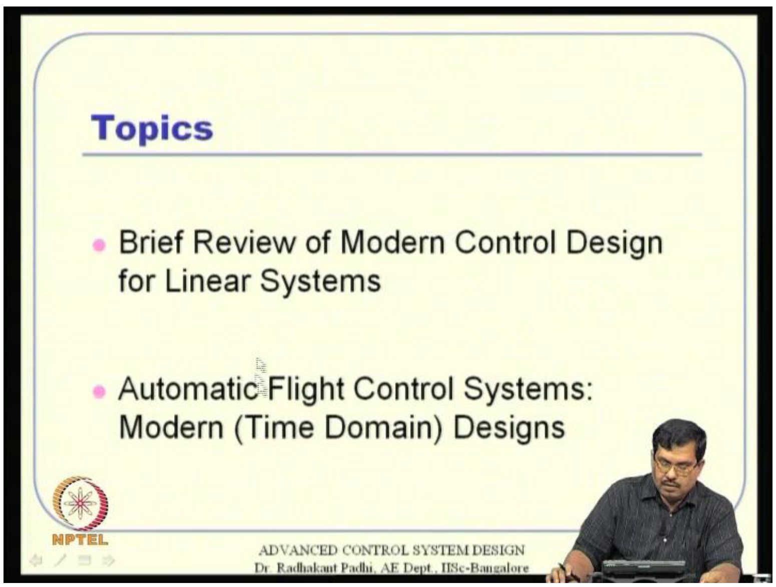
Advanced Control System Design
Prof. Radhakant Padhi
Department of Aerospace Engineering
Indian Institute of Science, Bangalore

Lecture No. # 30
Linear Control Design Techniques in Aircraft Control – II

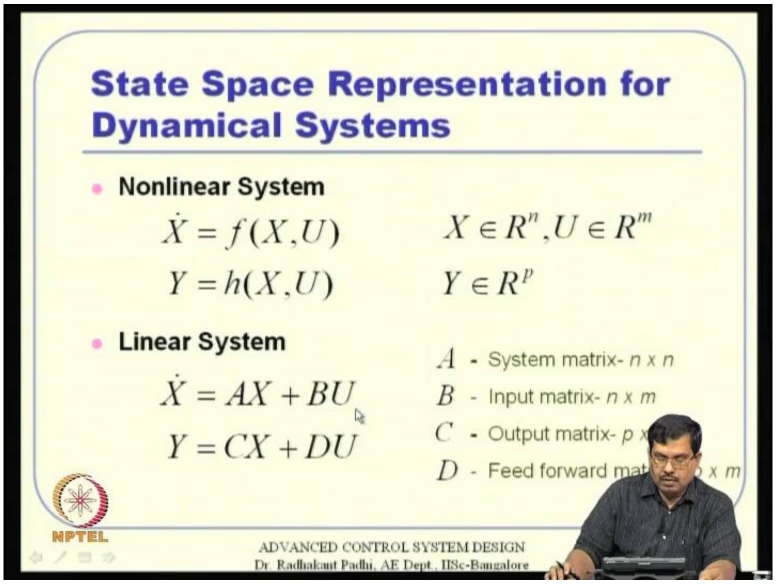
Hello everyone. We will continue with our lecture series. We are at number thirty today. And, continuing with our previous lecture we will, I mean, where we talked about applications of linear control design for an aircraft control, primarily from classical point of view.

(Refer Slide Time: 0:39)



We will continue this lecture from modern control point of view primarily. So, this lecture is organized something like this. We will first have a kind of a brief review of modern control design for linear systems especially. We have all studied about that. So, just to recapitulate what all you are talking here that way. And then, this automatic fight control system, especially from modern or time domain designs actually that way.

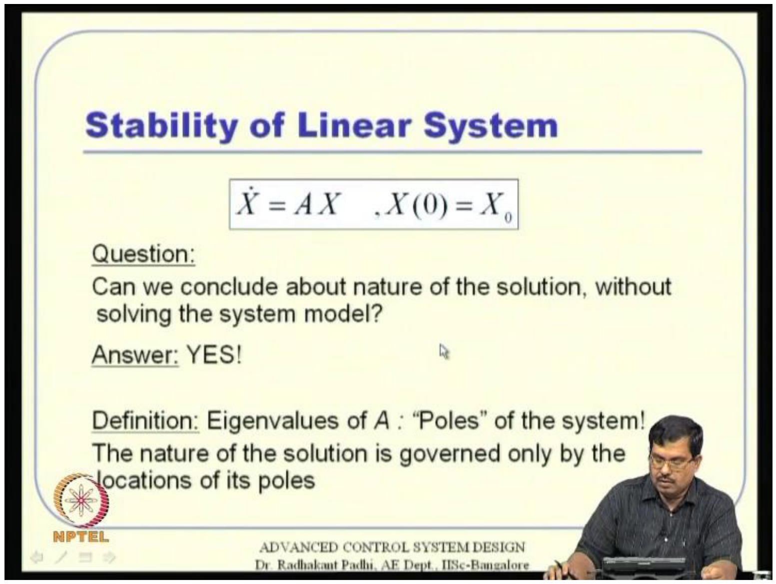
(Refer Slide Time: 01:07)



So, brief review of control design or modern control design for linear system is something like this. You have, this is state space representation. This is the back bone of modern control design.

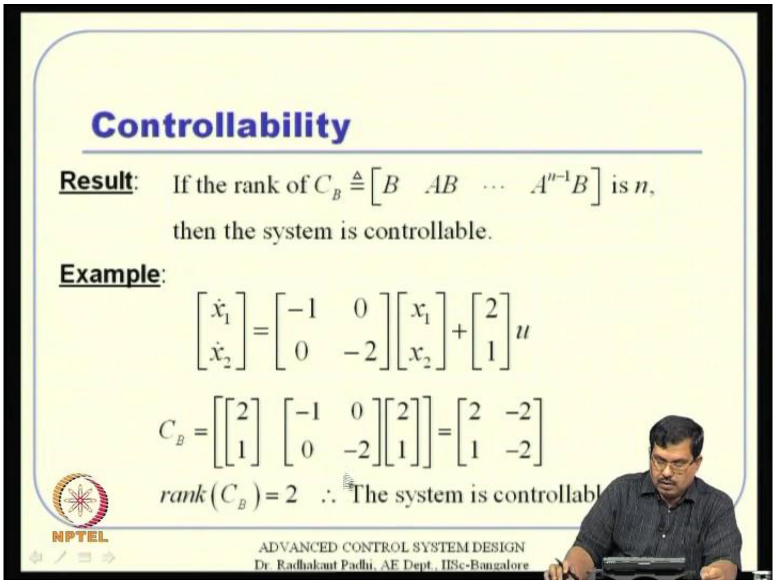
So, we need the system dynamics in this form. First is, if it is given in this form it is non-linear system. And, whereas the linear system or rather linearized systems are given something like this. So, this lecture we will continue primarily on linear system tools and techniques actually.

(Refer Slide Time: 01:35)



So, stability of linear system is first of all, when you talk about stability we do not talk about control input actually. We will only concentrate on X dot equal to AX actually. Now, the question is "can we conclude about the nature of the solution without solving the system model?" So, we do not want to solve the system model and then infer it actually. Fortunately, the answer turns out to be yes. And, it is all given by the location of the Eigen values actually. So, by definition, Eigen values of A matrix are known as the "poles" of the system. And, the nature of the solution is governed only by the location of the poles actually. So, if all poles are in the left hand side the system is stable, otherwise it is unstable. If one pole is in the right hand side the system is unstable. That is the conclusion actually.

(Refer Slide Time: 02:24)

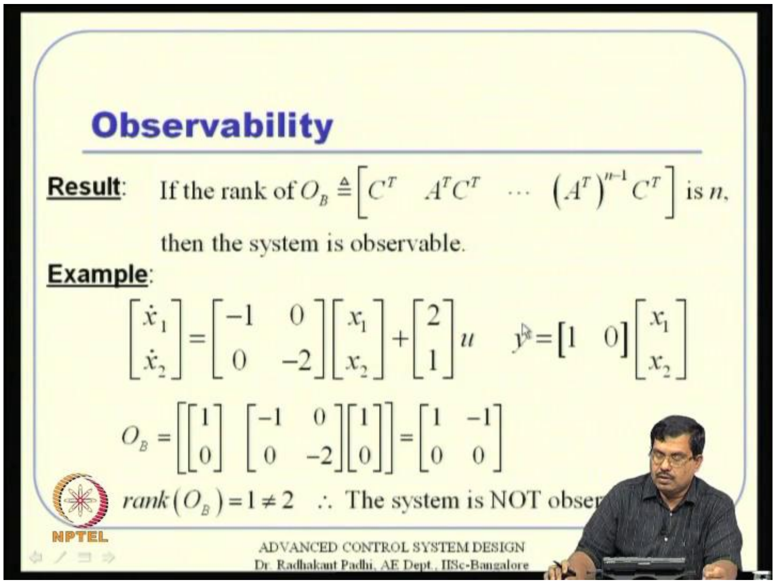


So, then coming to controllability; before we design any control system, we need to check controllability. And, for linear time invariant systems, the test is very straight forward. So, we just formulate this controllability matrix. And, if the rank of this matrix is n, then the system turns out to be controllable.

The whole idea is, if the system is not controllable, then no point in trying our, I mean kind of wasting our effort in trying to design a control system because that will not be possible actually. So, as an example, if you have this kind of a system X dot equal to A times X plus B times u, then is very straight forward.

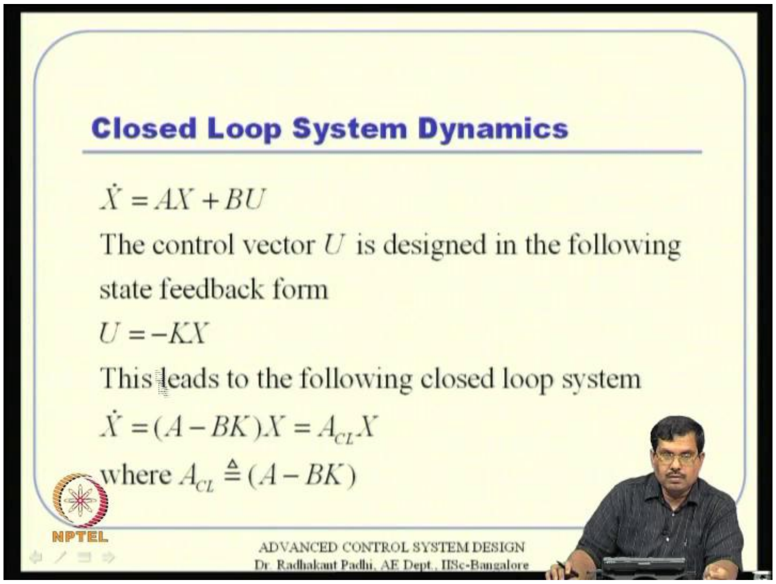
C B is something like this. This is B and this is A times B. Now, that turns out to be like that. So, if you see the determinant, determinant minus 2 plus 4, sort of a thing. So, it is not 0. So, that rank of C B turns out to be 2 and the system is controllable. That is how we check controllability.

(Refer Slide Time: 03:23)



And similarly, we will also talk about observability. Where, the observability matrix turns out to be like that. Again, you proceed with the similar test. The rank of the observability matrix is n, the system is observable. So, then as an example, here we require a C matrix as a property between C and A. So, we first take C transpose. C transpose is 1 0. And then, A transpose and then C transpose again. So, that turns out to be like that. The rank of this matrix is 1, then one row is completely zero. So, this system is certainly not observable. So, that is the way to, kind of go ahead and check controllability, observability.

(Refer Slide Time: 04:04)

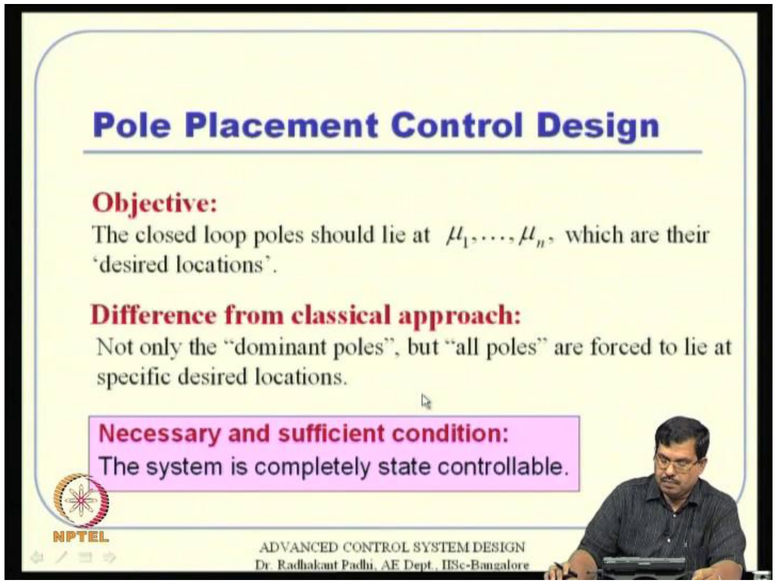


Then, there is an idea of closed loop system dynamics. So, these are all, when something is given like this, it is open loop system for say. And then, if you come up with a control design U in the form of state feedback, in other words U is equal to minus K X.

Then, this leads to the closed loop dynamics as is equal to B K times X. So, this A C L, which is A minus B K, it turns out to be like your closed loop system matrix. So, the whole idea of control design is if the system is unstable or something like that, we can certainly make it stable by selecting an appropriate gain matrix K.

And even, if the system is stable, if you really do not like the characteristic or the response of the system, thing like that, then you can alter the Eigen values by designing a control system in this manner actually. So, we will see this application in this class actually. How do you alter the Eigen value or closed loop locations actually.

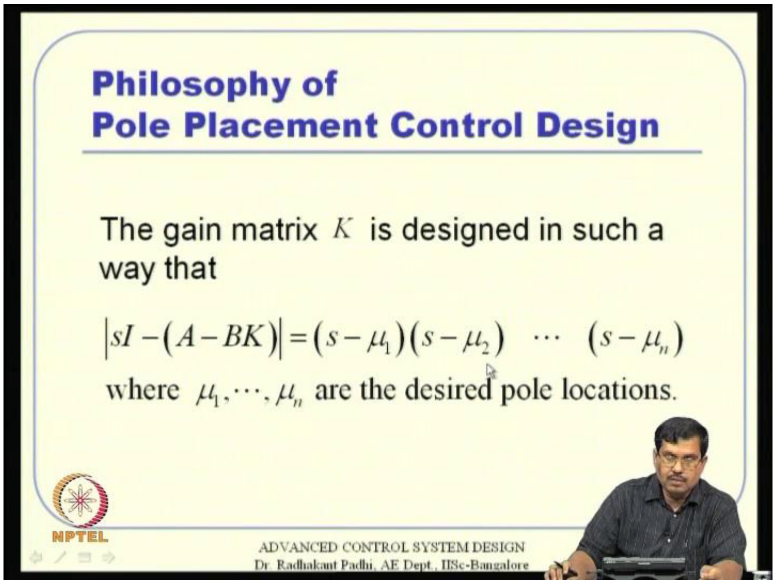
(Refer Slide Time: 05:07)



Then the pole placement control design we have studied. The objective was something like that, something like this. The closed loop holes should lie at specified locations mu 1 to mu n, which are the desired locations. Mu 1 to mu n is something that is desired. We want to place the poles there.

And then, the difference between this one and classical approach is not just the dominant pole, but "all poles" are forced to be placed actually. We should be able to, kind of no need of approximating it like a second order system and things like that. There is no concept or dominant pole. Every pole will be able to alter actually. And, necessary sufficient conditions turns out that the system needs to be state controllable.

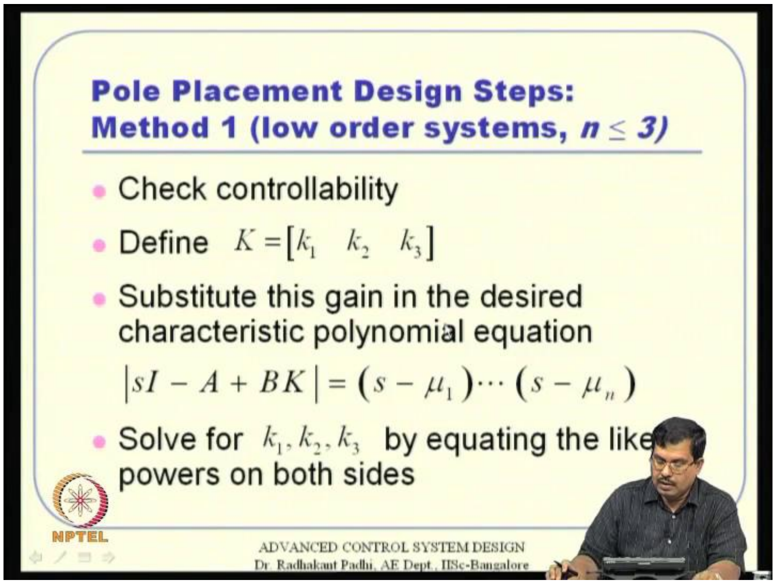
(Refer Slide Time: 05:54)



So, then this, what is the philosophy of pole placement control design that we have studied? Suppose these are the kind of desired pole locations, then certainly the right hand side part of it is the desired characteristic polynomial.

And, this is, this turns out to be the desired, I mean this closed loop system matrix A minus B K. So, left hand side is the characteristic polynomial after designing a K matrix gain, actually. So, this…two characteristic polynomial if you equate and then collect the various powers of s and then try to equate them and think like that and then, you will be able to solve for the gain matrix.

(Refer Slide Time: 06:37)



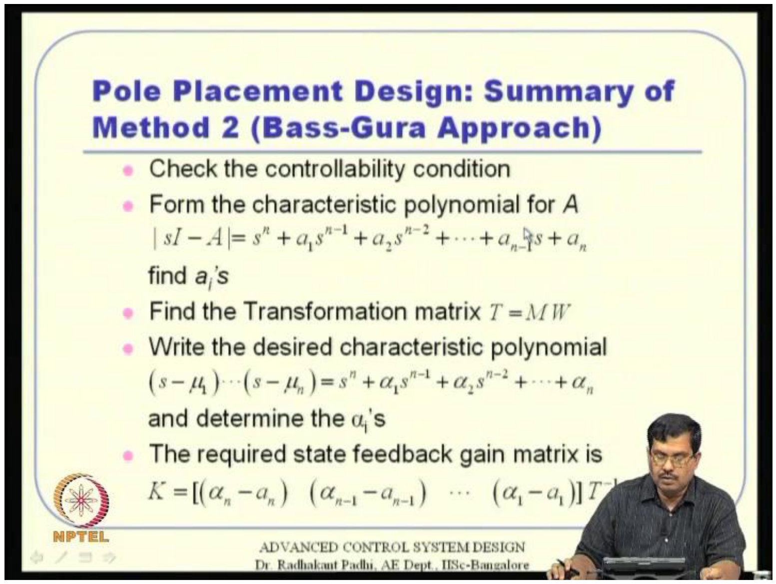
And, this turns out to be like not a very convenient method, but if n turns out to be less than equal to 3 sort of thing, may be to 4 maximum. Then, we can certainly go ahead and equate these polynomials… And then you can, we should be able to solve this k 1, k 2, k 3 directly, basically.

So for, as a method what what you do? I mean, first you check controllability. And then, suppose it is equal to three and then you also remember this rho vector, what you are talking here in is primarily a single input case. Multiple input, you have to do some control, I mean allocation and all that actually.

To convert it to an equivalent single input system, then you go ahead and apply this. And then, look at the controls appropriately. alright. So, if it is n equal to 3, then you select like that k 1, k 2, k 3, sort of thing. And then, substitute the gain matrix in the desired polynomial, so you will equate the two and then collect the various powers of s and then solve for k 1, k 2, k 3 actually.

And, once you get that, we have got the gain matrix. And, once you get the gain matrix, we have got the control vector actually. That is equal to minus KX. So, that is the way we proceed. And, this is not a very convenient approach, if for higher order systems, say if n is greater than 4 or things like that, things will be very messy.

(Refer Slide Time: 08:08)

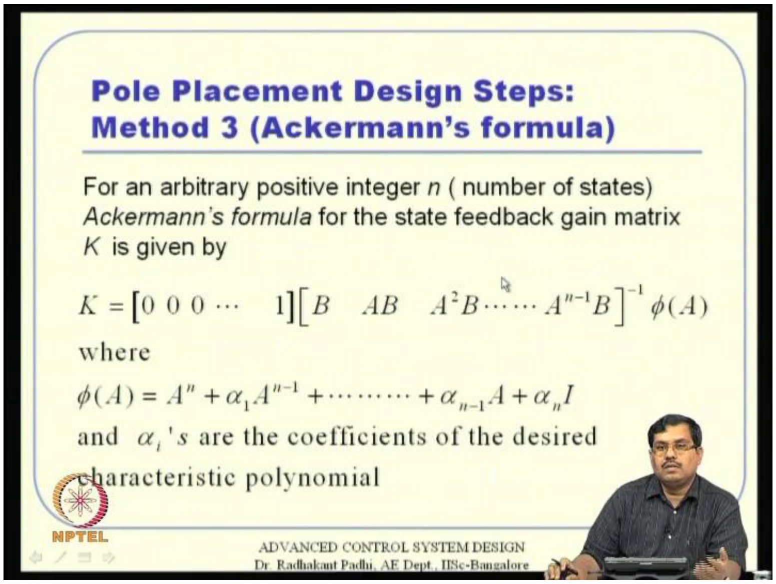


So, there are alternate approaches available. And, one of that is Bass-Gura approach. So... So, in this approach, what we studied before is something like this; like, first obviously when we see controllability anyway.

And, we need to form this characteristic polynomial for the open loop square matrix actually. So, like open loop square matrix turns out to be like that, then that will define various a's actually; that is a 1, a 2, a 3 up to a n actually, that way. Then, we have to find the transformation matrix T equals M times W; where M is nothing but the controllability matrix. And W, we discussed, is given in the specified format containing these co efficient a 1, a 2, up to a n actually.

So, that is the matrix which will utilize these coefficient a 1, a 2 up to a n actually, that way. And, M is already available to us. That is the controllability matrix. Now, we have the desired characteristics polynomial mu one to mu n of the desired pole locations. So, if you multiply this polynomial, you will get this alpha one up to alpha n. and hence, we can determine these alphas actually; that means alpha one, alpha two and all we can collect from there. Now, ultimately the required state feedback gain is turns out to be like that. So, where T is computed that way; T equals M times W. So, that is the approach that we studied before, method too.

(Refer Slide Time: 09:29)

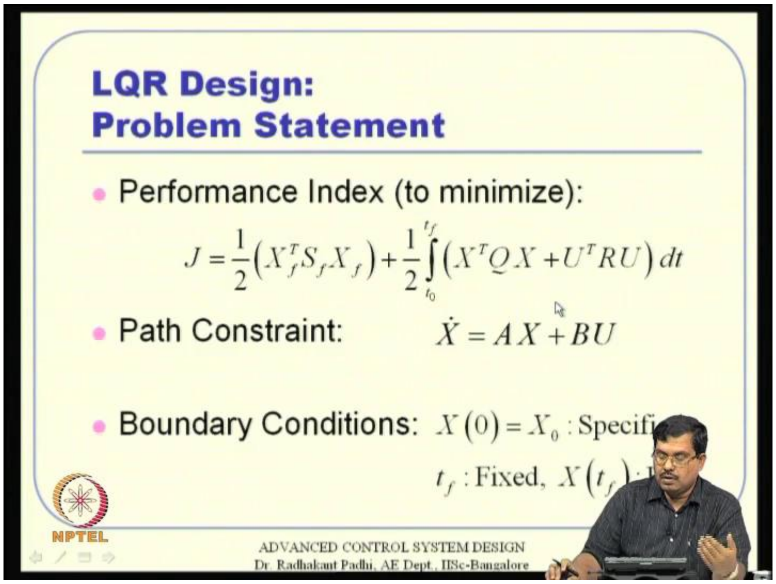


And then, there is an even more straight forward method, which we discussed at this Ackermann's formula. So, Ackermann's formula is, I mean is kind of a very straight forward. Then, what you see here is controllability matrix. So, you take inverse of that and then this clearly a rho vector defined…in all zeros and then the last element is 1.

And then, this phi A. and, this phi A is defined some like this and that counts from …Hamilton theorem like that. Remember, we have discussed all the details there in that class. So, this phi A is actually utilizes this polynomial again and this A is, sorry, not this polynomial, this alpha polynomial. So, this phi A's will come out from this polynomial; alpha one, alpha two and all are defined from this polynomial.

So, this phi A is defined in terms of that actually. This matrix polynomial is defined that way. So, utilizing this formula, we will be again able to compute the gain matrix directly as well. So, they are the three methods that we studied for control design. And equivalently, these three methods can also be used for …. I am not going to discuss that part over here. Our goal in this class is to design control systems finally.

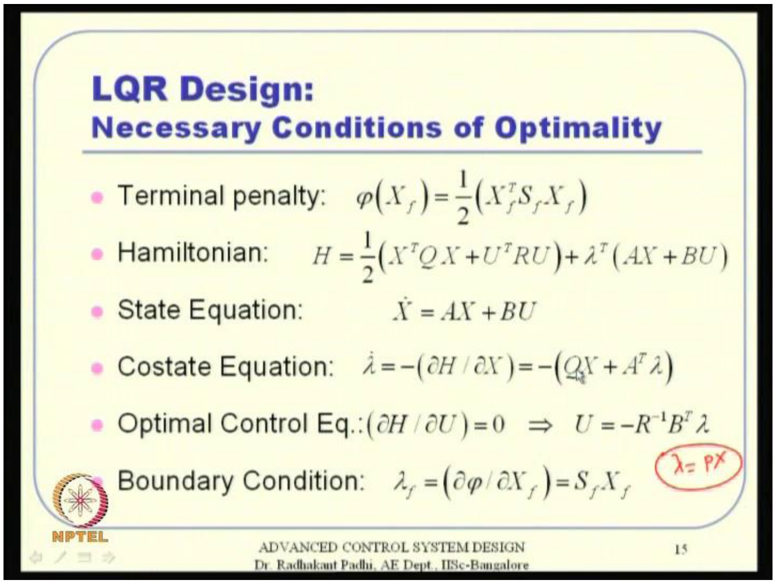
(Refer Slide Time: 10:48)



So, another alternative we studied before is the LQR design; linear quadratic regulated design. And, that is where we discussed this kind of a quadratic performance index containing a terminal penalty and a path penalty. And then, the path constraint is a linear system dynamics. The boundary conditions, initial condition was known to us. Final time was fixed, but final state was… I think.

So, then we went ahead. And now, I mean, followed the procedure of formulating an augmented cross function J bar, which is J plus, within the integral lambda transpose this X dot minus AX plus BU, think like that. Then, we define a Hamiltonian and carried out further…

(Refer Slide Time: 11:33)

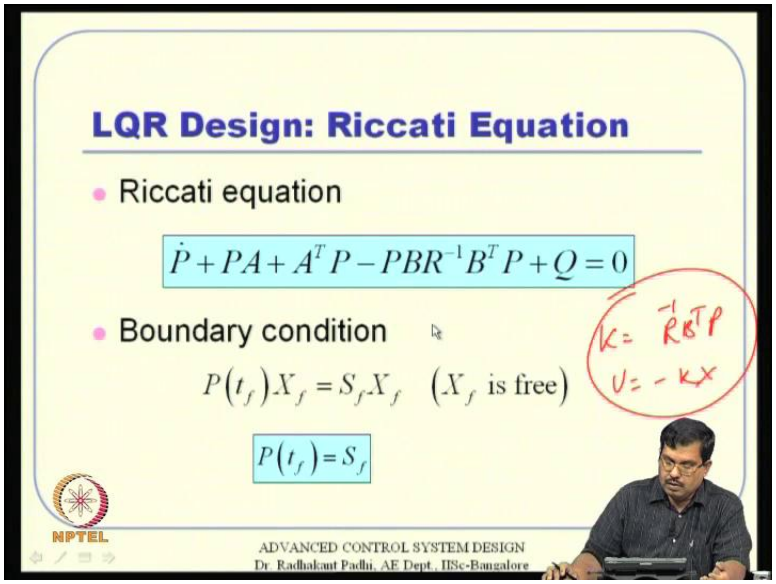


And so, this is what we did. Terminal penalty turns out to be like that. Hamiltonian turns out to be like that. And then, it leads to these three famous conditions; state, costate and optimal control equations. So, state equation was already known to us. this is the, … comes from here and the costate equation is lambda dot equal to minus del H by del X. so, if you carry out this, Hamiltonian is already available; H is already available. So, minus del H by del X if you carry out, it turns out to be minus Q X plus A transpose lambda. And, optimal control equation turns out to be del H by del U equal to 0 and that gives like U equal minus R inverse B transpose lambda.

The whole problem was to get lambda. So, if you get lambda, then your control is already ready. So, what we did is we approximated this. I mean the, not approximated the, that the theory also is there. It will tell you that lambda turns out to be a linear function of X. so, we took as lambda equal to this three times X, basically. If you see lambda equal to 3 times X and then lambda f turns out to be like del psi by del X f, sort of thing.

So, starting from lambda equal to P X, we carried out this derivative both sides that lambda dot equal to P dot X plus P times X dot. And then, X dot you substitute that and you substitute this one and in lambda substitute that, all these steps carried further and then lambda dot, also this expression and lambda is also like P times X. if you do all those substitutions and then try to figure out what is going on, it leads to this Riccati equation actually.

(Refer Slide Time: 13:04)

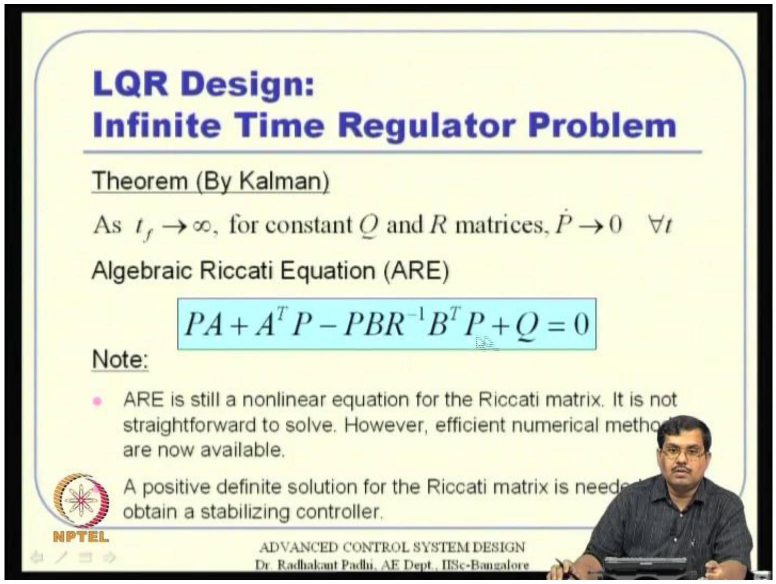


So, this Riccati equation if it is solved, then it is and we will be able to get that t. That t will be used for lambda. And once, you get for lambda and then we got this minus R inverse B transpose B times X. and hence, R inverse B transpose P will turn out to be the real matrix actually. So, but for solving lambda, I mean solving for P we also see the boundary condition. And, this boundary condition what you see here, if you utilize that you will get this kind of boundary condition.

So, in principle you can start from here and then using this differential equation, you can proceed backwards and then store the P matrix. And, at appropriate point of time, you cancel that particular matrix. And then, you can compute your gain matrix, which is nothing but this R inverse gain matrix is nothing but R inverse B transpose P and then U will turn out to be minus K X.

That is how we, kind of compute the gain matrix and the controller. However, this differential equation propagation and all is not very comfortable. And then, we do not know, suppose we do not know t f theory, then there is a problem for that.

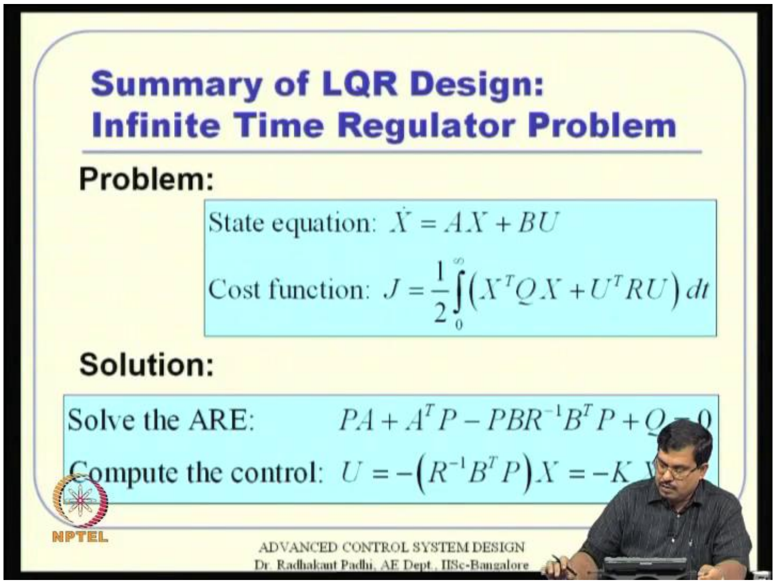
(Refer Slide Time: 14:23)



But, then there is a great theorem which tells us that for linear time invariants systems, if t f goes to infinity and then Q and R are also constant matrices, then P dot approaches to 0 for all times actually. So, that is the theorem by Kalman. So, this degenerates the algebraic, I mean differential Riccati equation to Algebraic Riccati Equation. And hence, if we solve this, one time we are done actually. So, if you solve it, remember that this is also a nonlinear equation because P appears both left and right here, so it will invariably lead to a number of nonlinear algebraic equation that we need to solve. And, what we really need to solve is the positive definite… That is what ultimately gives us stabilizing controller actually. So, you can eliminate all other possibilities and select a P using positive definite solution actually.

So, that is what is written here. Positive definite solution of Riccati matrix is needed to obtain a stabilizing controller. If all the conditions are met like a P is controllable and Q is positive definite and then R is positive definite, thing like that. Then, certainly it is possible to get positive definite solution from this matrix, this Algebraic Rickety equation.

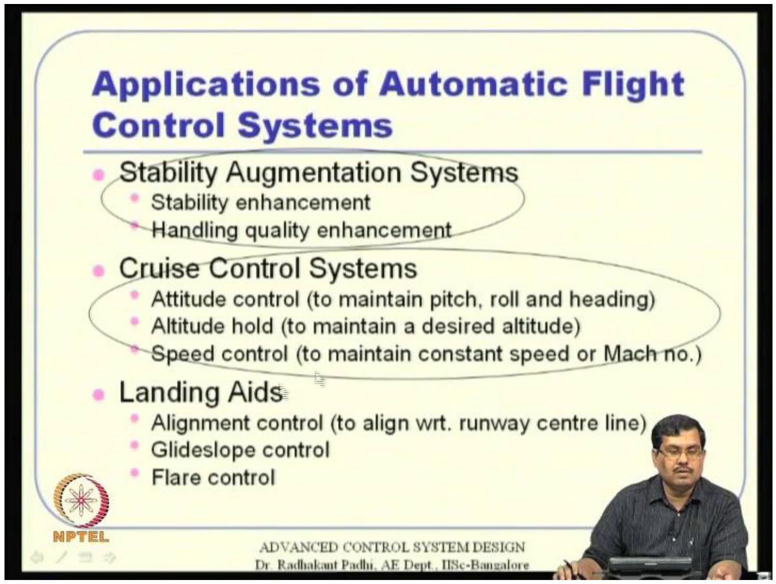
(Refer Slide Time: 15:28)



So, once you, summary of this infinite time regulator problem is like that. So, we have a state equation; X dot is equal to A X plus B U and we…have… matrices. And then, cost function is something like this. So Q and R are also known as …matrices. Q is positive definite and R is positive definite. So, solution turns out to be like that, very straight forward… solve this Algebraic Riccati equation, get a solution for P matrix and then you have this U equal to minus R inverse B transpose P, where R inverse B transpose P turns out to be the gain matrix K actually. The beauty of this LQR over …design is, you really do not need to bother about it. You bother either by single controller or multiple controllers. So, in other words, this is equally valid for multiple input without doing any further manipulation actually. …theory is very nice only if you have a single input system.

The moment it is multiple input, we need to do this control allocation and think like that. And then, we may lose some of the beauty there actually, that way. Alright. So, these are all the conditions and techniques that we have studied before. Now, let us go and see what way we can utilize this thing for flight control systems actually. And, especially we will see for time domain designs.

(Refer Slide Time: 16:55)



So, last class we also saw that application of automatic flight control systems can be many. I mean applications of control theory for automatic flight control system; that is what we are talking here. And then, largely you can categorize them as three to four categories.

And, one of that is stability augmentation system. So, what we do is stability enhancement here. And, as well as something very important thing what we do is handling quality enhancement. Stability enhancement is something like, if the system is kind of unstable or close to instability and you claim that you certainly want to make it… stable…with the help of feedback actually. Handling quality is something like, what the pilot perceives to be and we should be able to read that actually, that way.

And… if the response is too fast, then the pilot may also get confused actually. So, it may not, the too fast response are not very nice, at the same time this… minimum response, think like that all very confusing to the pilot actually. We also remember that when you have a telecontrol flight vehicle and most of all aerospace vehicles are telecontrolled anyway. It excites this closed loop response in a way that it certainly leads to non minimum…actually.

So, that response needs to be minimized as soon as possible. Then, handling quality also discusses about like what kind of…you should, how much percentage you should and thing like that. So, in a nut shell what it tells is like, the before we, I mean like if the pilot gives some input or something, the system or the aircraft should respond the way the pilot is comfortable actually.

So, that way you can handle the aircraft. that is called handling quality. And mathematically speaking, this Eigen values of the closed loop matrix should satisfy the desired locations and all. It will lead to this desired response characteristics like percentage over… and things like that. So, stability enhancement is just to enhance the stability; I mean they are nothing to do with exact four locations and think like that. Even though, you can even talk about that there itself. For handling quality, it was little more than that. It tells, ok, exactly the, I want some kind of response from the system.

And then, cruise control systems are also very important application of control theory. This is actually takes out large amount of load from the pilot actually. For example, if it goes for a long duration flight and thing like that, the pilot cannot concentrate on controlling the vehicle at each instant of time.

So, for small jobs, it says good to automate that process. And, that actually elevates the load from the pilot required …actually because if you see commercial aircrafts, for example, for large part of their trajectory, may be more than ninety percent of the time, they just go on a cruise control mode actually.

So, unless there is a warning sort of thing, pilot does not play much of a role actually. Like warning in the sense of let us say something drastic happens, some big disturbance comes or the collision… something like that with another aircraft. Unless, those type of warning situation comes, it is actually the aircraft goes in a cruise control mode. And then, this cruise control has also come; as I told you in the last class, to the automobile industry already basically. Anyway, the cruise control comes in variety forms.

And, first of all we can discuss about attitude control; that means to maintain pitch, roll and heading I mean pitch, roll and heading angles basically. So, you can maintain certain angle and then rest of the things will be taken care. For example, if your aircraft takes off, then we need to maintain certain angle, pitch angle theta with respect to the horizontal, then the aircraft will… So, that is, that kind of an attitude control.

Then, we also have this altitude hold. That is what happens, in long duration of flight, cruise control sort of thing. We just climb to a certain altitude like ten kilometer and twelve kilometers like that and then hold that altitude, hold the heading also.

Particularly, if you know the direction in which you are going, and you know the altitude where you want to go and think like that, then combing these two attitude control and altitude hold. And, this will be able to come up with a nice cruise control and take the, automatic control system takes over the pilot control actually. So, then, that is what the cruise control system is all about.

Speed control is also important because they… various systems…when we discuss, then automatically it should maintain certain speed actually. So, that is where this has come to automobile industry. Whether you climb up wheel or down wheel or things like that, so your vehicle should go at constant velocity and constant speed actually. So, that is where you manipulate your brakes or manipulate gas pedal, accelerator or something, so you can maintain your constant speed actually. So, these are the application of cruise control. And then, landing aids also is, I mean it is possible. And, especially when you discuss this flight trajectory, it consists of three parts. First is like a climb up, then go on a cruise mode for a long time and then land actually.

And, after these three main segments, the landing turns out to be the most crucial part of it, where most of the accidents do happen there. So, that is where lot of care has to be taken because your vehicle is subjected to lot of, I mean this horse forces moments and all when it tries to land there. And, any amount of errors and all can be very penalizing there actually.

So, even if it touches, if your wing touches the ground as little bit because your velocity is so high the momentum turns out to be or your energy content turns out to be so high so that, the wing will break… actually. So, there this very…kind of trajectory where the automatic control turns out to be most useful in the loop. Even though, you cannot completely automate; in the sense, there are lot of work going on in that direction how do you do the complete automatic landing, especially in the UAV side for example. No pilot is there in the loop anyway, basically. And, even if there is a pilot, it can only control the vehicle remotely; remote control vehicle and all that. But, truly speaking the UAVs should land automatically actually. And, when you try to do that, there are variety of issues there and lot of challenges are there actually. But in a regular sense, what is used already actually what is called alignment control; that your vehicle has to align on the runway first.

Then, there is a large segment, in which it will simply glide actually; that means it will simply try to come down in a straight line and sort of thing. But in a very, when the aircraft is very close to the ground you consider the straight line path and go for the exponential path actually so that, the touching will be very smooth sort of thing. And, that is what normally birds do.

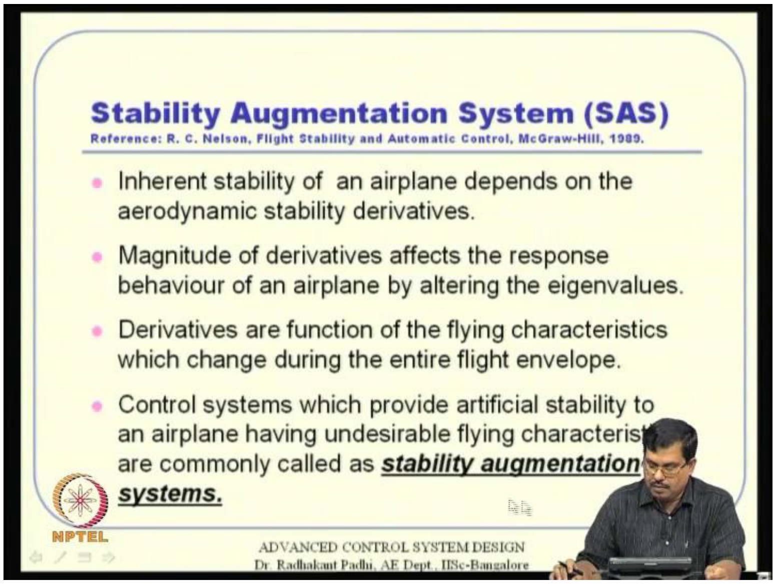
So, this glide slope control comes out from observing simply, I mean simply observing birds actually. It will come down in a straight line and then take a little nice…upwards and then the rest of the path it moves exponentially smooth sort of thing. And, there are other applications very important also. That is something called automatic path learning and guidance actually. That is where optimal control theory is coming in to a very, I mean it plays a very important role actually.

So, if you really talk about automatic path learning; that means the vehicle should plan its path automatically. Let say… And, in between also there are obstacles there… And, if it is a, in a collision mission sort of thing like a war mission sort of thing, so not only find the target, go and collide with the target that way. These are all many important class of problems for which this automatic flight control system turns out to be very useful actually.

So, these are lastly the things that we have discussed here; I mean the rest…guidance and path learning. What we concentrate here is not everything in detail. But, something like, some typical ideas of stability augmentation system followed by some examples actually. And similarly, like small idea of like cruise control system, how do you do it? How do you mechanize it actually?

So, once you have some ideas there, other things are the extension of those ideas any way. So, that is where I will take you through this class actually. So, let us talk about stability augmentation system first.

(Refer Slide Time: 25:41)



So, when you do this stability augmentation system, first of all note that inherent stability of an airplane depends on the aerodynamic stability derivatives. So, that is the stability characteristics actually. For example, it all depends on like wings shape, it depends on the length of the thing, it depends upon the body shape, it depends on the, I mean kind of what angle of it…you are flying and depends on the incident angle and many things it will be… part of the design like design variables and all that actually. So, in the design stage itself it needs to be optimized actually. So, and many times you remember that we may not need a stable aircraft all the time. That is also true actually. For commercial aircrafts, we need stable aircraft. But, for fighter airplanes and all we purposefully need an unstable aircraft for even track, I mean track minimization number one and then we discuss about like higher maneuverability and think like that. So, it reduces the… So, that means when it turns at high speed…that way. So, that is where you need this. And, the response time also becomes smaller actually. That is what it is crucial in…

So, we do not necessarily need a stable aircraft. But, once you have unstable aircraft, certainly pilot cannot handle it manually. So, we need to have kind of closed loop system, partly at least… this inherent stability of the airplane depends on the aerodynamic stability derivative. And also, lastly remember it also depends upon the CC to CP location basically. If the CC is in front of C P; that means center of gravity is in front of center of pressure, then it turns out to be a stable aircraft, longitudinally stable aircraft basically.

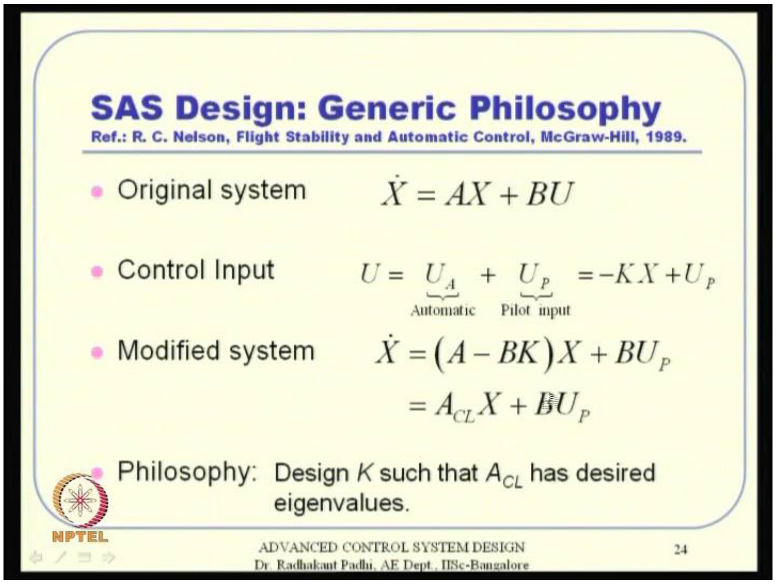
Magnitude of the derivatives affect the response behavior of an airplane by altering the Eigen values; certainly, obviously actually. The derivative means the stability derivatives for c m alpha c n delta like that. Those are called the stability derivatives. So, aerodynamic stability derivatives affect the response behavior of an airplane by altering the Eigen values. Suppose you have a different value, somehow you alter the values, then obviously it will reflect in the Eigen values….

So, derivatives are the function of the flying characteristics which change during the entire flight envelope as well because these are not constant numbers, they keep on changing as functions of angle of attack, functions of Mach number and functions of, I mean… think like that. But, largely there are functions of … and functions of angle of attack actually; angle of attack, Mach number and to some extent, this dynamic pressure. So, dynamic pressure means it is a function of … anyway.

And, also remember the control systems which provide artificial stability to an airplane having undesirable flying characteristics are commonly called as "stability augmentation system". So, what does it mean? So, the system in, I mean by itself is not stable, but by designing a control system you are making that closed loop plant stable actually. So, that is like artificial stability actually.

And, the moment the control system stops working, the only option for in this class of problems is to eject and go. I mean the pilot can never aim to stabilize the plant manually. So, that is, the aircraft may crash if it ejects and goes and that the pilot's life can be saved. That is the reason why these fighter aircrafts will have ejection seats and all that. For whatever reason if the control system does not work, there is no hope of making the system stable again. So, it is just…

(Refer Slide Time: 29:42)



Alright. What is the philosophy? The idea is extremely simple, which is a clever application of control… I will put it that way. So, let us see this. The original system is something like this; X dot equal to AX plus BU. And, the control input U, what goes there to the plant will kind of visualize; that is two components actually.

One is the automatic feedback component and the other one is pilot input component. And, this automatic feedback component, let us try to design it in this way; minus K times X. And, U P can be designed based on the overall plant. That is the different issue actually again. But, forgetting UP, we will just keep the U P as it is. Then, what happens this X dot, if you substitute that U A plus U P and that U A is nothing but minus K X. Then, what happens? X dot equal to A minus B K times X plus B times U P. so, that means instead of visualizing this plant that X dot equal to A X, what I am visualizing X dot equal to A CL X where CL is A minus BK plus B times U P. And then, this U P is pilot input. So, then if you want to make automatic everything, then this U P also needs to be designed in automatic sense. But, in general, if the pilot gives command through sticks and buttons and things like that, then this is were it will come and affect the system which is already in the form of A CL. Where X dot is equal to A CL plus B times U P actually.

So, part of the control is already designed, automatically designed, built in actually. And, that is what will give stability to the system and rest of the things will come from U P. and then, U P will be given directly by the pilot actually. So, what is the… and that is what the stability augmentation system design is all about actually.

So, this A does not have good stability behavior. But by partly automatic, by partly making it and operate in an automatic way, we are actually giving the desired stability behavior to the system actually. And, this desired stability behavior can also accommodate this, what is that, handling qualities and all that actually. That is how we design it actually.

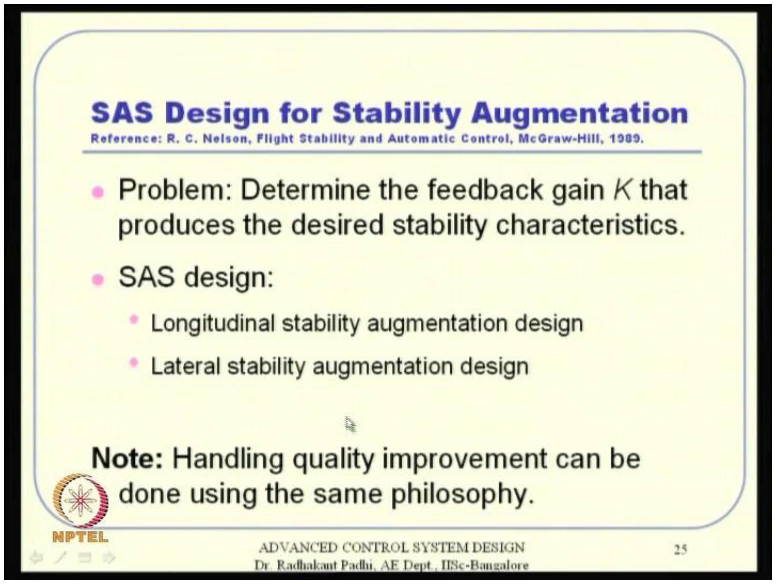
So, what is the philosophy? Philosophy is to design K, this gain K such that, A CL has desired Eigen values actually. So, once you realize this, you can all bring in our earlier idea of either pole placement or LQR or whatever it may be...to design this closed loop part. And, for example, when the, I mean there is another utility of this kind of approach. For example, when your aircraft is getting designed, in other words you have not gone for extensive flight testing and all that, we do not know how this aircraft is going to behave in air actually.

So, what people normally do is, you design a artificial gain K in such a way that A CL will have characteristics of a non-aircraft already; that, which is already flown on so many times actually. So, that way… And later, this will not be required once the aircraft is, I mean once you study the aircraft in a good way how it happens, what it happens and think like that, you can, ok, you do not need that.

So, what you do? Initially, you design a K matrix in such a way that, it will …once you …once you have activate this, the same aircraft… actually. So, the pilot can think though I am flying that with the other one that I am comfortable with each other. Ok. Response characteristics will become like that. Then, once it is intermittently, you can take out this loop and study the natural behavior how it happens with their own control design, which is already influenced. And, if you are comfortable, well, get yourself trained; if you are not comfortable… so that, you can get that aircraft big like that another aircraft actually. So, that goes as a very handy tool for training pilots in a new aircraft actually.

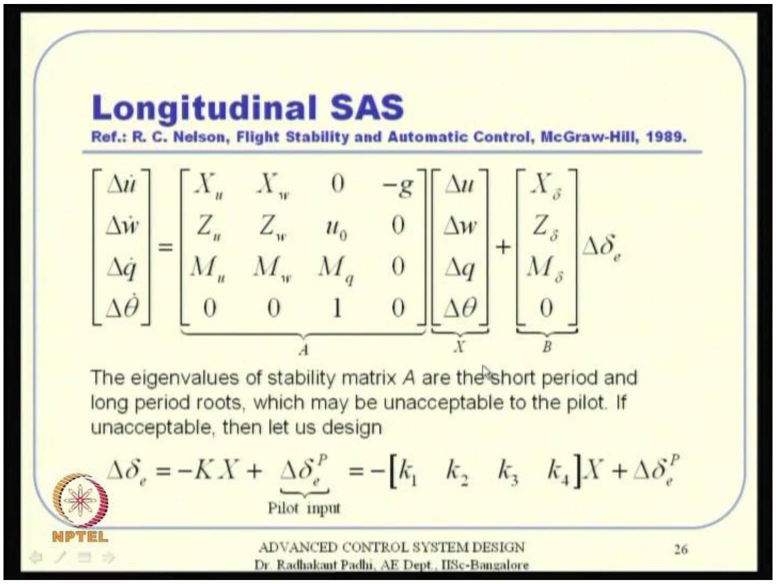
So, that is and there are many examples… very useful technique basically. Alright. So, that is what we do there actually. Alright. So, the philosophy is like this. You design part of the control in automatic sense, so that you can alter the response characteristics….actually. Ok.

(Refer Slide Time: 34:12)



So, problem is what? Now is the, with the problem we will turn to determine the feedback gain K that produces the desired stability characteristics actually. And, stability augmentation design can obviously have longitudinal…I mean the longitudinal stability augmentation design and lateral stability augmentation design. And, as I told before, this handling quality improvement can also be done using the same philosophy. Handling quality is nothing but how do you feel the response characteristics that is nothing but the Eigen values of the A CL actually.

(Refer Slide Time: 34:47)



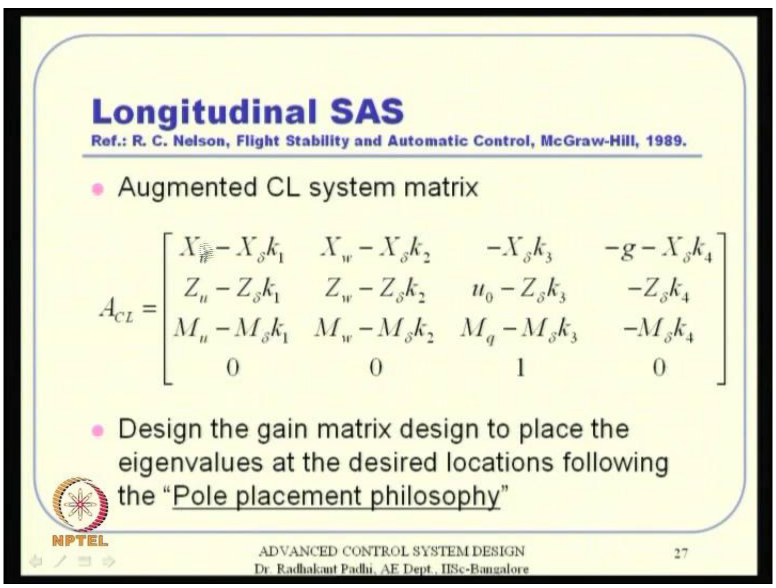
Now, we have studied before that the flight dynamics in longitudinal mode can be written like that. So, out of these nine variables…model, we can separate that into four plus four mainly, actually. And then, you can discuss about this u dot... in a linearized manner. The… play so much of important role there. So, out of these nine equations, you can reduce it to eight and that eight you can divide into like, four plus four. And, this four plus four; these four variables u, w, q and theta is something that comes into the bracket of longitudinal dynamics actually.

So, this longitudinal dynamics is largely dictated by thrust to manipulation as well as elevator manipulation. And, in a linearized manner we consider, we assume that the thrust is not manipulated, thrust remains…because that cannot be manipulated in a fast way. Thrust can only be manipulated very slowly. So, we no need of, kind of taking into account when we are discussing about a linearized model actually. So, this is the, in general this is the linearized model in longitudinal mode actually.

So, what you are doing here, this delta delta e, I mean this delta notations I have kept it purposefully. And, this book also…Ok, lastly this material is also taken from this book like last lecture, say very good book and lot of details are there in there actually. Anyway, so like the book, I also thought that I will keep the delta notation because this delta, when you discuss about the linearized model, these are all… things actually. Let us keep it in mind purposefully actually that way. So, this delta delta e is, what are you doing here is, partly automated minus K times X, ok, plus this delta delta e P, which is the pilot input actually.

So, remember this is a four dimensional system. So, we have this gain matrix and the single control input. So, that is where the pole placement can be very handy actually, that way. So, we have this; gain matrix K we will write it that way k 1 k 2 k 3 k 4. And then, this pilot input will not bother actually so much. Pilot input is pilot…whatever he wants to be actually. Ok. Once again if you want to automate the complete trajectory, in other words path planning and all sort of things if you talk about there, then this one also needs to be done in the automatic way. This delta delta e P is like pilot input basically. We will not bother about that in this class. Anyway, so this k 1 k 2 k 3 k 4 is something that we need to find out actually.

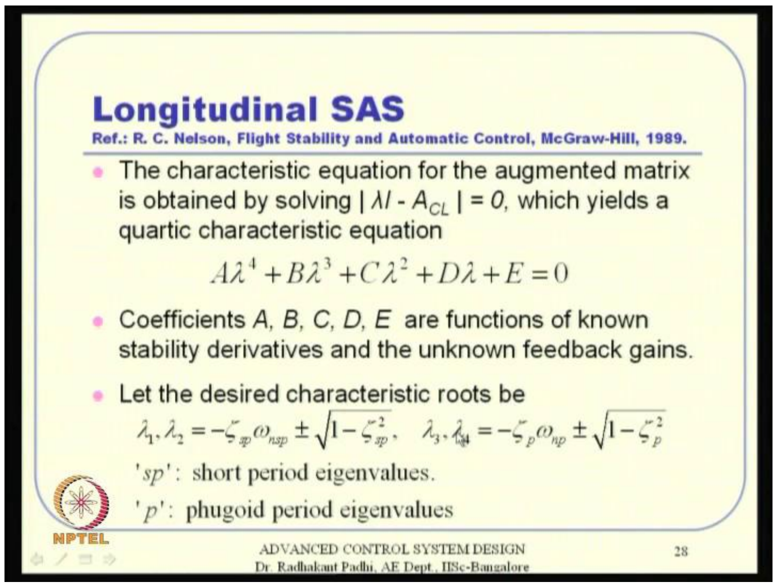
(Refer Slide Time: 37:14)



So, when you do this A minus B K, then A minus B K turns out to be like this. So, you have this B times K; B is here, B times K is a matrix say 4 by 1, this is one by four. So, it will turn out to be four by four.

This is already 4 by 4. So, A minus B k if you do it, it turns out to be something like that actually. So, we will be able to design now the gain matrix to place the Eigen values at the desired location following the "pole placement philosophy". That is now very standard actually.

(Refer Slide Time: 37:51)

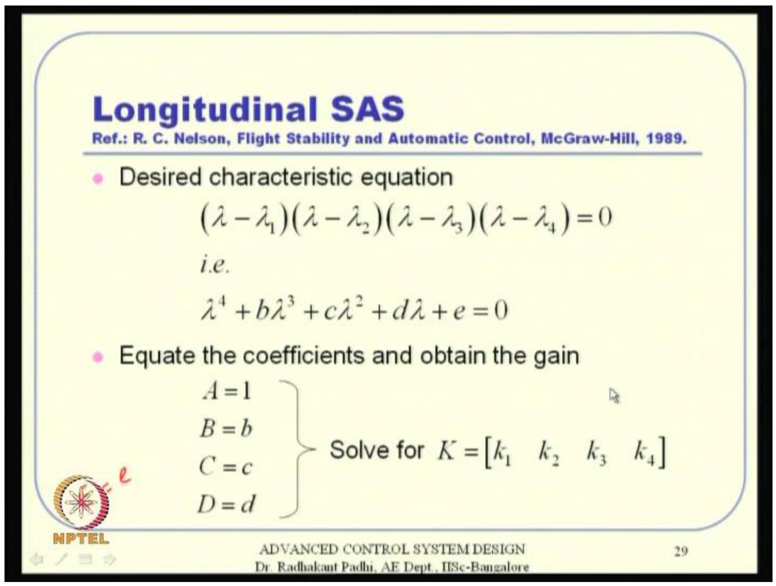


So, what do you do? Your A CL is known to you, so you compute the characteristic polynomial so that, lambda e minus A CL determinant equals 0. So, that means this determinant will throw ultimately a fourth order polynomial in terms of lambda.

So, which will have coefficients A, B, C, D, E; fourth order polynomial will have five coefficients. And, these are functions of this stability derivatives. I mean this A, B, C, D, E will come from X u delta X delta z u delta… The numerical values of that will dictate what values you will get there actually. Alright. So, now let the desired characteristic roots be at these locations; lambda 1, lambda 2 and lambda 3, lambda 4 actually. So, lambda 1, lambda 2 turns out to be like this; these are kind of specified actually. And, lambda 3, lambda 4 is of course like that actually. Now, why you write this 'sp'? Ok, that typically, now what we have discussed before, this lambda 1 lambda 2 can correspond to something like short period Eigen values; desired Eigen values for short period dynamics. And, this p is I mean desired Eigen values for plugoid dynamics.

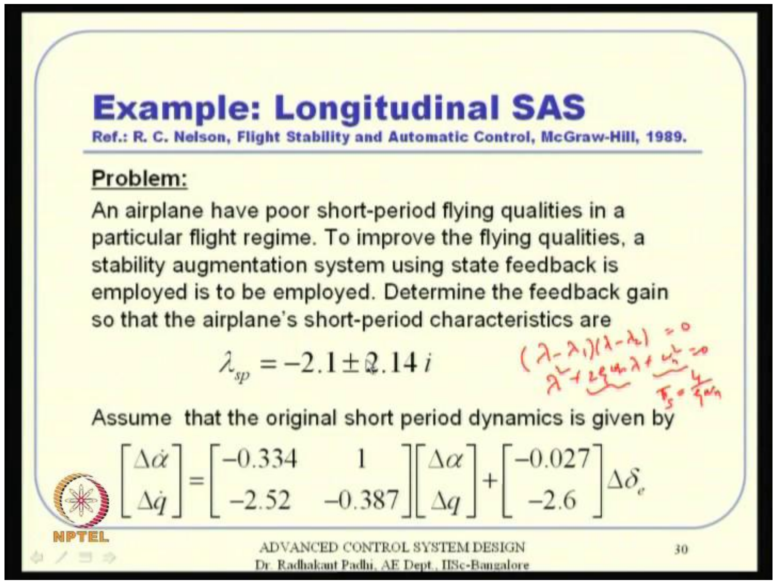
So, it is, very cleanly you can divide this Eigen values spectrum to kind of branches and then talk about okay this is my response for plugoid dynamics and this is my response for short period dynamics. But nonetheless, this lambda 1, lambda 2, lambda 3, lambda 4, all are available. So, we can formulate this characteristic polynomial now.

(Refer Slide Time: 39:15)



So, this characteristic polynomial will throw a polynomial like that again. So, now we equate the coefficient; that means, if I compare this polynomial with that polynomial; this characteristic equation with that one or other two polynomials actually. So, if I equate the coefficient of the two polynomial, I will get A equal to 1 because this is A and that is A equal to 1 here. And, B equal to small b and C equal to small c like that actually. This is like…. This E equal to small e also; there is one more equation actually. Ok. E equals small e basically. Ok. The equation turns out to be small e actually. So, this equation; if you see there are number of equations one, two, three, four and then five equations and you have… actually. So, you can solve for that… And, what are the… Anyway, so that is the equations and all, once you put it together, then it will throw you some polynomials and all. We will be able to solve this coefficient k 1 to k 4 actually, that way. So, one equation that you may not… think like that.

(Refer Slide Time: 40:54)



Anyway, so if you, I mean this is the generic procedure. But, let us talk about a small numerical; I mean example also to make our radius more clear. So, this way, let us talk about only short period dynamics. So, we really do not have to bother about… also in the loop actually.

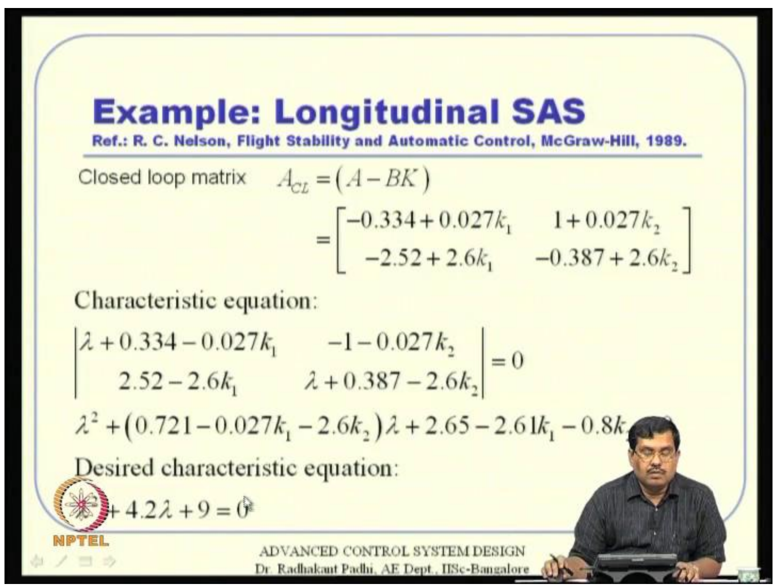
So, short period dynamics is largely like two variables; so, alpha and q. Instead of going to w q, I can also replace w by alpha. That is also another possibility actually. So, alpha, delta alpha dot and delta alpha q dot, when I have this is a function of alpha and q. That is how the short period dynamics behave. And, this is given by that, where the numbers are already available to oscillate. The stability derivatives are all given with respect to some specific aircraft and with respect to some desired trim condition basically. Some desired position; I mean this reference values and all that.

So, if these numbers are available and this is our A matrix and this is our B matrix actually. What we really need is that the airplane short period characteristic Eigen values would have this Eigen values actually. Now, where does it come from? I mean in a classical location sense and all that, so if you really, let say this is lambda 1 and lambda 2, then if you do lambda minus lambda 1 into lambda minus lambda 2 equal to 0, then it will give you lambda square plus 2 zeta omega n lambda plus omega n square equal to 0 that way.

Lambda 1, lambda 2 are known to us actually. Now this zeta and omega n, we know that they can come from time domain specifications and all that. So, if the pilot wants some sort of a settling time t s, which is… that is the good for recently… or in this particular case the … within certain specified settling time, then obviously this is four by zeta omega n s. This is one restriction. The other one will come from over settling time actually.

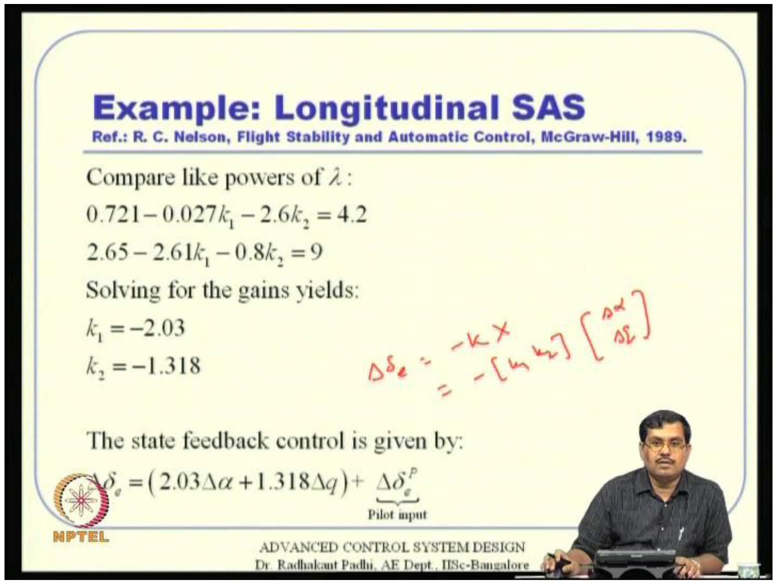
So, if you start from these specifications, you will be able to compute lambda 1 and lambda 2. And, once this lambda 1 and lambda 2 are known, these are nothing but that actually. So, incase somebody has…that it is given as this is what that actually. But, if you compute the Eigen values of this A matrix only, then it will not follow there. Then, how do you that?

(Refer Slide Time: 43:17)



So, when we go ahead and do this closed loop matrix, so listen this. So, we have a beautiful technique of control design with us. We will do that in all of a second and we will operate that in the form of U equal to minus K X. The closed loop matrix will turn out to be A minus BK. So, it will be like this actually. So, characteristic polynomial turns out to be like that. And, when you do this desired characteristic equation from here that will turn out to be like that. So, obviously this one does not give us anymore kind of information one equal to 1.

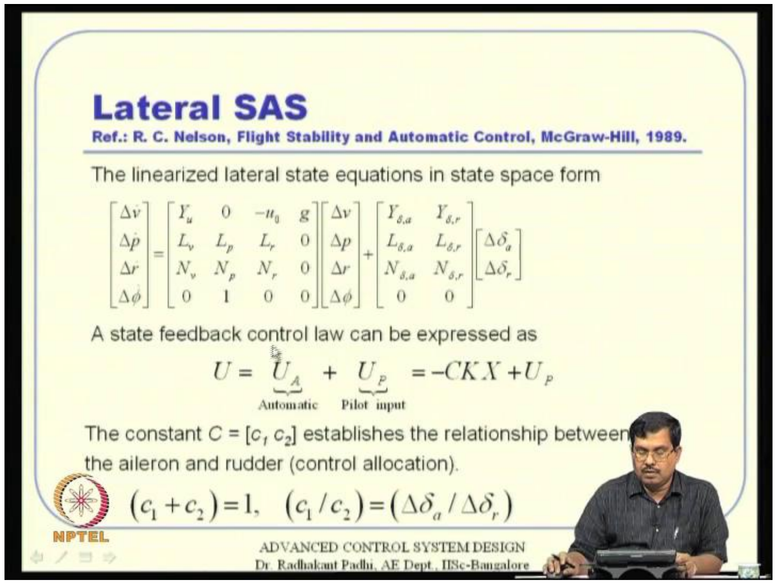
(Refer Slide Time: 43:59)



But, this one gives equals 4.2. And, that is nothing but 9. So, if we equate the two and solving these two linear equations, you will get the gains like that actually. So, the state feedback control law ultimately turns out to be delta e or rather delta e equals minus K times X. So, this is like delta delta e equal to minus K times X and this is minus k 1 k 2 times X. X is nothing but delta alpha and delta q... And, k 1 k 2 are available here. So, if I substitute that, then k 1, ok minus k 1, minus k 1 is 2.03. That is where it comes. 2.03 times delta alpha and this minus k 2 is 1.318.So, that is where delta q…

So, if the pilot gives the command, he will not feel the Eigen values of the original system. He will feel the Eigen values of the desired system. That means he will feel responses as if it is behaving that way. Ok. That is good for … in handling the aircraft actually…Alright. And, many of the luxury cars are also driven that way. So, for example, trucks and control system, AVS system and things like that. There are many automatic control systems that on the high end cars actually. So, you do not feel like you are…that way.

(Refer Slide Time: 45:36)



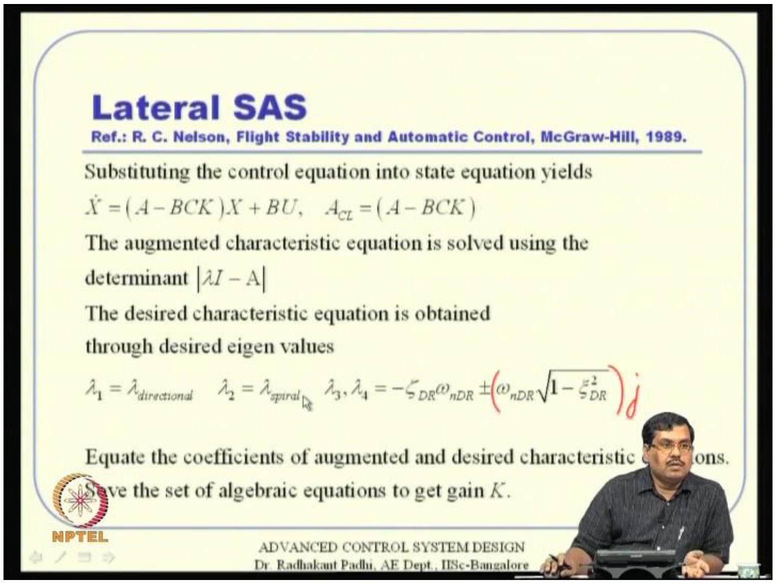
So, what is about lateral stability augmentation system? It is something like this. Very similar, very parallel, actually. So, this is v, p, r and phi. If we put them together, but the problem here is this is like both of the control that is… are equally effective; that means they are equally fast actually. So, I cannot neglect the one over the other for sake.

So, when I have to take into account both of the things together actually. That is the only difference probably here. And, once you have multiple control input, we also know that pole placement design technique. So, we have to do something else to meet the design work. And, there is something else, nothing but control allocation actually. And, one simple allocation is something like this. Anyway, the control philosophy is the same; U equal to U A, the automatic part and then pilot input part. But, this automatic part is minus K times X… But, let us do minus C times K times X; where C is the row vector; c 1 c 2, which will combine these two effects actually that way.

And, the constraints on c 1 c 2 is something like this; c 1 plus c 2 should be one and c 1 over c 2 is this one; which will actually dictated by their maximum limits actually. Like, how much they can be deflected and think like that. So, if you assume that the effectiveness does not change from the, with respect to the magnitude of the deflection; that means the control and influence numbers, whatever numbers you see here they remain constant, then dividing and presuming a specific ratio is not ... actually. Ok . Because you know this limit, you know that limit.

So, you kind of allocate that based on maximum deflection levels actually. However, this c 1 plus c 2 should also remain one. And, with respect to that you can select coefficients c 1, c 2 so that, you can define in a row vector C basically. And, once you define row vector C that is your kind of control variable, control… that operates somewhere actually.

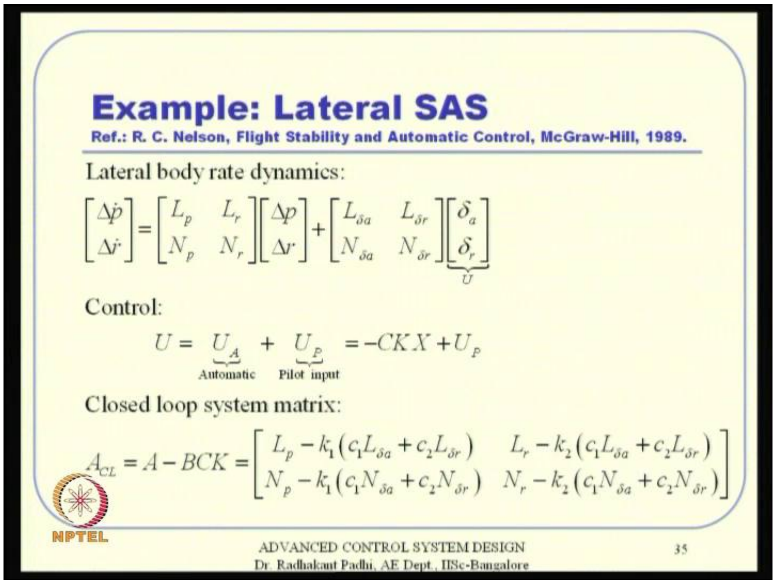
(Refer Slide Time: 47:32)



So, what is your closed loop matrix? This time it is A minus BCK. And, hence A C L, the closed loop matrix turns out to be A minus BCK. So, the augmented characteristic equation is solved using like lambda I minus A. And, in this case, let us say the desired characteristic equation to the desired Eigen values again. And, the desired Eigen values, remember you are talking about Lateral dynamics. And, if you remember Lateral dynamics which we also reviewed last class, it consists of this directional diversions, spiral diversions as well as…actually. So, when you see this lambda 1, still we have lambda 1 coming from directional diversions property point of view; lambda 2 can be spiral diversions property and then lambda 3, lambda 4 can have this kind of…DR stands for…actually. I think there is a printing mistake again here. This is like, times j actually. So, that is the complex part of this actually. So, this is how it operates actually.

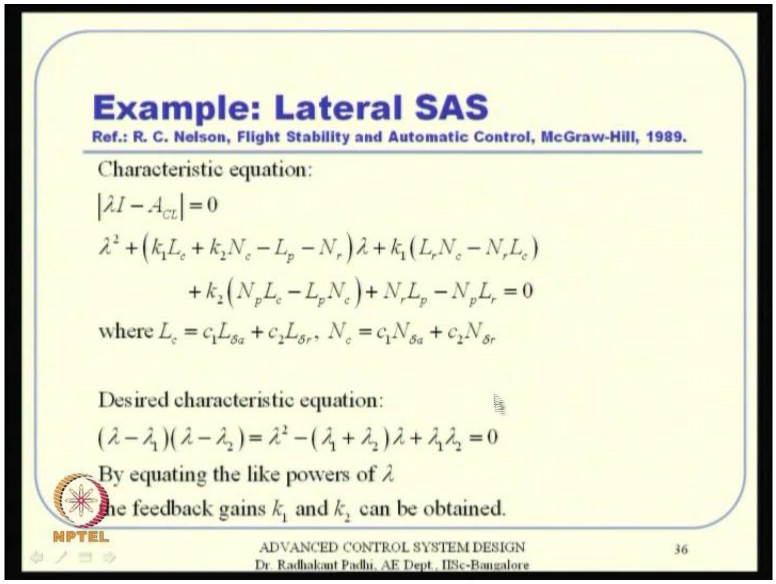
So, again the philosophy is same. Once you know lambda 1 lambda 2 lambda 3 lambda 4, we know the characteristic polynomial again. And, that characteristic polynomial will equate and get the gains actually that way.

(Refer Slide Time: 48:54)



And, just to give a flavor of that, we will consider only this p; let us say delta p and delta r part of it. So, we do not need to go for this fourth order polynomial and all that. But, I mean truly speaking you should work on this fourth order basically. Any way, so if you see this delta p delta r only and that is where this… so, this is how your control is again divided into two parts; automatic part and pilot input. So, automatic part turns out to be like that. Remember, c 1 c 2 is… So, there is no confusion and no need of solving for c 1 c 2. What you solve is for k 1 and k 2, not c 1 c2. So, that is the... that is how we need to solve it actually.

(Refer Slide Time: 49:49)

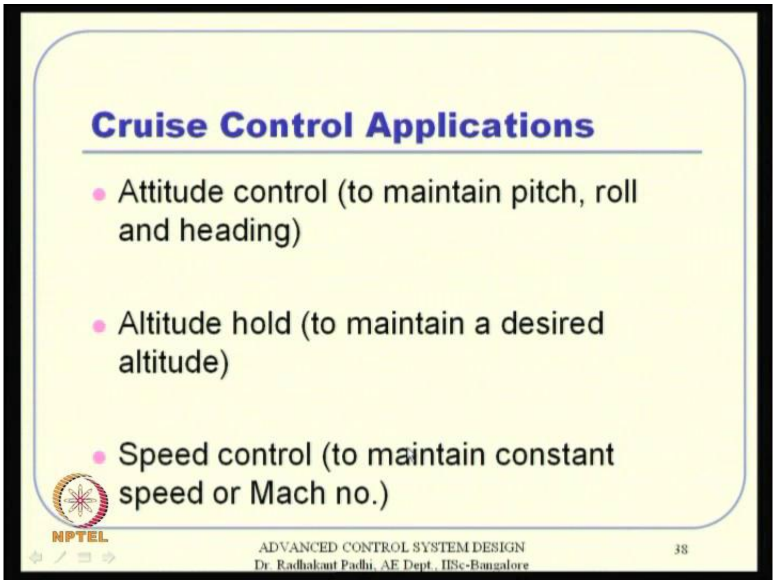


So, the characteristic equation for this system turns out to be like that. It is a simple second order polynomial; the lambda square, lambda and then… actually. So, that will be equal to 0. Where, L c and N c and all you can be defined for further simplicity actually. Remember… the resulting characteristic equation; again lambda 1 lambda 2 you know, so you can multiply and then try to equate the coefficient and think like that. Remember, all these equating coefficients and all you can also use this Bass-Gura approach and think like that. So, no need to go, struggle…

Alright. Before we stop we will also go through a little cruise control system. That is another application. So, in other words, when the aircraft is flying on a cruise mode, ok, there is nothing like pilot input actually. That pilot input; see, whatever we have been doing here is partly it is pilot input, partly it is automatic and think like that.

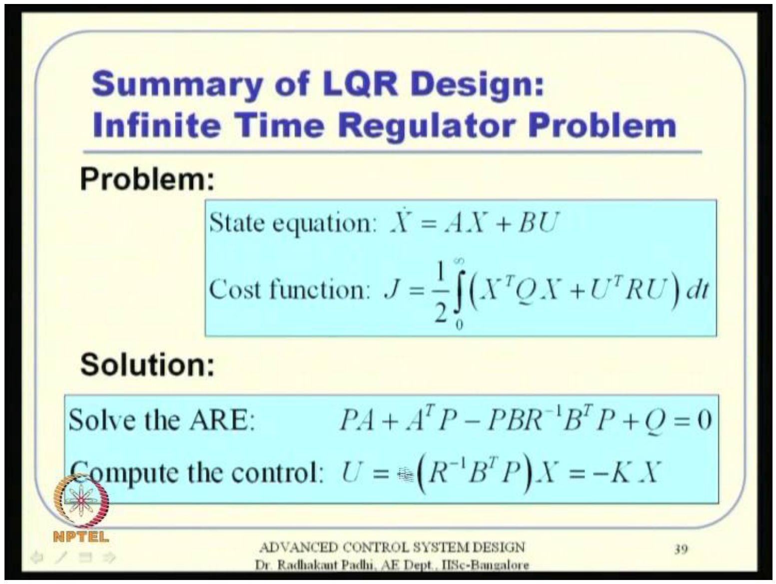
Let us combine this together and give the control input directly actually. That you can interpret that way or you can, I have got closed loop system already. Ok, this is that…Ok, I have got the closed loop system already, so let me also design this part of it in automatic sense. I mean, that means this control input, whatever you are telling we can interpret or okay we can design this control input directly. There is no pilot input... Pilot input is zero actually. Or, that is embedded into … actually. Or, you can design this; this matrix is already available as A CL is already there. So, now this part also we need to design over... actually; so, either way. So, what I will interpret is, I will design the control input. In this class we will assume that we will design the control input U directly, not through the split of mechanism actually. So, that is the, that is what we need to do there actually.

(Refer Slide Time: 52:01)



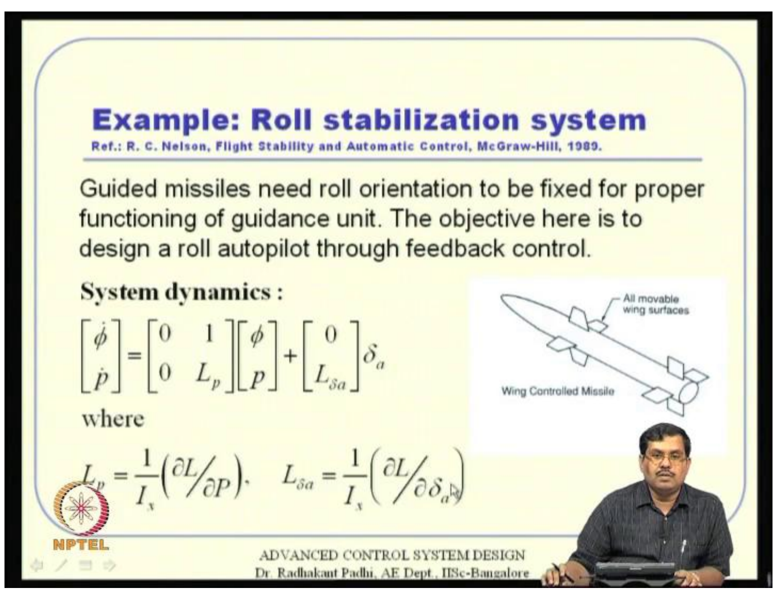
So, cruise control system; again we will revise little bit. You can use this attitude control to maintain pitch, roll and heading angles. And then, altitude hold to maintain a desired altitude. We can also design a speed control actually. And typically, this speed control is kind of a… design sense, it is designed in a decoupled manner, it is never put together because this is slow dynamics actually. And, the attitude and altitude will be fast dynamics; attitude and altitude rate will be fast dynamics. So, these all can be designed in a separate manner actually.

(Refer Slide Time: 52:35)



So, here we will use this LQR type of philosophy. And then, as we discussed this is like, this is the problem, like we are talking here. And, this is the solution that we are talking here actually. You just solve this Riccati equation and then compute the control or in other words compute the gain matrix directly that way actually.

(Refer Slide Time: 52:55)

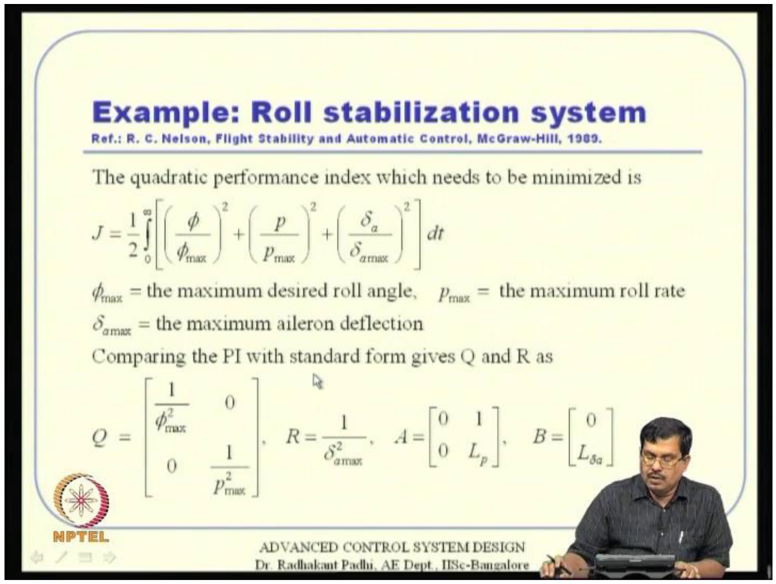


So, the roll stabilization system will, there is an example. We will discuss the stabilizing the roll dynamics of the missile, so that the guidance system can function properly. You know for the guidance system of the missile is largely dependent on; there is some corrections. In other words, the corrections in terms of pitch axis as well as yaw axis actually. So, it is, it should turn about equally or it should change its, of course direction towards the target actually.

Now, once you, while you are doing that you really do not have to; that means the roll should be stabilized actually. If the roll is, roll mechanism is there, first of all, the part of the control effectiveness goes because this, remember this missile is actually symmetric about the roll axis in two planes actually. Once it becomes symmetric in two planes, the characteristic in pitch and yaw are no different actually. So, if it starts rolling and all that because partly it will correct in yaw, and partly it will correct in pitch and then keeps on doing that. Then, it goes to this oscillatory mode sort of thing… and you do not need that actually. So, what is normally done is roll stabilization.

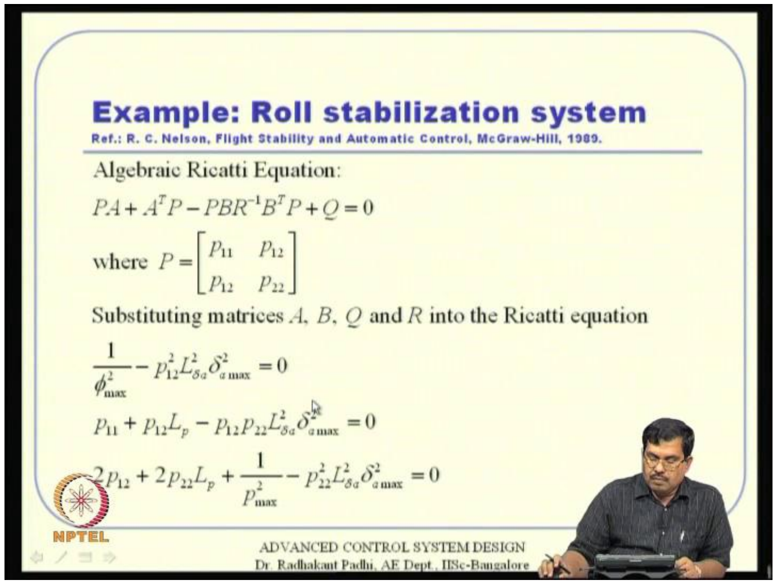
So, you stabilize the roll and then apply the pitch and yaw actually that way. So, how do you stabilize the roll? And, this is, these are the like control surfaces here. Partly it can be there in the front around the c c and partly it can also be there in the tail. These are called fins actually. And, this can be like all movable surfaces. It is not partly movable and all that. The total thing will, kind of move up and down and sort of thing. Anyway, so that tells that…completes that which roll, which axis; I mean depending upon the roll angle, you have to separate it like pitch and yaw fins and thing like that. Ultimately, you can decompose these four fins into three fins, like an aircraft. Then you, I have got like… primarily for roll stabilization and the dynamics considered to be like that. Where L P A is defined like this and L delta is defined like this. We have stability derivatives, which are available to us actually.

(Refer Slide Time: 55:11)



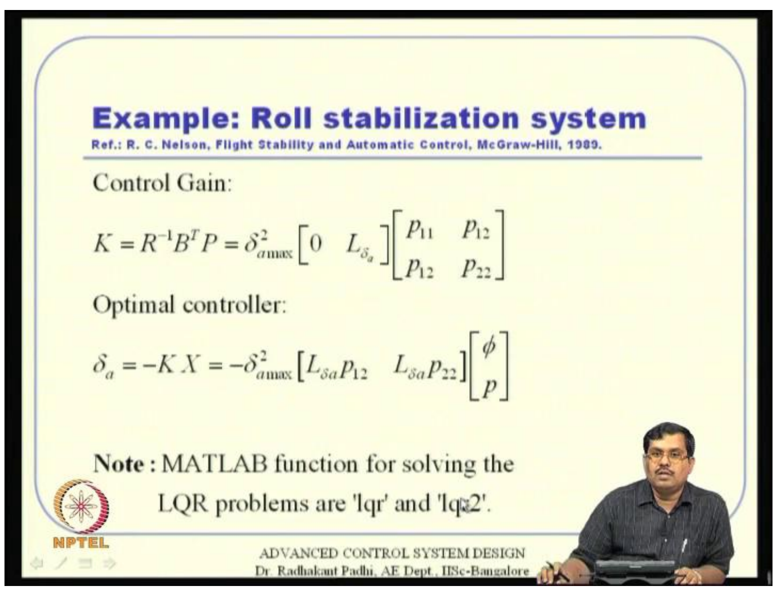
And, the cos function I have selected that way because my main aim is to arrest phi and arrest P. So, phi should be as small as possible and P should be as small as possible. And then, while doing that delta should also be small actually…that way. And, theoretically also remember this coefficients are whatever you are talking, it cannot be 0. It needs to be positive definite actually. So, that is how I select the cos function. And, once I select the cos function, my Q and R matrices are available. Q is like that, R is like that. A and B is already available to me. So, now I go and substitute that in the Riccati equation, matrix Riccati equation. So, then remember Q matrix…symmetric and positive definite solution. …symmetric matrix and then we carry out… this A, B, Q and R are known to us. P is defined like that.

(Refer Slide Time: 55:55)



We will land up with these three equations actually. Once we land up with these three equations, there are multiple solutions. You have to eliminate one, ultimately one first definite solution and all.

(Refer Slide Time: 56:25)

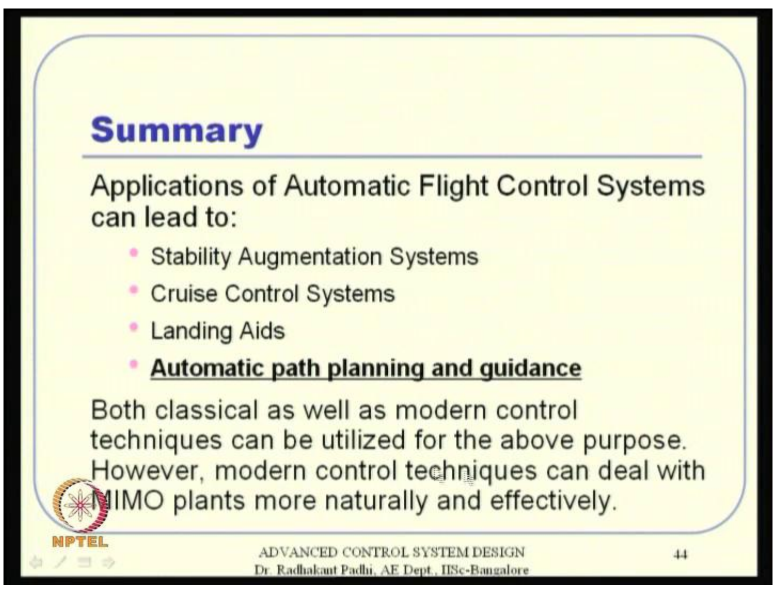


So, ultimately we will be able to find a positive definite P matrix. That is the message actually. Once you find it, then this K turns out to be like that actually. This gain matrix is R inverse B transpose P. So, that will be like that actually.

So, the optimal control what will be needed to act on this system is like that. And, this P one P two, all that solution are available to us from this equations, solutions and all that. And from, this delta a max are also known information…and this gain matrix, gain components… this into that will turn out to be k 1 and this into that will turn out to be k 2. So, this entire mechanism will operate based on that. And then, we are done actually. That is how the roll stabilization can take place.

With the appropriate information of P, I mean phi and P, so all these, remember all these things assume that to make our state value information available, so that control operate based on the gain and all that actually. Also, remember that MATLAB. So, if you use MATLAB, you can actually get a solution on this Riccati equation and all that by using these functions lqr and lqr two… functions already available to you. So, you can take and prove that.

(Refer Slide Time: 57:38)



So, to summarize the application of automatic flight control system can lead to a variety of advantages. It can be used for stability augmentation, cruise control, landing aids as well as what I told automatic path learning and guidance actually. We can also talk many things on that. Both classical as well as modern techniques can be utilized. But, modern control techniques can deal with MIMO plants more naturally and effectively. So, with that message I will stop this class. So, thanks a lot.

.