function f 1 some non-linear function will represent this x 1 dot, and similarly, x 2 dot, x 3 dot all over up to x n dot will be represented by some sort of a system dynamics f n, which can again be a function of t and then all states and all controls actually.

So, in a vectorial notation x dot is nothing but f of this t x u, remember this when you talk about this f will still write small f, assuming that it is still a vector actually, the reasons for that the moment you write big f there are notations of the vector becoming a matrix really actually, so as long as long as the f is concerned will still use a small f all the time actually, so x dot equal to f of t x u that is the that is the genetic function in general, that is a state space representation for any non-linear system in general actually.

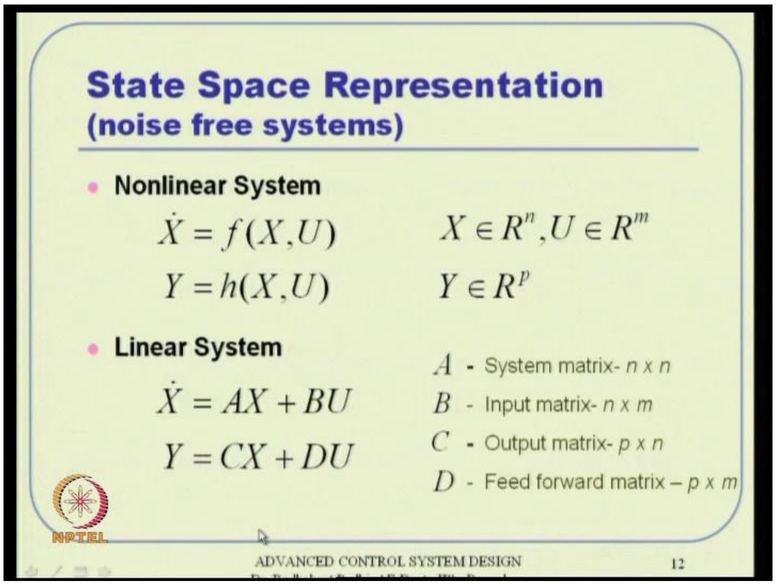
(Refer Slide Time: 25:18)



Now, coming to the output variables will still again similar to state in state vector and control vector, will assume that this is like y is a p dimensional vector again these by default I will assume that is a pre dimensional vector, so it consist of y 1 to y p it is also again a column vector, and all these variables are also will assume these are real numbers actually, and then these are represented something like this it is some very close to what is going on here x 1 dot is something x 2 dot is like that, but here it is y 1 equal to g 1 y 2 equal to g 2 like that actually.

So, it is actually a pre dimensional vector y 1 to y p, and given by g 1 g 2 up to g p actually, so in general it is y equal to g of t x u not y dot remember that, so it is x dot equal to f of t x u, and y equal to g of t x u, f of t x u and g of t x u are algebraic equations, so obviously the first one is a set of a differential equations and the second one is a set of algebraic equations.

(Refer Slide Time: 26:23)



So, in general state space representation and mostly we will talk noise free system, because noise input is typically ignored as far as control design is concerned, what that is considered as far as filtering designed is concerned, which we will not talk about here actually, like a colemon field of example and all that actually, otherwise these 2 rely on the philosophy of separation that you cannot you need not talk about noise input directly.

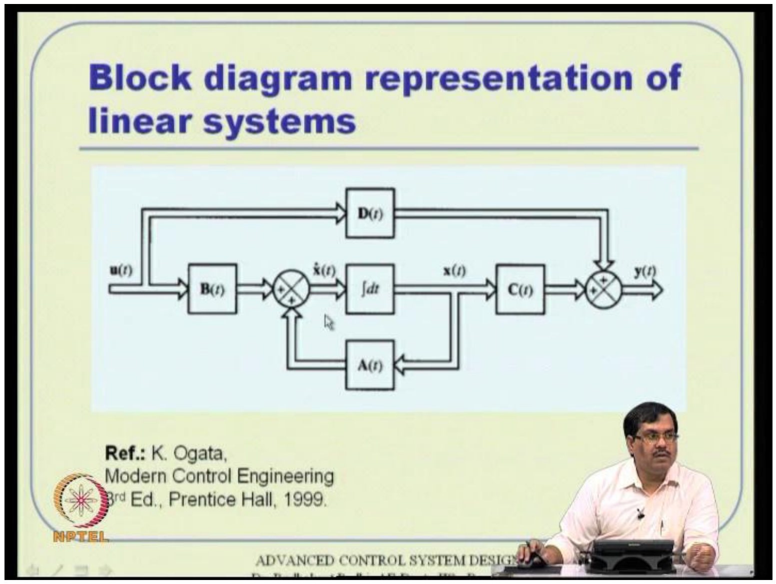
In other words you can design a controller taking I mean assuming is noise free system, and you can also design a filter assuming no control input, and then you can put them together and that is a separation principle theory and all that specially for linear systems, if you have a linear system these two can be proven that you can you can do that that way on a non-linear system also people do that actually. So, as far as control design is concerned we will invariably take some sort of a noise free system throughout and see what we can do actually, so for non-linear system that is the form that we have already discussed, sometimes people write is not g but h g actually, whatever you see as g you can you can write it a h, because this g is reserved for some other thing I mean if you talk about something like control affine system then people talk about x dot equal to f, f of X plus g of X times U so g is reserved for that actually, so that is that is the reason why sometimes people tell no we will use h actually instead to avoid that conflict.

You know in general we will talk t does not come explicitly most of the system, we will see the t really not does not appear unless otherwise you are forced to take it that way for example, rocket trajectory when rocket goes up and up the amount of propulsion that is consumed is huge, and that is typically given in terms of thrust and curve actually, like the m dot mass flow rate time verses time curve.

So, in those situations t will come or most of the time the t does not come explicitly, most of the time the parameters remain fairly constant, so we will not be able to account for this time dependent explicitly it is not needed with basically, that is the non-linear system representation, and what when you talk about linear systems they are mostly linearised system about some operating point of this non-linear system, naturally the system dynamics are non-linear in general, so we still talk about linear systems, but remember always that these are linearised system and we will see one class how to lineraised and all that actually, we will starting from this non-linear equation.

So, this linear system dynamics is given something like this, and then this is x dot equal to AX plus BU and y equal to CX plus DU, A, B, C, D is given like this actually, and A, B, C, D are typically corresponding appropriate matrices of dimensions that are compatible for example, if you see X dot is n dimensional vector X is n dimensional vector, so obviously the A matrix has to be n n by n matrix similarly, B has to be n by m think like that actually.

(Refer Slide Time: 29:30)

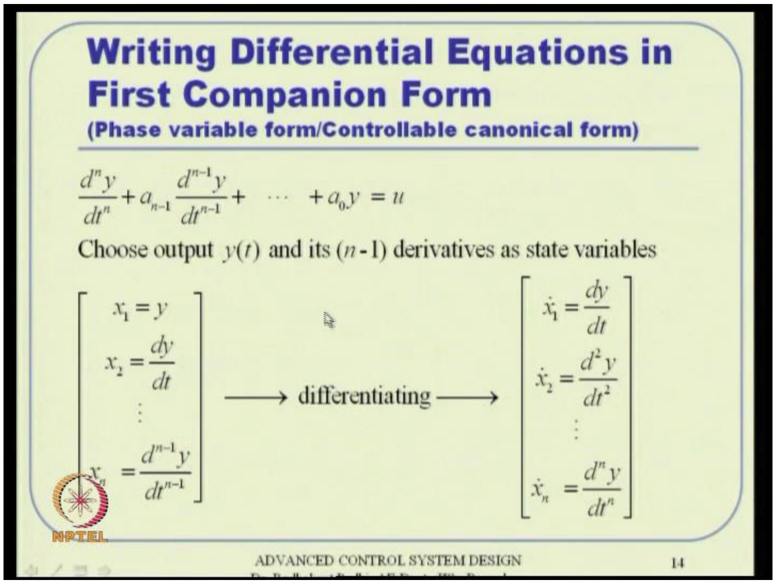


Now, how do you represent it in a block diagram representation, this is something like this like suppose you take this u sometimes in some textbooks you will see bold letters actually, like bold letter means vector and then non bold letters means scalar and thing like that actually, so this is like u of t, remember this we are representing this linear system here, so u as a we are trying to kind of replicate this two equations X dot equal to x plus BU, and y equal to CX plus DU, so what is happening here let us construct this x dot what happens here actually, so if I start from this integration loop there is x dot remember these are set of integrators, say n integrator sitting there, so x dot comes component by component goes out component by component, I collect this vector and then collect this vector as well actually here.

Now, once I have X X of t X dot is what X dot t is AX plus BU, so I will construct X pass it through a then add it up actually, so this a x component coming here then there is A B and there is a u then B u component comes here, if I see X dot, X dot equal to a times X here plus B times u here, so that that is constructed AX plus BU, now if I consider y is CX plus DU so y is here, so I will pass it through C matrix so that will cross the that will give me C f C f x here, and then I will pass this U through d of t matrix, so I will give me sort of a CX plus DU representation here, and in general remember this matrices A, B, C, D are all time dependent things actually, so that is why you see time varying and thing like that here.

Otherwise these are if we can suppress these brackets, this B of t a of t and all that you will get a L t h system anyway, but still remember even if you suppress a of t, B of t, C of t and d of t you will still be left out with u of t x of t and x dot of t that is our system variable that live all with time actually.

(Refer Slide Time: 31:35)



Anyways now that is a philosophical representation of system dynamics, now we will see exactly let us say you are start with a differential equation this let us say it comes some sort of a physical representation, in other words for a simple example you can talk about m x double dot equal to f actually we know that, so if you something some representation of nth order like that then how do you represent it in a state space form, and very frequently we will come back to that cosine n gain and gain actually.

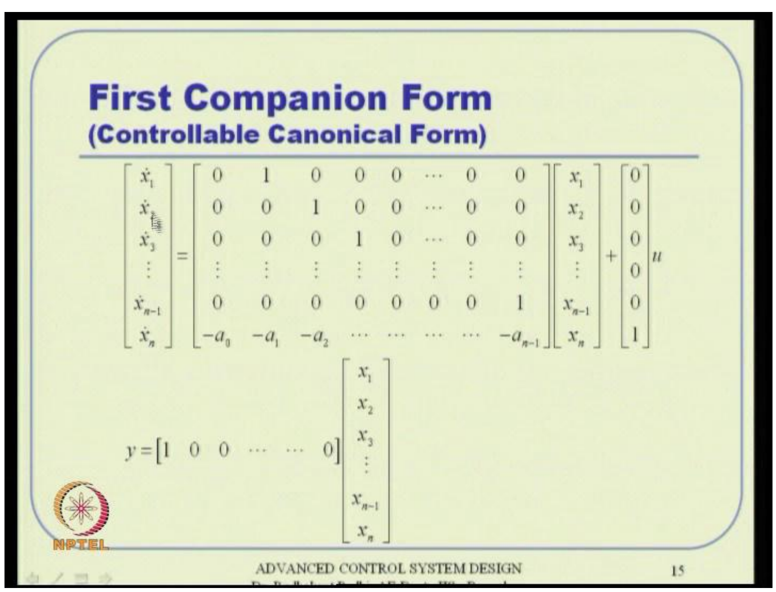
So, let us let us know how to do that, suppose this is a system differential equation again y is a scalar and u is a scalar, but still it is given by nth order differential equation in the left hand side, so because it is nth order we will really require n system variables to represent the system dynamics completely, and in a first company and form which is also called as phase variable form or controllable canonical form.

So, in that sense what you define is x 1 equal to y, we start in a reverse order sort of thing then x 2 equal to y dot or dy by dt, we all the way we go to x n and x n is this portion this quantity, this n minus 1th derivative of the output actually, so you start with the 0th derivative, first derivative up to n minus 1th derivative, so these are something like system variables that will give me some sort of n variables actually, so that I will differentiate it and take it through then what I this definitions here x 1 and x 2 and thing like that will constitute something like some sort of a system variable definition, but after that what are the dynamics involved.

So, I will differentiate x 1 by 1, let us say x 1 dot actually I differentiate that, then that is nothing but y dot, and y dot is dy by dt and hence it is x 2, so x 1 dot what you see here is y dot y dot is dy by dt and that is x 2 by definition, so x 1 dot is dy by dt x 2 dot is d square y by dt square all the way you will find that up to x n minus 1 dot there is no problem I mean you can just write it from definition, x n minus 1 dot is nothing but this quantity right so that way it is it is actually.

However if we see the last equation that is that will consist of x n dot, x n dot is dy n sorry d n y by dt n that that expression actually what you see here, you have to write it all in the right hand side, and then try to see what all variables they take actually.

(Refer Slide Time: 34:12)



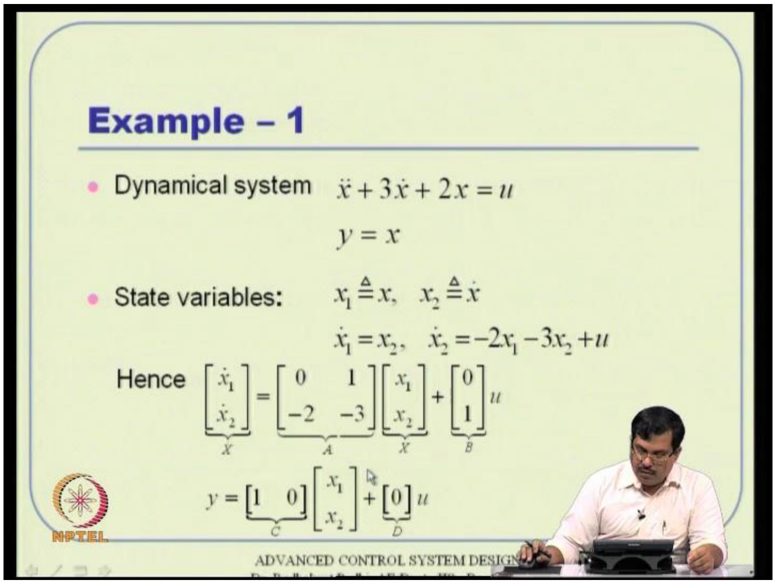
So, for up to first n minus 1 derivative that come from simple definition, x 1 dot is x 2, x 2 dot is x 3 like that actually, so x 1 dot is x 2, so that is why 1 here x 2 dot is x 3, that is why third column I mean third column second row is 1, and then thing like that we will consist of up to x n minus 1 dot, that all these things are coming simply from definition, and also along with that 0, 0, 0 will go here because the system dynamics does not contain any higher order I mean any stay I mean y is not a function of y y dot and thing like that actually.

Now, coming to the last equation x n dot x n dot is all these things you put in the right hand side and you see that it is a minus h 0 times y and y is x 1, so it is minus a 0 times x 1 then minus a 1 times y dot y dot is x 2 like that actually, so it is minus a 0 times x 1 minus a 1 times x 2 like that actually, so minus a n minus 1 times x n, and then there is a u component here that is 1 into u, so it is that last component will come as 1.

So, this will consist of your a matrix, this is x dot equal to a times x x is here a C plus B times u, b can be vectors or matrix depending on how many controls you talk about, be a single control it is turns out to be a vector obviously, so and then if you talk about a output and let us say primarily my displacement is what is my concerned that y whatever I have then y is nothing but x 1 only, so while talk about y is equal to x 1 but I have to represent in the in terms of C x plus DU form, then I will consist this C of x matrix means C matrix that way 1, 0, 0, 0 it is a rho vector here, followed by the same state space vector the same state vector, plus DU and d 0 obviously here, so you got a matrix you got B matrix you got C matrix that way.

So, starting from this nth order differential equation, it is possible to write a n dimensional state space representation this way basically, and that will come back to this I mean idea of representing states variable form transfer function also in x plus actually, the variety of ways to do that by the way.

(Refer Slide Time: 36:36)



Now, we will see a variety of examples which will convince us that state space representation is more generic, and it can be possible to write lot of different systems in a unified manner actually, so let us start with the simple equation, which kind of corresponds to the example that we just discussed, so this is this particular example will be something like this, x double dot plus 3 x dot plus 2 x is equal to u, and I will simply take y equal to x.

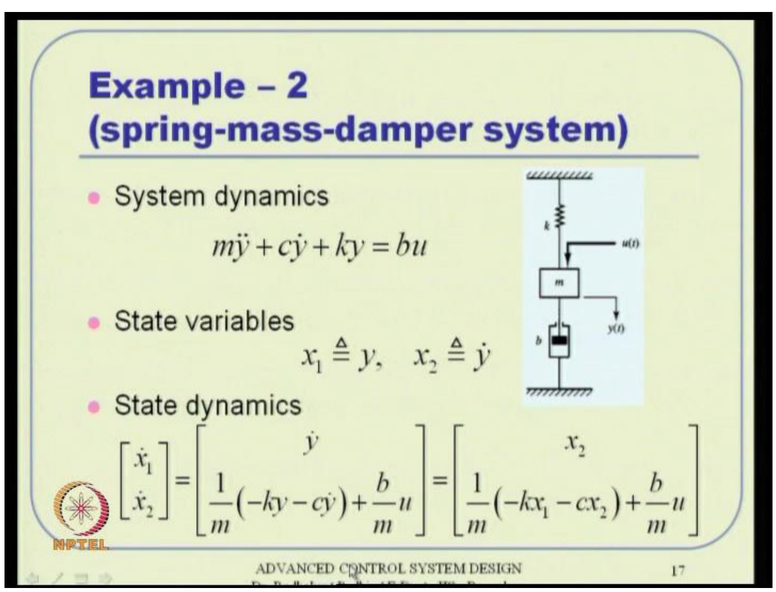
In that sense again going back to the phase variable form what I will define, I will define x 1 equal to x and x 2 equal to x dot, I will the reverse way I will stop it n minus 1th derivative as far as definition are concerned, because I need n variables actually so it is a it is a second order system so I need 2 variables here x 1 and x 2, now when I when x 2 sorry the when this differentiation of x 1 that is x 1 dot is x dot and x dot is x 2, so that will come simply by definition, but x 2 dot I have to go back here and then put it in the right hand side it is a this is minus 2 of x that is y minus 2 of x 1 because x is x 1, and minus 3 of x dot so that is minus 3 of x 2 because x 2 is x dot.

So, x 1 dot is x 2 simply from definition, and x 2 dot is minus 2 x 1 minus 3 x 2 plus u obviously u is the this side, so because this a linear system I will be able to write it in the standard form and hence I will put it in a vector matrix representation, I will put the same equation what I see here x 1 dot is x 2 that means 0 times x 1 plus 1 time x 2 plus 0 times u, so that consists of my first row and x 2 dot is minus 2 of x 1 minus 3 of x 2 so minus 2 and minus three here plus u that means 1 u, so this is my a matrix that is my B matrix.

And what about y, y is nothing but x x is x 1, so I will put the matrix C as like this 1 0 x 1 and x 2 here the same state, and then you have a 0 times u, so this is your a matrix; this is B; this is C; this is d actually, that is how you construct this state space representation again remember that this state space representation is certainly not unique, somebody can always argue that I will define x 1 as x dot and x 2 as x in a reverse order, and hence you'll represent a matrix representation in a different way, that means your A, B, C, D matrix will turn out to be slightly different.

However the nice property in linear system is like no matter whatever representation you come up with they are all similar actually, and nicely they are also tied up why is what is called a similarity transformation, will see that as we go along actually, different matrix forms wherever you come up with they are all tied up with similarity transformation, and similarity transformations preserve Eigen values, and Eigen values are poles of the system, that means the stability nature and all that are not perturb by whatever way you write actually, so the system variable definitions can be different but system property remain actually, that is the that is the nice thing about this type of analysis.

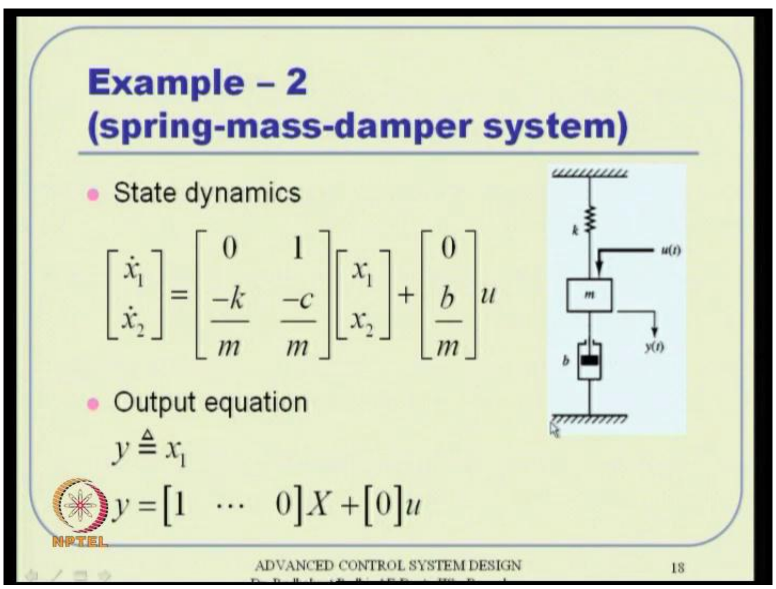
(Refer Slide Time: 39:55)



Next example let us let us be slightly more practical, and then talk about a spring-mass-damper, this is a building block in mechanical systems by the way; a small components we can always represent in an approximate way by a spring-mass-damper system; very close to what example we saw this is also a spring-mass-damper system in a way, but it is more theoretical, but this is slightly more practical because we really talk about physical quantity for m, physical quantity for c, damping factor I mean damping coefficient, physical number for k spring constant, and then there is a control input which is actually coming to the system through some sort of multiplication with B system parameter.

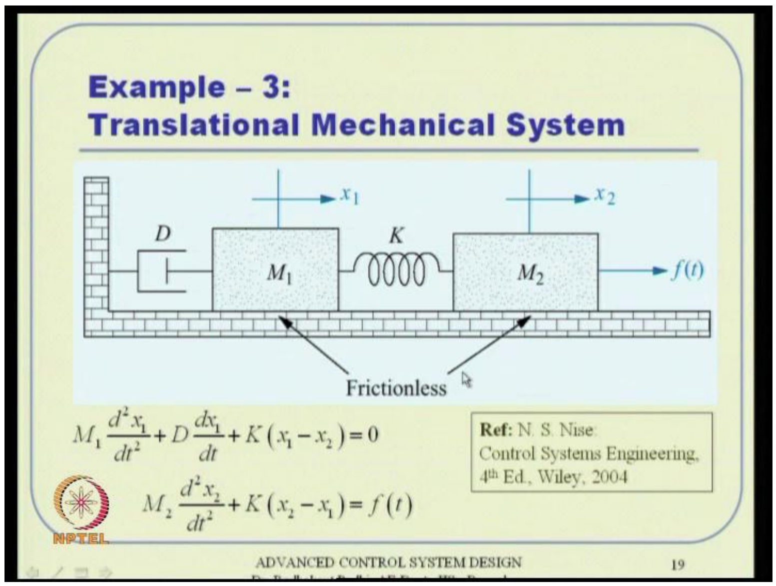
So, this m, c, k and b are nothing but system parameter, and they may or may not vary with time actually, and mostly they for all practical purpose or control design for a spring-mass-damper system specially we will can assume that these variables are fairly constant actually. Anyway, so how do you go about it? You will see again go in a very similar way you define x 1 as y and x 2 as y dot, and again come back with these equations that way; x 1 dot is y dot and y dot is x 2 by definition; and x 2 dot is all these whatever is in the right side divided by m of course, that is your y double dot, and then you represent it in terms of your x 1, x 2 variable sense and all that actually here.

(Refer Slide Time: 41:22)



So, then again because it is a state I mean it is a linear system we should be able to write it in a state space form, so in general the system matrix will consist of this matrix in general and that may that vector in general actually, so that is a just an extension of what we discussed in the previous example.

(Refer Slide Time: 41:38)

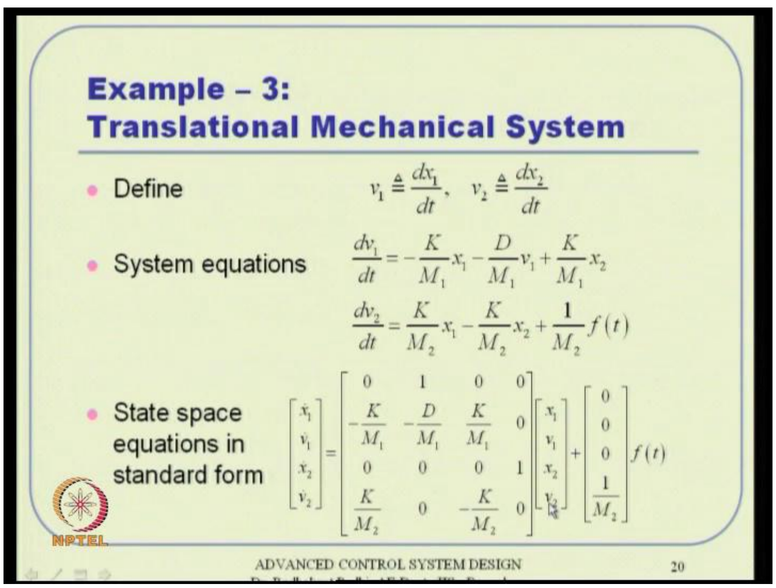


Now, coming to little more I mean little more complex system, we can we can imagine this like some sort of a double mass connected by a spring and a damper actually, in a very rough way you can assume this to a some sort of a train compartment basically, the very rough way, like there is a engine sort of thing and there is a compartment basically like that way.

So, I mean it will consist of n number of systems by the way it is not just not 1 actually, but we can think of that as an engine pulling and then there is something like a spring and damper that can makes to the second one actually, so then this how do you how is it represented I will assume the truck is frictionless sort of thing, and then it will turn out that if I take three body diagrams which is very popular in mechanical system analysis for both M 1 and M 2, then from the free body diagram of M 1 I will be able to write this equation, and for the free body diagram of M 2 I’ll be able to write this equation, and for most of this particular course will assume that system dynamics are kind of given to us, who will not to worry so much about how the system dynamics comes from where it comes, what the physical relationship and think like that, however we will also remember that these are coming from certain basic principles of nature.

Essentially these are Newton's law what you see here, mechanical systems are primarily they'll come from Newton's law actually, so for M 1 I will write the system dynamics from the free body diagram of M 1, I am just equating the forces that is all various components of forces I will take, and then there is no particular force that pulls this force explicitly so that will become 0 if and then there is a particular force that pull so that will become f of t actually, but remember as far as M 2 is concerned the d does not come actually, d comes only to M 1 damping factor actually, so that will that is embedded in this equation actually.

(Refer Slide Time: 43:37)



Now, again similar trick I will define v 1 as dx 1 by dt that is the velocity of this M 1 that is v 1, and then v 2 is dx 2 by dt which velocity of that, so x 1 and x 2 define the position of this M 1 and M 2 at any point of time, and v 1 and v 2 define the velocities of this bodies at any point of time, so then I will talk what is my dv 1 by dt and what is my dv 2 of dt, you know these are not now they are not independent quantities anymore actually, the way this x 2 is coupling here you see that this x 2 term here; there is also x 2 term here, that means if I pull the if I vary this force f of t my x 2 is going to change and hence my x 1 is going to change because there is x 2 input also coming there actually.

So, these are coupled systems basically and that is why you see m I m o system, there are even if there are multiple word is like this I will able to still describe the M 2 get there basically, so what I am going here what I am doing here I am analyzing what is my dv 1 by dt I will represent I will find that out from his expression, that is my dv 1 by dt and similarly, this is my dv 2 by dt so that is what I'll take it to the right hand side.

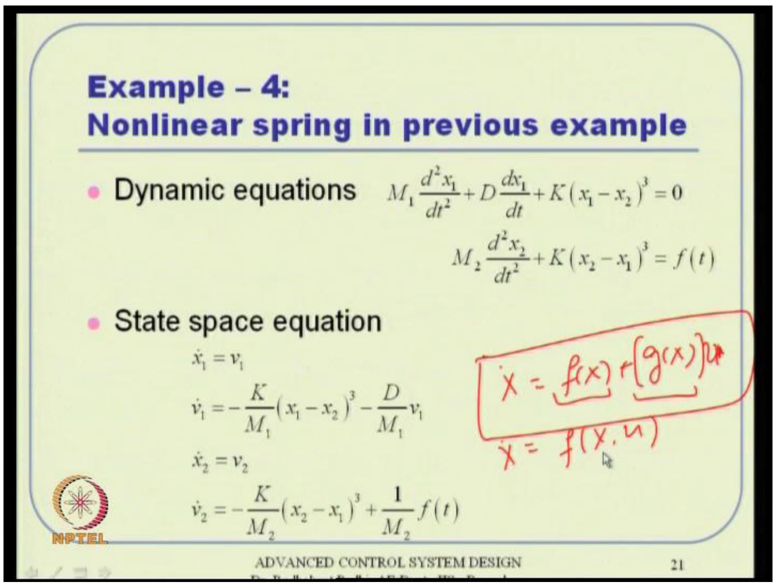
Now, once I do all these things, I will be again able to do this in a state space representation form, so x 1 dot is nothing but x v 1 so that is 0, 1, 0, 0 similarly, x 2 dot is v 2 and that is why it is 0, 0, 0,1 so v 2 is fourth element here, and also remember even if you define this variables that you have the freedom to put the state vector in whatever order you want actually, that also features the properties anyway, you do not have to really write x 1 below that v 1 below that x 2 and thing like that somebody can always write x 1 below that x 2 below that v 1 below that v 2 that is also possible, but then these entries have to be put in a careful manner, wherever whatever way you define the corresponding elements necessarily put at the appropriate place actually.

So, in this in this definition and of the variable and this definition of the state vector I have got this type of matrix here, which is m I a matrix and this is my v matrix here, so corresponding look at the equations carefully, x 1 dot is v 1 so that is why this is 1 x 2 dot is v 2 that is why there is a 1 over v 1 dot is like this whatever you see here, that is a minus K by M 1 into x 1 so that is why this minus K by 1 here; minus d by M 1 is here; and K by M 1 here.

But this element is associated with f of t and f of t is something we interpret as a control variable, so this 1 by M 2 will go to the control matrix actually control influence matrix, the B matrix is typically by the way this if you go back to this equations I mean in general this is what is called system matrix; this is what is called control influenced matrix; and this is what is called an output matrix and thing like that actually.

So, anyway this is we coming back to this so this is what we are discussing, so this is my a matrix that is my B matrix, so these are what we see is if starting from a simple linear system as long as whatever order is there we will be able to write it in the form of a x plus B u, even if they are coupled together, they need not be only 1 at a time basically, if I see this a B matrix I am talking the dynamics of both M 1 and M 2 together, so I will be able to analyze the system dynamics in a coupled way basically.

(Refer Slide Time: 47:07)



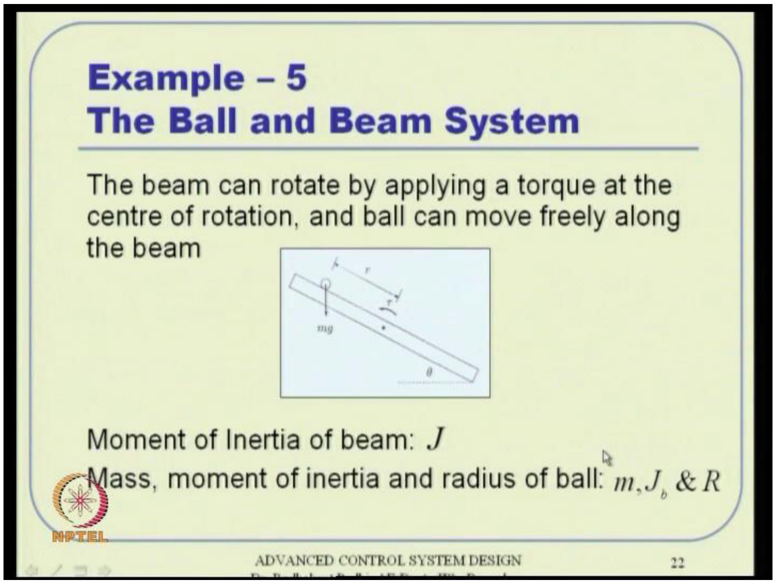
Now, we will move on to non-linear things then we know the state space representation can very well depict non-linear systems as well, to begin with I will assume that the same previous example however the spring what you see here is actually a non-linear spring, sometimes these springs are designed that way to fasten the response and things like that actually, so there are so nice properties of cube cubic spring equation, that means if you see this force that is generated here is a not a function of linear expression like the but it is a function of cubic expression.

So, this is what you see here, so the system dynamics takes the form of q here, earlier it was just this with the power 1, now it is power 3 both the sides, now we will be able to write this state space equation but you will not able to write in a a x plus b u format that is not possible anymore, but you will still be able to write the system dynamic in a state vector representation form actually, x 1 dot is still b 1 x 2 dot is still b 2 and v 1 dot if this 1 which is still valid, and v 2 dot is this 1 which is still valid, but I will not be able to write it in the form of a x plus b u form, but that that is still because it is it is still a state space representation actually.

By the way this example is possible to write in the form of f of x plus g of x times u, I mean that is a system like x dot what I mean is it is possible to write in the form of x dot is equal to f of x plus g of x which is a matrix times u, this is u, so this is my this is a vector this is a matrix rather that is my u vector actually, so this is a specific form which is called controlled a fine form and all, we will see some point of time down the line in this course probably that what is a there is certain nice properties of that, and then especially in optimal control framework this equations have been studied heavily, but even in dynamic inversion framework these are all such a nice non-linear system nice class of non-linear system for which we can actually talk about close form expressions of control design, that means there would be a formula that will be coming off with the control variable, so that is why it is liked it is studied heavily and all that.

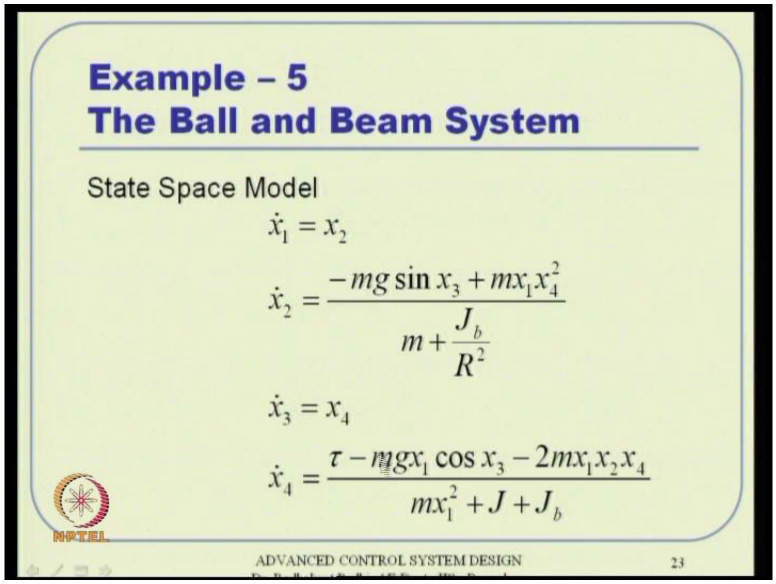
But in general we will be still I mean what we are not discussing that very much here, what we are discussing is x dot equal to f of x u directly actually, so that is still in this equation what you see here in the left hand side is still satisfies the need that I can still represent it is x dot equal to f of x e that way.

(Refer Slide Time: 49:44)



Another little more challenging problem this is something like a ball and beam system, and what would happens here is there is a beam out here which I can tilt it I can tilt it with angle theta, it is like a pen which you can tilt it actually, like say ball beam sort of thing you can tilt that way whatever you want actually, then there is a ball out here which will roll along the beam, and if there is a there is a k hinge out here the ball is I mean about which the rotation takes place obviously, then the ball will roll to the right or the ball will roll to the left actually, depending on what angle you are talking there actually, and will assume the moment of inertia of the beam is a and these are all other parameters that means that the mass moment of inertia radius of the ball and all that will be assuming that these are known to us actually.

(Refer Slide Time: 50:49)

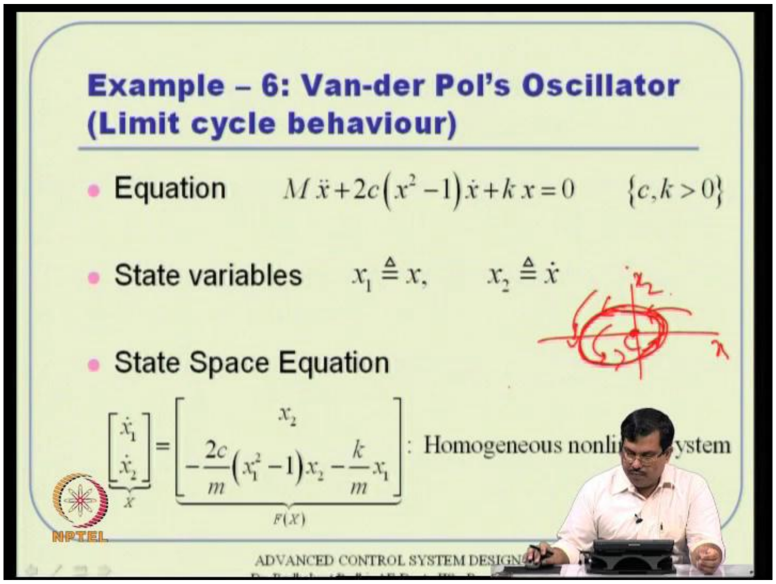


Then if you again go back and carry out the analysis of this dynamics, it will give you some sort of dynamics like this, there is sin theta angle which will come in to picture and these all the details I will skip probably, I mean all that it matters here, it is possible to write the system dynamics of this complicated system completely using state space equation without any approximation.

So, n ultimately it will turn out to be a four dimensional system x 1 dot, x 2 dot, x 3 dot, x 4 dot, and this tau is the term that is a control variable that I am interpreting here, that will control my ball on that actually, so using this I will be able to control the system, in other words I will be able to place this ball wherever I want actually on the B, and I can also dance this ball in a trajectory tracking sense like in what speed it has to go where it has to stop and where it has to come back and thing like that these are all control objectives probably.

So, the message here is it is possible to write this system dynamics in a complete manner without any approximation, remember this expression that we are seeing here is fairly complex non-linear equations actually, sin of x 3 is there; x 1 times x 4 square is there; and here there is x 1 x 2 x 4 term I mean three terms multiplied together there is a cosine of x 3 term there is x 1 square term all sort of things are there actually, but still it is possible to write and possible to analyze things like that actually.

(Refer Slide Time: 52:11)



Next example it is a is interesting example and there is a system dynamics where there is a non-linear expression actually, it is a you can interpret is a second order non-linear differential equation of a specific form, and c k are positive numbers and thing like that and this is actually a small m here, let us say small m is mass of the system and all that actually whatever.

Now, again you go back to the system variable form this x 1 and x 2, so the same standard way x 1 is x and x 2 is x dot and thing like that, so state space equation will turn out to be like that, so what happens here we are not putting any control thing here control input is probably not here, but as long as there are initial conditions available to us we will still be able to propagate the system and all, so these are called homogeneous non-linear systems these no control input explicitly appearing.

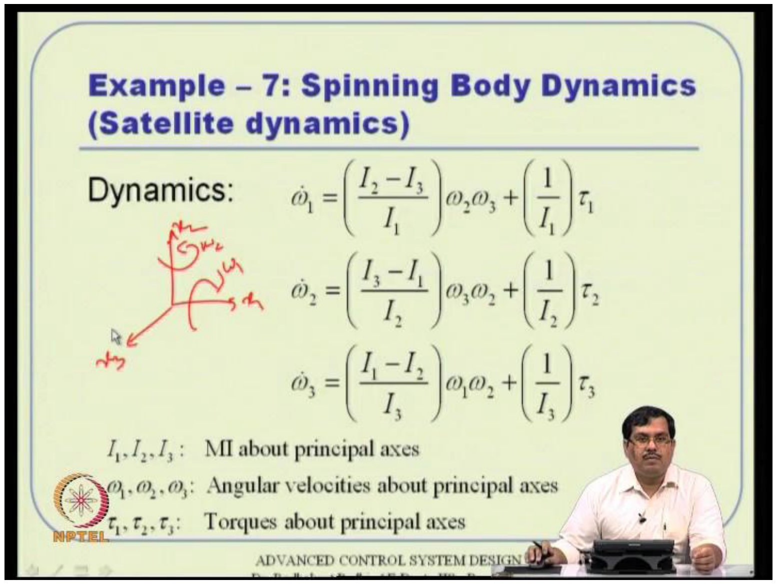
So, this is still a non-linear system, the problem of this system the beauty of this system or whatever you can see if you really analyze this equilibrium conditions and all that means you have to put these to equal to 0, x 1 dot x 2 dot equal to 0, and try to find solutions for x 1 and x 2 will see that slightly later also basically.

Then we will see that there are not unique equilibrium point, that means unlike linear systems linear homogeneous system the moment you put x dot equal to 0 this vector entire vector then x has to be 0, because x x dot equal to x plus v u and u is not there that means x dot equal to a x the moment x dot is 0 then x is 0, that is the only equilibrium point that you have, but non-linear system you can have multiple equilibrium points in general, and interestingly it turns out that it actually exhibits the limit cycle behavior and that is what we discussed in the first equation also, like if you see x 1 and x 2 here x 1 and x 2 if you plot it will keep on revolving.

And there their will be one equilibrium point which is there on the origin that is an equilibrium point, and unfortunately it turns out to be unstable equilibrium point by the way, so in if it is on the equilibrium point it will stay, but anything other than that it will try to converse to this limit cycle but once it converges it just keep on revolving on the limit cycle actually.

So, if you really want to do a good control system design this kind of gives us some sort of a benchmark problem, because anything other than that will again lead us to the equilibrium, I mean lead us to the limit cycle from the outside also it will attract, it will try to merge to the limit cycle inside also it will try to merge actually, so if you really want to get out of this limit cycle and go towards the equilibrium point and your control system should do a proper job actually, that is why these are some of these benchmark problems and all.

(Refer Slide Time: 54:58)



The next example is the spring mass as there is the spinning body dynamics, we can assume that it is a satellite dynamics, that is what the dynamic equations can be represented something like that, remember there will be also attitude kinematics which will be coupled with that, that means the where this satellite is oriented that will come through certain angles and those angles can be like Euler angles can be there are various representation like m R p, there are direction cosine matrix there are quaternion's and thing like that that part we are not talking here, we are just talking about the dynamic component part of it.

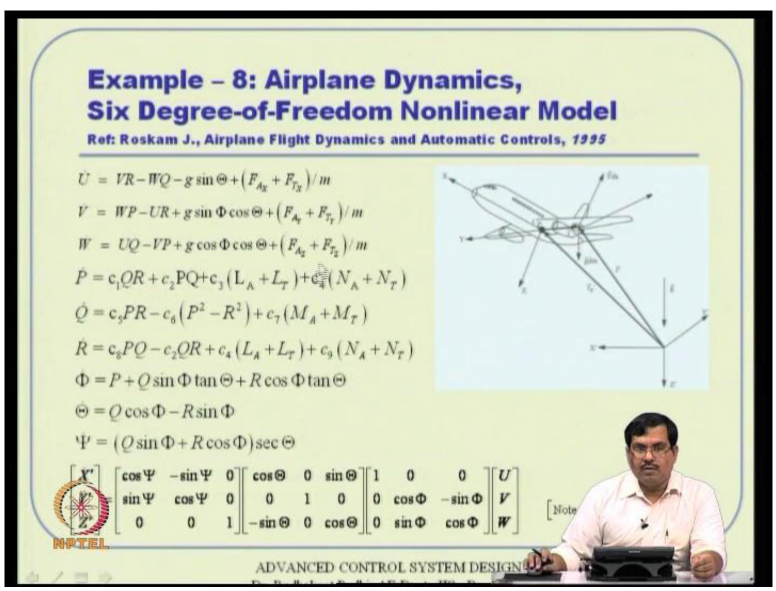
The dynamic component is given something like this, assuming that these are three equations about the three principle access, that means if you take these three principle access something like this x 1, x 2 and x 3, this is x 1; this is x 2; this is x 3 along x 1 there is a about x 1 there is a there is a moment which will be like omega 1, about x 2 there will be a moment that is omega 2 like that actually, then you will be able to represent the system dynamic that way.

And remember this is actually a non-linear system equation again, because there are double multiplication term omega 2 omega three and all that actually, around and unfortunately what happens if you really want to linearise the system dynamics about 0 velocity 0 angular velocity, the first part of the equation simply drops out actually, that means you are left out only with this equation and that is omega 1 dot is this expression 1 by I 1 into tau 1, and omega 1 dot is nothing but theta one double dot, if you take theta 1 dot is omega 1 then it is nothing but theta 1 double dot is this 1 into tau 1, and that is essentially something like a double integrator actually, so these double integrator problems are also available there, but that is only linear expression that may or may not be able to do a job, but the moment you talk about the full system dynamics this is the equation that you have to deal with actually.

So, and linearised system dynamics linearised linearization trick or linear control system will not be able to do the job because of this, this entire term will drop out actually, you will not be able to even see that actually there, so these are the various definitions these are moment of inertia about the principle axes, these are the angular velocity, these are various torques these are the principle torques that are given to this for control purpose actually.

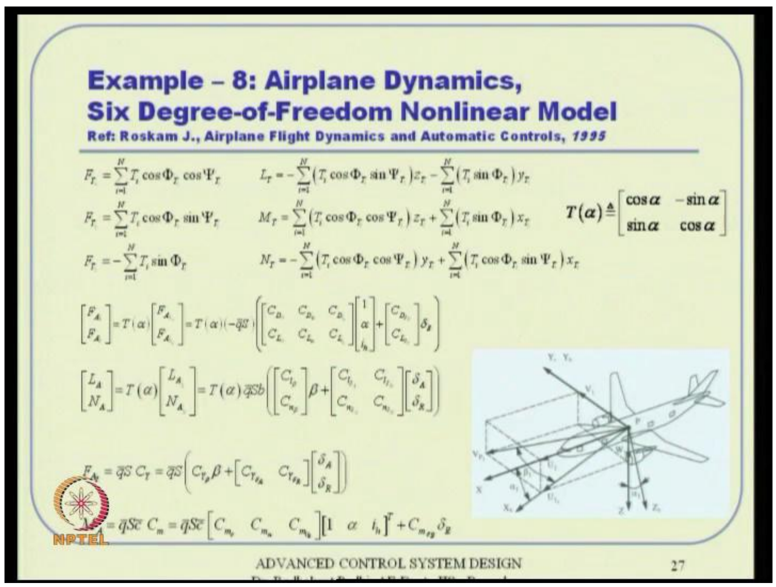
So, it is still possible to write this in the in the state space form together basically, remember this is also a coupled equation, because all the all of the omega 1 dot is coupled to omega 2 omega, 3 omega 2 dot is coupled to omega three omega 1 like that actually, so the effect if you really want to keep then that is here.

(Refer Slide Time: 57:32)



Now, you coming to the last example in the series, this is like a real complicated example airplane dynamics, what you see is lot of variables in the left hand side about 12 equations essentially, and this is all the coupling that goes on here, if you really want to discuss something called 6 degree of freedom non-linear model. We will discuss that in a little more detail in one of the classes later, but this turns out that this if you see this equation carefully is highly coupled.

(Refer Slide Time: 57:59)



And if you see that component carefully also these are all the expression that you have to deal with. So linearization may probably may or may not be able to capture all the dynamics completely and that is where the difficulty comes. So, we want to see this example in slightly little later actually. So that is that is the message can see that various things we can discuss in the framework of this state space equations actually; that is all I will stop thank you