

Intelligent Systems and Control
Prof. Laxmidhar Behera
Department of Electrical Engineering
Indian Institute of Technology, Kanpur

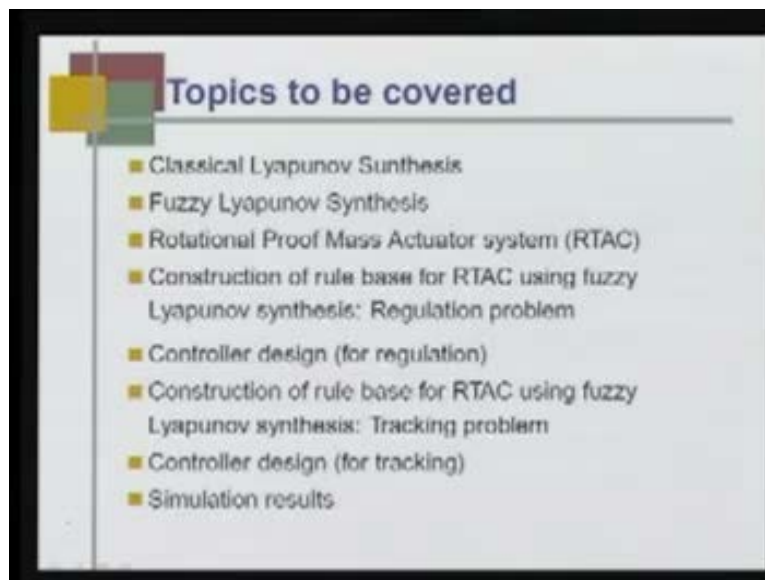
Module – 4 Lecture – 4

Fuzzy Lyapunov controller: Computing with words

Fuzzy Lyapunov controller computing with words, this is the lecture 4 of this module 4 on fuzzy control. As I said in the last class that given the mamdani type of fuzzy logic controller, the membership functions are optimized using genetic algorithms.

But the most important aspect of mamdani type of controller is to generate a proper rule base. Here, of course you can also generate rule base using genetic algorithm but today, we will be learning how to generate a rule base using the concept of Lyapunov function. So this is the topic today, fuzzy Lyapunov controller computing with words.

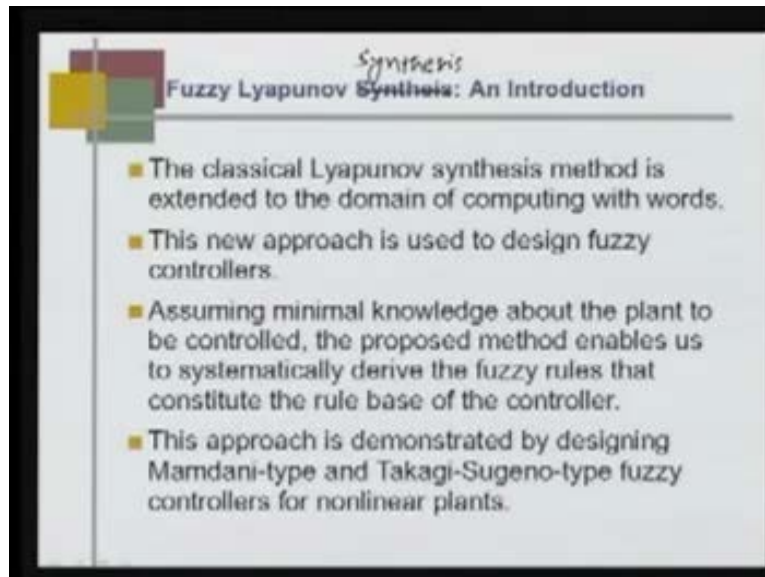
(Refer Slide Time 1:34)



Topics to be covered today are classical Lyapunov synthesis; Fuzzy Lyapunov synthesis; rotational proof mass actuator system, this is the physical system on which you will be testing our algorithm; construction of rule base for RTAC, the short form is RTAC here,

RTAC using fuzzy Lyapunov synthesis: regulation problem; controller design for regulation; construction of rule base for RTAC using fuzzy Lyapunov synthesis: tracking problem; controller design for tracking; and finally, simulation results.

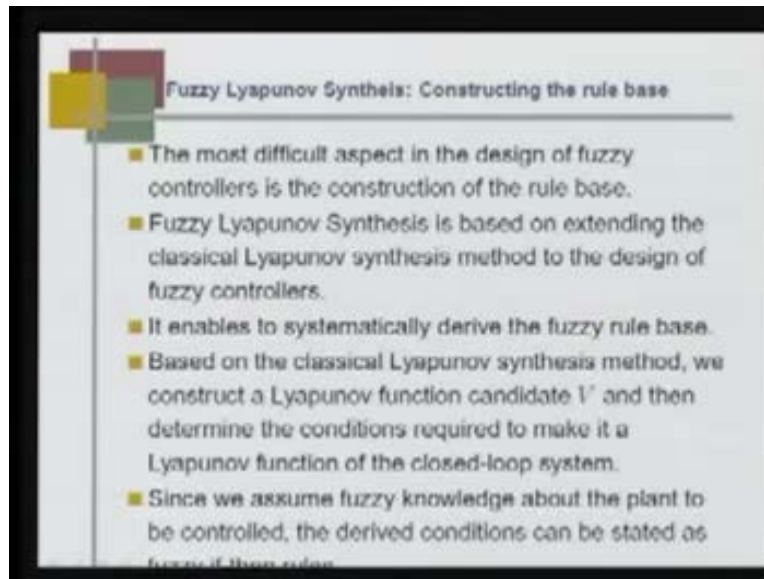
(Refer Slide Time 2:14)



Fuzzy Lyapunov synthesis: an introduction- the classical Lyapunov synthesis method is extended to the domain of computing with words. This new approach is used to design fuzzy controller. Assuming minimal knowledge about the plant be controlled, the proposed method enables us to systematically derive the fuzzy rules that constitute the rule base of the controller. The objective is in this method, how do we obtain fuzzy rule base given the system, some prior in knowledge of the system or approximate model of the system, how do we generate a rule base?

This approach is demonstrated by designing Mamdani-type and Takagi-Sugeno-type fuzzy controllers for nonlinear plants.

(Refer Slide Time 3:1)

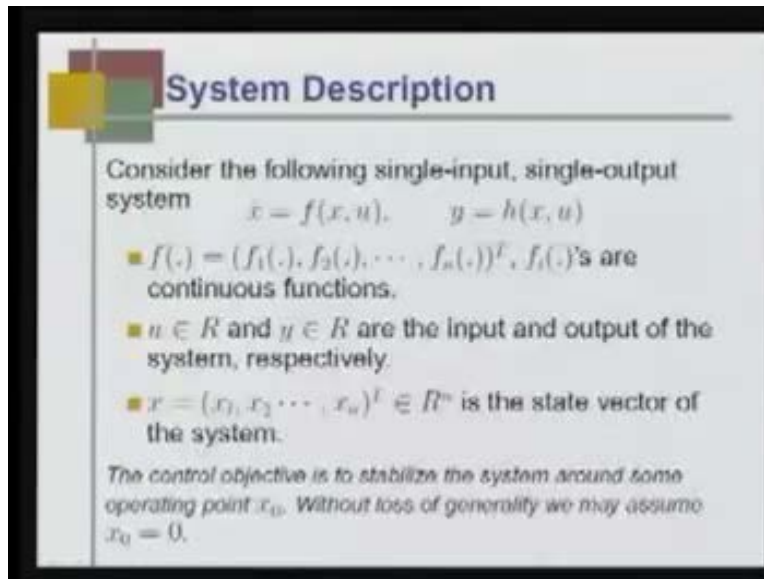


Fuzzy Lyapunov Synthesis: Constructing the rule base

- The most difficult aspect in the design of fuzzy controllers is the construction of the rule base.
- Fuzzy Lyapunov Synthesis is based on extending the classical Lyapunov synthesis method to the design of fuzzy controllers.
- It enables to systematically derive the fuzzy rule base.
- Based on the classical Lyapunov synthesis method, we construct a Lyapunov function candidate V and then determine the conditions required to make it a Lyapunov function of the closed-loop system.
- Since we assume fuzzy knowledge about the plant to be controlled, the derived conditions can be stated as fuzzy if then rules.

The most difficult aspect in the design of fuzzy controllers is the construction of the rule base. Fuzzy Lyapunov synthesis is based on extending the classical Lyapunov synthesis method to design of fuzzy controllers. It enables to systematically derive the fuzzy rule base. Based on classical Lyapunov synthesis method, we construct a Lyapunov function candidate V and then determine the conditions required to make it a Lyapunov function of the closed loop system. Since we assume fuzzy knowledge about the plant to be controlled, the derived conditions can be stated as fuzzy if then rules.

(Refer Slide Time 3:50)



System Description

Consider the following single-input, single-output system

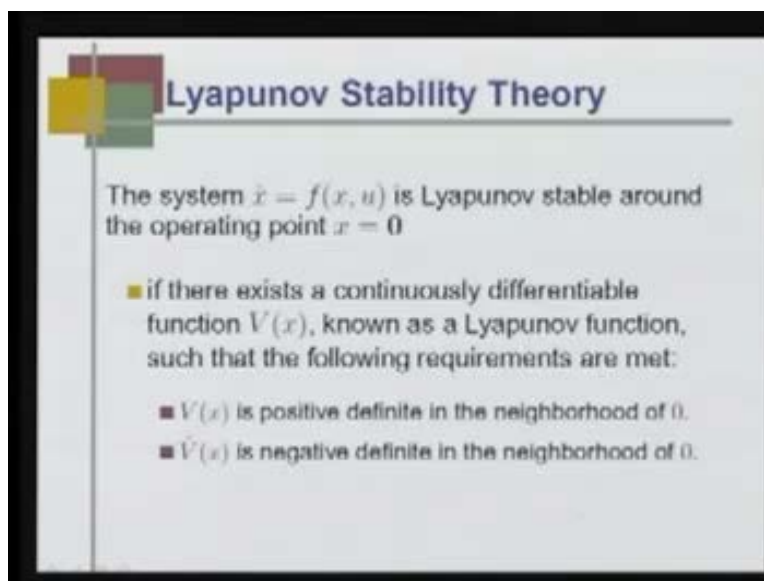
$$\dot{x} = f(x, u), \quad y = h(x, u)$$

- $f(\cdot) = (f_1(\cdot), f_2(\cdot), \dots, f_n(\cdot))^T$, $f_i(\cdot)$'s are continuous functions.
- $u \in R$ and $y \in R$ are the input and output of the system, respectively.
- $x = (x_1, x_2, \dots, x_n)^T \in R^n$ is the state vector of the system.

The control objective is to stabilize the system around some operating point x_0 . Without loss of generality we may assume $x_0 = 0$.

So now let us go to the system description. Consider the following single-input, single-output system. $\dot{x}$ equal to $f(x, u)$, y is $h(x, u)$ where f and h are nonlinear functions. This is n dimensional state space, so f is a n dimensional vector; and u and y they are the input and outputs of the system respectively; x n dimensional system x_1 , this is x_1 x_2 until x_n , R^n is the state vector of the system. The control objective is to stabilize the system around some operating point x_0 . Without loss of generality we may assume x_0 equal to 0.

(Refer Slide Time 4:40)



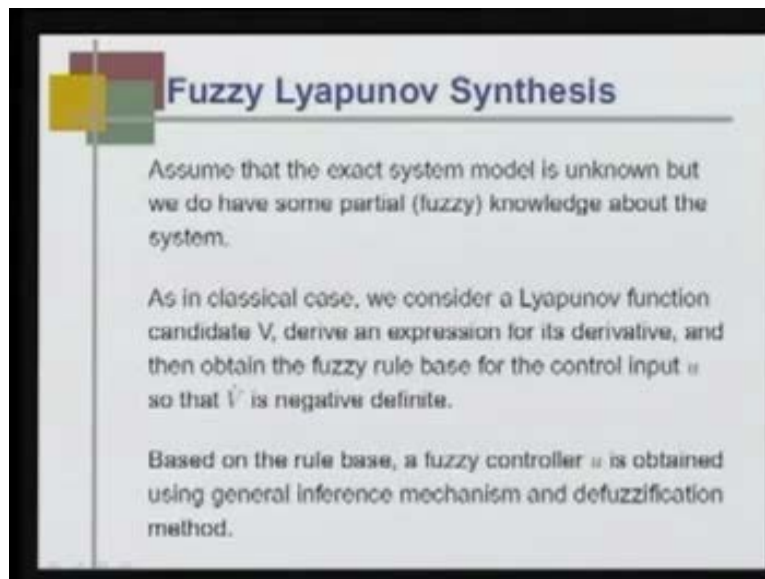
Lyapunov Stability Theory

The system $\dot{x} = f(x, u)$ is Lyapunov stable around the operating point $x = 0$

- if there exists a continuously differentiable function $V(x)$, known as a Lyapunov function, such that the following requirements are met:
 - $V(x)$ is positive definite in the neighborhood of 0.
 - $\dot{V}(x)$ is negative definite in the neighborhood of 0.

So Lyapunov stability theory which we have been talking again and again, I am just revisiting that, the system $\dot{x} = f(x, u)$ is Lyapunov stable around the operating point $x = 0$. If there exists a continuously differentiable function $V(x)$ which is known as Lyapunov function such that the following requirements are met. $V(x)$ is positive definite in the neighborhood of 0 and $\dot{V}(x)$ is negative definite in the neighborhood of 0. If we can ensure that then the system is Lyapunov stable.

(Refer Slide Time 5:6)



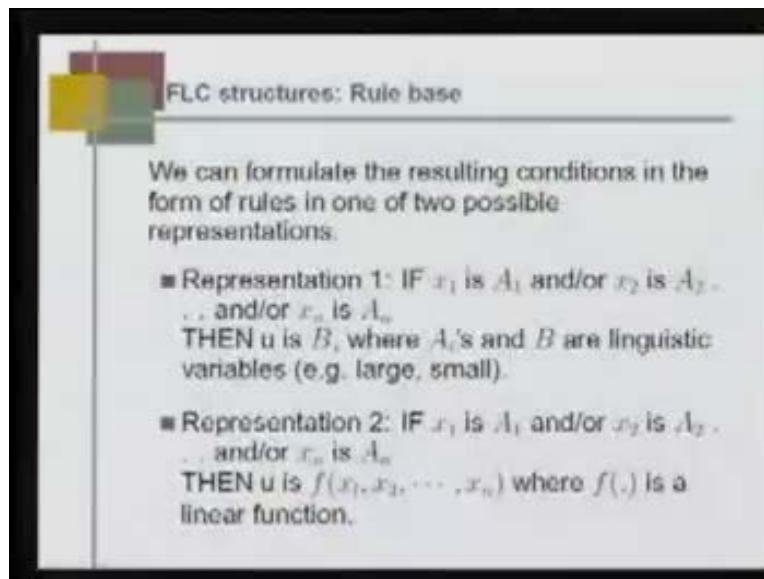
So fuzzy Lyapunov synthesis- assume that the exact system model is unknown but we do have some partial fuzzy knowledge about the system. As in classical case we consider a Lyapunov function candidate V , derive an expression for its derivative and then obtain the fuzzy rule base for the control input u , so that $\dot{V}$ is negative definite. Based on the rule base, a fuzzy controller u is obtained using general inference mechanism and defuzzification method.

What is the objective here is that we prescribe a Lyapunov function for the system and then we construct, find the right derivative of the Lyapunov function, substitute the dynamics of the system in that and now we create a rule base in such a way this Lyapunov function is qualitatively negative definite, not quantitatively. So what is fuzzy

Lyapunov synthesis is we are using a qualitative negative definite of $\dot{v}$. This is the difference qualitative negative definiteness of $\dot{v}$, instead of a quantitative negative.

The actual Lyapunov function approach to test the stability of the system is to find quantitatively $\dot{v}$ is negative definite. But now we are talking about a concept qualitatively, if we can make $\dot{v}$ to be negative definite.

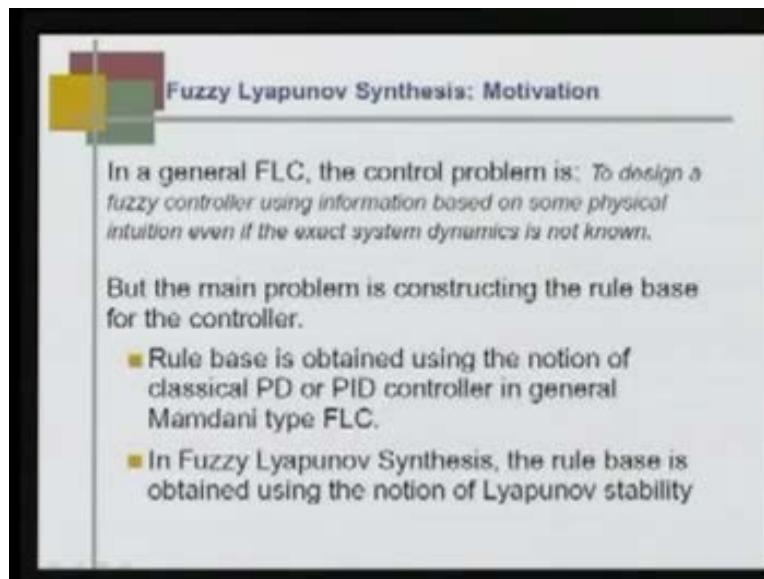
(Refer Slide Time 7:18)



FLC structures of the rule base, so normal we have two kinds of we can formulate any fuzzy logic controller of mamdani type. We can have two categories of fuzzy rule base. The representation 1 is if x_1 is A_1 and or x_2 is A_2 and or x_n is A_n then u is B where A_i 's and B are linguistic variables large or small. So x_1 x_2 and x_n they are either state variables or error in state variables.

Representation 2 is if x_1 is A_1 and or x_2 is A_2 and so on or x_n is A_n then u is a definite function here the B , u is B are the fuzzy linguistic variable where in the second representation u is an exact function. u is $f(x_1 \ x_2 \ \dots \ x_n)$ where f is a linear function; this can be also nonlinear function, f can be also linear function here, normally is a linear function.

(Refer Slide Time 8:27)



So fuzzy Lyapunov synthesis: motivation- what is the motivation here? In a general fuzzy logic controller, the control problem is to design a fuzzy controller using information based on some physical intuition even if the exact system dynamics is not known.

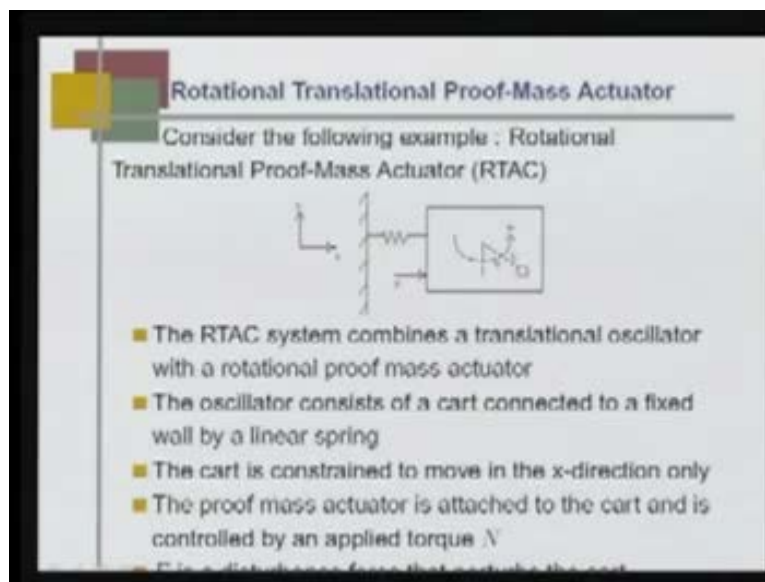
But the main problem is constructing the rule base for the controller. Rule base is obtained using the notion of classical PD or PID controller in general for mamdani type of fuzzy logic controller. But in a fuzzy Lyapunov synthesis a rule base is obtained using the notion of Lyapunov stability. So this is the prime motivation. I can tell you how normally we do a fuzzy rule base, if this is my general characteristic of a second order system. So what I would like, when I describe this response, this is my t , this is my y (Refer Slide Time: 09:29), my output. So you see that if I am here at this zone, so at this zone I can say error, error at this zone is very large.

I would like when I am here I should come very fast to this line, which is my desired command signal 1.0. What I will say, when my error is small and the derivative of error, the rate of change in error is either small or large whatever it is, when it is small I would like it to come fast. My controller must be very large. I can write for this kind of PD or PID type of controller the rule base is if e is small in this zone and $\dot{e}$ is small or large then u is large. By simple commonsense I can write.

Similarly, if I say e is small here and if e is small in this zone and $\dot{e}$ is also small then u is small because I have already reached almost I am reaching the set point. But if e is small and $\dot{e}$ is large, so obviously I must reduce the control action, so that is, in that case I have to say u is negative small, instead of positive small. Like that the normal rule base is done in case of PD or PID type of controller.

But today we will see a very systematic method of generating rule base. Actually this entire class today is to let you know or we will learn how to generate this rule base systematically using Lyapunov stability theory. Before that, I will just explain to you one physical system on which will be applying this thing, this is called rotational translational proof mass actuator.

(Refer Slide Time 12:00)



You can actually see this is my rigid surface to which a cart is bound using a spring. Inside the cart there is the pendulum which rotates by the action of the torque and the pendulum link length is l and there is an external disturbance f that is applied on the cart. See the objective is to control this pendulum the way we like.

I may like to keep this pendulum at an angle θ or vertically downward vertically upward, whatever is the desired thing or it has to follow certain trajectory, so whatever

may be the requirement this is a typical control system because it is nonlinear and also it a bench mark problem.

The RTAC system, consider the following example rotation translational proof mass actuator RTAC. The RTAC system combines a translational oscillator with a rotational proof mass actuator. This particular portion now oscillates by the action of the spring. So this called combines a transitional oscillator. This oscillates in the x direction with a rotational proof mass actuator. This is rotation. The oscillator consists of a cart connected to fixed wall by a linear spring. The cart is constrained to move in the x-direction only. The proof mass actuator is attached to the cart and is controlled by an applied torque N. F is disturbance force that perturbs the cart. F is a disturbance force that gives such a disturbance input.

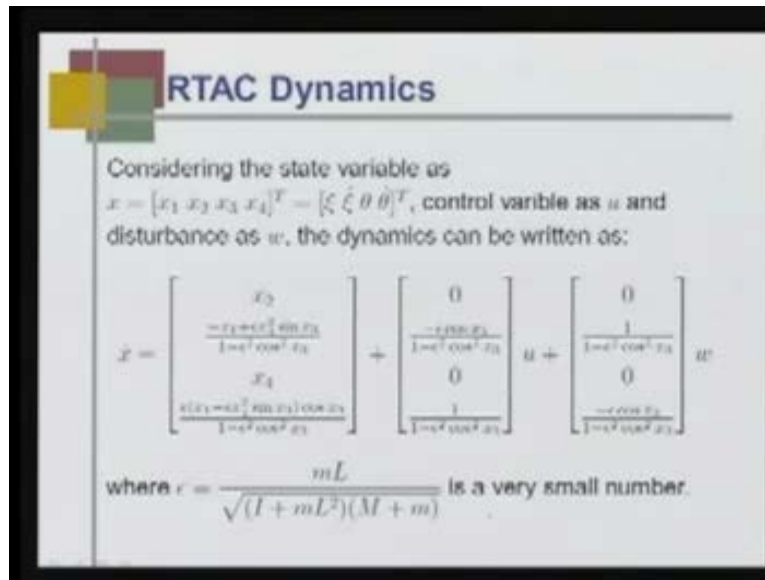
(Refer Slide Time 13:58)

RTAC paramters	
The cart position and the actuator angle is x_c and θ respectively.	
Mass of the cart	$M=1.36$ Kg
Proof mass	$m=0.096$ Kg
Distance	$L=0.059$ m
Moment of inertia(proof mass)	$I=0.00022$ Kg-m ²
Spring constant	$k=186.3$ N/m
Define the normalized variable as:	
$\xi = x_c \sqrt{\frac{M+m}{L+mL^2}} \quad \tau = t \sqrt{\frac{k}{M+m}}$	
$u = N \frac{M+m}{L+mL^2} \quad w = \frac{1}{L} \sqrt{\frac{M+m}{k}} F$	

The cart position and actual angle is x_c and theta respectively, that is the cart which you saw here this is x_c and this theta already which have already shown. Mass of the cart is given 1.36 Kg; proof mass is 0.096 that is the mass of this cart is M and proof mass is small m. So small m is 0.096 Kg; distance of the link length of the pendulum is 0.059 meter; Moment of inertia of the proof mass is I is 0.00022 Kg meter square; Spring constant k is given 186.3 Newton per meter.

Now we define some normalized variables which is ζ is x_c root over $M + m$ upon $I + mL$ square; τ is t k upon $M + m$; and u is this is my actual input and $M + m$ upon k into $1 - mL$ square; K is the spring constant; and ω is 1 upon $k M + m$ by $I + mL$ square F .

(Refer Slide Time 15:28)



RTAC Dynamics

Considering the state variable as $x = [x_1 \ x_2 \ x_3 \ x_4]^T = [\zeta \ \dot{\zeta} \ \theta \ \dot{\theta}]^T$, control variable as u and disturbance as w , the dynamics can be written as:

$$\dot{x} = \begin{bmatrix} x_2 \\ \frac{-x_1 + c x_1^2 \sin x_3}{1 - c^2 \cos^2 x_3} \\ x_4 \\ \frac{c(x_1 - c x_1^2 \sin x_3) \cos x_3}{1 - c^2 \cos^2 x_3} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{-c \sin x_3}{1 - c^2 \cos^2 x_3} \\ 0 \\ \frac{1}{1 - c^2 \cos^2 x_3} \end{bmatrix} u + \begin{bmatrix} 0 \\ \frac{1}{1 - c^2 \cos^2 x_3} \\ 0 \\ \frac{-c \cos x_3}{1 - c^2 \cos^2 x_3} \end{bmatrix} w$$

where $c = \frac{mL}{\sqrt{(I + mL^2)(M + m)}}$ is a very small number.

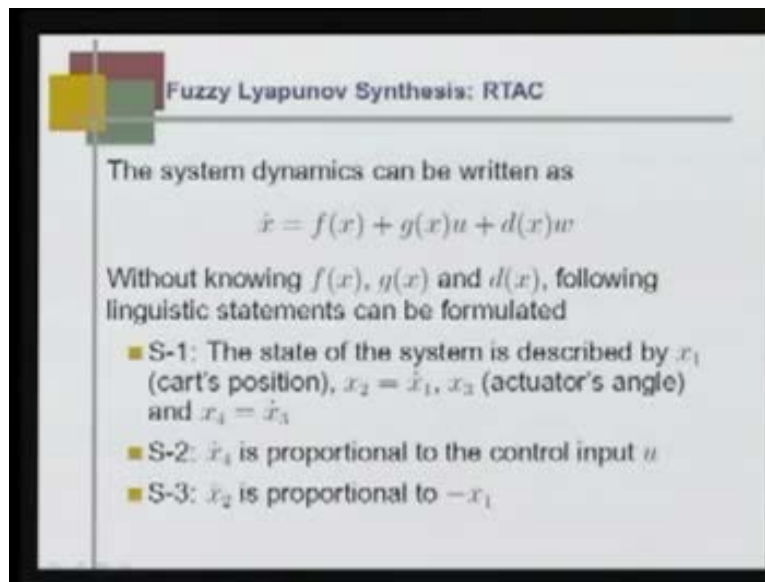
Given that, you see that we can write, we can derive the dynamics of the RTAC system, you can easily see that it has four states. x_1 is ζ is actually represented x_c normalized; this is x_c dot; the velocity of the cart normalized which is x_2 ; and θ is the pendulum angle x_3 ; $\dot{\theta}$ is the angular velocity of the pendulum which is x_4 ; and control variable is u which is also normalized; n the torque; and disturbance is w , so this one w is the disturbance.

If we look at $\dot{x}$ this is the fourth order system and the expressions are $\dot{x}$ equal to x_2 , the advantage is that we have not kept any parameters associated, this is simplified $\dot{x}_2$, you can see $\dot{x}$ is in this, this I can say, this is simply a function of x , this is another function of x may be $g(x)$, and this is $d(x)$.

You can say this is a function vector with four dimensions, with function vector with another four dimensions and another function vector with four dimensions. You can see that system is highly nonlinear if ζ is defined as mL upon root over $I + mL$ square

into M plus m , you can easily derive this equation, I will not discuss further on this. All that I am trying to tell you is that RTAC system has a nonlinear dynamics which has type $\dot{x} = f(x) + g(x)u + d(x)w$, where w is the external disturbance; u is actual control input.

(Refer Slide Time 17:43)



Fuzzy Lyapunov Synthesis: RTAC

The system dynamics can be written as

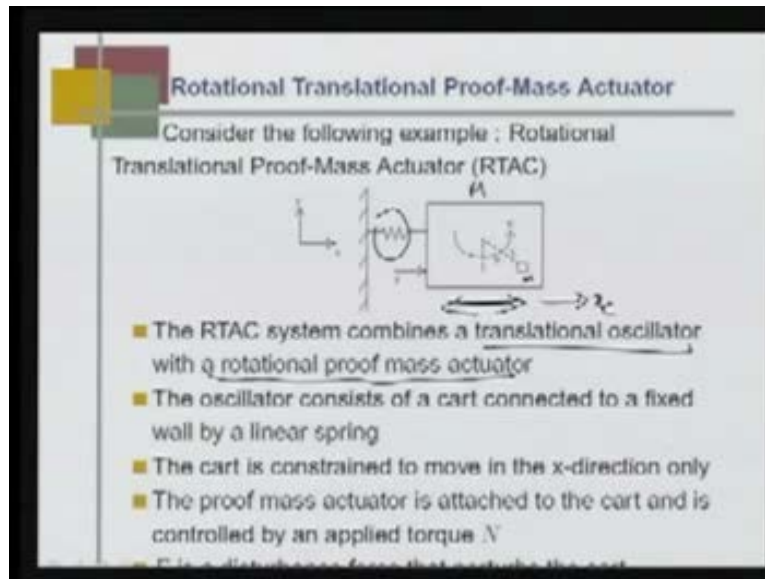
$$\dot{x} = f(x) + g(x)u + d(x)w$$

Without knowing $f(x)$, $g(x)$ and $d(x)$, following linguistic statements can be formulated

- S-1: The state of the system is described by x_1 (cart's position), $x_2 = \dot{x}_1$, x_3 (actuator's angle) and $x_4 = \dot{x}_3$
- S-2: $\dot{x}_4$ is proportional to the control input u
- S-3: $\dot{x}_2$ is proportional to $-x_1$

The fuzzy Lyapunov synthesis of RTAC- the system dynamics now can be written as, as we said $\dot{x} = f(x) + g(x)u + d(x)w$. Without knowing $f(x)$, $g(x)$ and $d(x)$ following linguistic statements can be formulated. Even if I do not know what is $f(x)$, $g(x)$, $d(x)$ exactly, I can still write the state of the system is described by x_1 the cart's position; x_2 equal to $\dot{x}_1$; and x_3 is actuator's angle; and of course x_4 is $\dot{x}_3$. That is x_3 is theta, x_4 is theta dot, x_1 is x_c normalized, x_2 is $\dot{x}_1$. This is another important thing that I can easily tell you, $\dot{x}_4$ is proportional to the control input u that you can easily see here.

(Refer Slide Time 18:46)



Because you see that this is the my pendulum and if I assume that there is no friction and other things here, then my pendulum, the acceleration of this pendulum, angular acceleration obviously is directly proportional to the torque applied N and since my input u is normalized value of N and hence I can always say that $\ddot{x}_4$, the acceleration of the pendulum is proportional to the control input u .

Similarly, $\dot{x}_2$ the velocity of the cart is proportional to the negative of x_1 , you can easily see that because the spring is connected so it is like this, spring and this is the block mass. You can see that $\dot{x}_2$ the velocity is always, because of spring action, it will be always in this direction the velocity. If I move x in this direction, x_2 the position in this direction, the velocity is in the negative direction. So $\dot{x}_2$ in this is proportional to minus x_1 .

(Refer Slide Time 19:59)

Fuzzy Controller: Regulation problem

The control objective is to find a u such that the system is stable around $x = [x_1 \ x_2 \ x_3 \ x_4]^T = 0$

Let $V = \frac{1}{2}(x_1^2 + x_2^2 + x_3^2 + x_4^2)$ be the Lyapunov function candidate. Then, using S-1,

$$\dot{V} = x_1\dot{x}_1 + x_2\dot{x}_2 + x_3\dot{x}_3 + x_4\dot{x}_4$$

using S-3 and S-2,

$$\dot{V} \approx x_3x_4 + x_4u$$

Classical Lyapunov synthesis suggests the design of u that will guarantee $\dot{V} < 0$. Same idea is applicable here but in the domain of computing with words.

The control objective is to find u such that the system is stable around x equal to 0. This is the regulation problem. Let V is half x_1 square plus x_2 square plus x_3 square plus x_4 square be the Lyapunov function candidate. Then using S-1, the assumption that we have four states V dot is $x_1 \cdot x_1$ dot plus $x_2 \cdot x_2$ dot plus $x_3 \cdot x_3$ dot plus $x_4 \cdot x_4$ dot, because we know already that x_2 is x_1 dot and x_4 is x_3 dot, so this second step comes.

Now you see that, we have already said x_2 dot is proportional to minus x_1 . Obviously, qualitatively if I put x_2 dot is minus x_1 , these two terms qualitatively cancels off, where as only these two terms remain, $x_3 \cdot x_4$ is remained and I know already that x_4 dot is proportional to u , so instead of x_4 dot I put u .

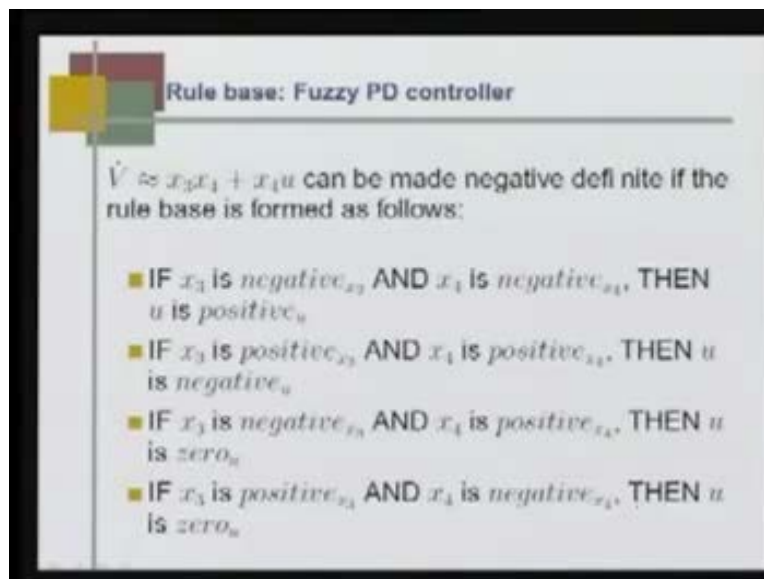
You see that, what is the advantage here is that I am designing a rule base but while designing rule base I am not utilizing the system dynamics. I am finding how to what V dot is equal to $x_3 \cdot x_4$ plus $x_4 u$. While finding this equation, I am not using the system dynamics. This is very important, system dynamics is not used.

What is used is the simple physics of the system is that, here we use x_2 equal to $\dot{x}_1$ obviously the velocity x_2 and x_1 is positioned and similarly, here we use the notion that the x_4 is $\dot{x}_3$ dot, so x_4 is theta dot and x_3 is theta, which is very simple and also we used other rule from the systemic connections.

We know already that the acceleration of the pendulum is proportional to the input and x_2 dot is proportional to minus x_1 , just by constructional feature. Using these two rules, we found $\dot{V}$ to be qualitatively $x_3 \dot{x}_4$ plus x_4 into u . Now, even if you do not know the system dynamics, looking at this figure we can create a rule base for which this $\dot{V}$ can be qualitatively negative definite.

So classical Lyapunov synthesis have design of view that will guarantee $\dot{V}$ is negative definite. Same idea is applicable here but with the domain of computing with words means, this is qualitatively not quantitative. What we are trying to do is we are applying Lyapunov synthesis to the system dynamics partially known system dynamics for generation of rule base.

(Refer Slide Time 23:54)



Rule base: Fuzzy PD controller

$\dot{V} \approx x_3 \dot{x}_4 + x_4 u$ can be made negative definite if the rule base is formed as follows:

- IF x_3 is *negative_{x3}* AND x_4 is *negative_{x4}*, THEN u is *positive_u*
- IF x_3 is *positive_{x3}* AND x_4 is *positive_{x4}*, THEN u is *negative_u*
- IF x_3 is *negative_{x3}* AND x_4 is *positive_{x4}*, THEN u is *zero_u*
- IF x_3 is *positive_{x3}* AND x_4 is *negative_{x4}*, THEN u is *zero_u*

Now we take the example, when I say so $\dot{V}$ derived $\dot{V}$ equal to approximately qualitatively $x_3 \dot{x}_4$ plus x_4 into u . This can be made negative definite if the rule base is formed as follows. You see that I will take care that you understand is because this is

what you should learn how to generate a rule base, you see x_3 is negative. If x_3 is negative and x_4 is also negative, that means this quantity is positive. Now x_4 is negative and if u would be come negative then this whole thing is positive, so no way you can make the system is stable. Obviously, the rule base should be u to be positive, that is the rule.

I repeat if x_3 is negative and x_4 is negative, in that case $x_3 x_4$ is positive, this implies x_4 into u should be negative to make these two quantities to be qualitatively negative. So x_4 u should be negative, x_4 is already known as negative, so u has to be positive. This is first rule. If x_3 is negative and x_4 is negative then u is positive. Similarly, if x_3 is positive and x_4 is positive. x_3 is positive and x_4 is positive means again x_3 and x_4 is positive in this case. And this implies again x_4 and u should be negative and since x_4 is positive so u has to be negative which is negative. The second rule is if x_3 is positive and x_4 is positive then u is negative. I hope you are able to follow what I am presenting you.

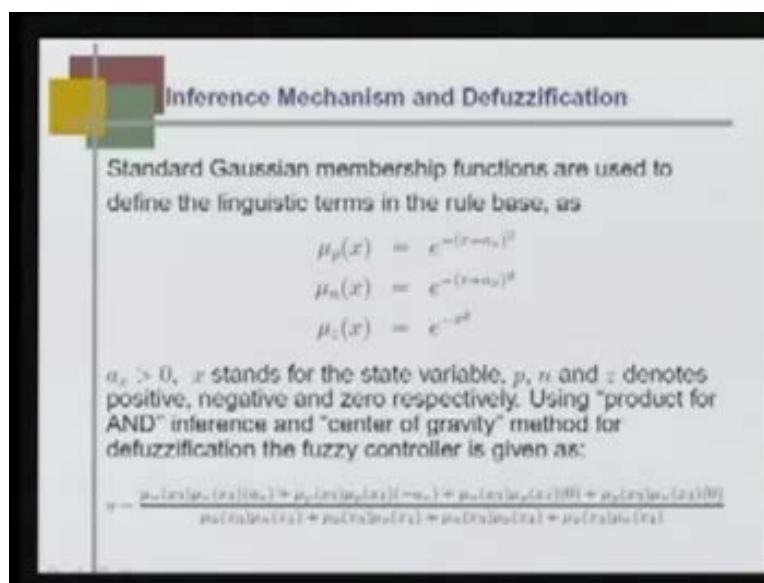
Now let us see the other possible combination. If x_3 is negative and x_4 is positive, in this case $x_3 x_4$ is negative. If $x_3 x_4$ is negative I must ensure that $x_4 u$, since it already negative this should be 0 is sufficient, if I make x_4 is 0 because $x_3 x_4$ is negative is very good. This is simple because there are other ways also is $x_3 x_4$ is negative I can say $x_4 u$ is also negative much better. But this is simplest one x_4 .

x_3 is negative and x_4 is positive then u is 0, you can also think another possibility, because x_4 is positive u is negative, this is also possible. Similarly if x_3 is positive and x_4 is negative then again $x_3 x_4$ in this case x_3 and x_4 sorry x_3 negative, x_3 is positive and x_4 is negative, $x_3 x_4$ is negative. So this implies you can make $x_4 u$ is 0 or $x_4 u$ again negative. Since x_4 is negative so u is 0 or u is positive, but in this case we have only considered this 0.

I wish this is the most important part of this lecture, first of all using partial knowledge of the system and without taking the help of system dynamics we first of all found out the rate derivative of the Lyapunov function is approximately qualitatively equal to $x_3 x_4$ plus $x_4 u$ and using this expression we computed set of rule base such that $\dot{V}$ can be negative definite.

What we have not until now found out in this is the parameters negative, what are the parameter associated with negative. I am assuming this fuzzy linguistic variable to be Gaussian functions, which we have taken in this case. Gaussian function is normally represented by two parameters- one is mean and another is sigma. If I say sigma is 1 then these linguistic variables they are all represented by one parameter that is the mean and the u is positive or negative that we have defined here, we have taken them to be single term.

(Refer Slide Time 29:38)



Inference Mechanism and Defuzzification

Standard Gaussian membership functions are used to define the linguistic terms in the rule base, as

$$\mu_p(x) = e^{-(x-a_p)^2}$$

$$\mu_n(x) = e^{-(x+a_n)^2}$$

$$\mu_z(x) = e^{-x^2}$$

$a_p > 0$, x stands for the state variable, p , n and z denotes positive, negative and zero respectively. Using "product for AND" inference and "center of gravity" method for defuzzification the fuzzy controller is given as:

$$u = \frac{\mu_p(x_1)\mu_p(x_2)\mu_p(x_3) + \mu_p(x_1)\mu_p(x_2)\mu_n(x_3) + \mu_p(x_1)\mu_n(x_2)\mu_n(x_3) + \mu_p(x_1)\mu_n(x_2)\mu_z(x_3)}{\mu_p(x_1)\mu_p(x_2) + \mu_p(x_1)\mu_p(x_2) + \mu_p(x_1)\mu_p(x_2) + \mu_p(x_1)\mu_p(x_2)}$$

Standard Gaussian membership functions are used to defined the linguistic terms in the rule base as mu positive of x is e to the power minus x minus x whole square, this is the parameter that we have to identify. Similarly, mu negative membership function of negative of the variable x is e to the power minus x plus x whole squared. And mu z x is 0 e to the power minus x square.

So x represents mean, x stands for state variable, p, n, z denotes positive, negative and zero respectively. Using the product for AND inference and center of gravity method for defuzzification the fuzzy controller is given as we have four rules. So for four rules the u if you look at here what I will do is that in this I find out the membership function of this, the membership function of this. So this is mu x₃ into mu x₄ so that is the membership

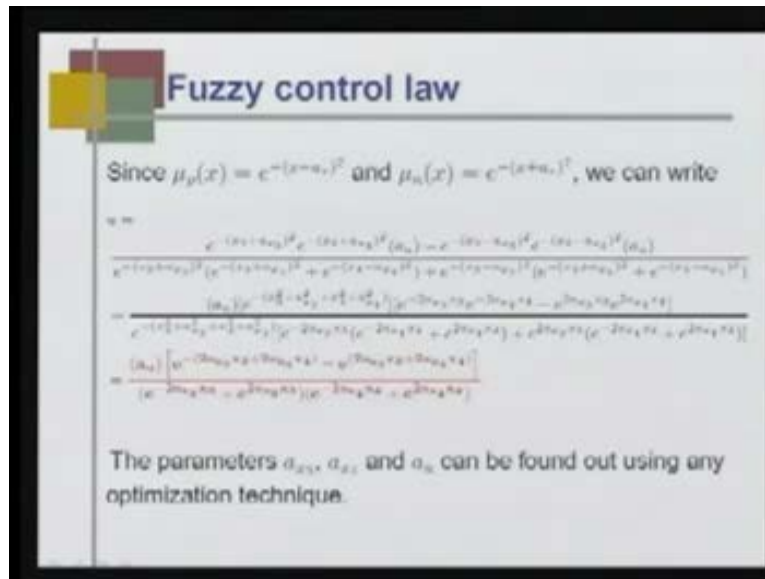
function for my control action and this is the fuzzy singleton which I define as a_u , the fuzzy singleton, therefore my control action is a_u .

$\mu_{\text{negative } x_3}$, $\mu_{\text{negative } x_4}$ into a_u is the output of the first rule. Here, again to let to know I will again explain, here how we are actually doing this I do not have to do every thing again you can easily understand, what is control action from this rule base? Obviously, I find out what is μ and x_3 , and μ and x_4 the membership function of this into u is positive and this is a singleton and that is why this is a_u .

a_u is the singleton value associated with this control action a_u , simply multiplied. This is the output of this controller, similarly you can easily see this is output of the rule base, this rule and the second rule output will be $\mu_p x_3$ into $\mu_p x_4$ into this is negative, so minus a_u . That is what we doing here.

So μ and x_3 , μ and x_4 a_u , $\mu_p x_3$, $\mu_p x_4$ minus a_u I will not have been considered this because we are assuming. Thus you see that this is 0 if I simply say the control action is 0, because control action is 0 this is sufficient for me to make this rate derivative to be negative definite. Simply making that what advantages I am gaining is that, I am not reducing the parameters that has to be optimized or heuristically that has to be tuned. This is my control action, by using center of gravity method you just add all these individual membership functions associated with control action.

(Refer Slide Time 33:35)



Fuzzy control law

Since $\mu_p(x) = e^{-(x-a_p)^2}$ and $\mu_n(x) = e^{-(x+a_n)^2}$, we can write

$$u = \frac{e^{-(x-a_p)^2} e^{-(x+a_n)^2} (a_n) - e^{-(x-a_p)^2} e^{-(x+a_n)^2} (a_p)}{e^{-(x-a_p)^2} e^{-(x+a_n)^2} (a_n) + e^{-(x-a_p)^2} e^{-(x+a_n)^2} (a_p)}$$

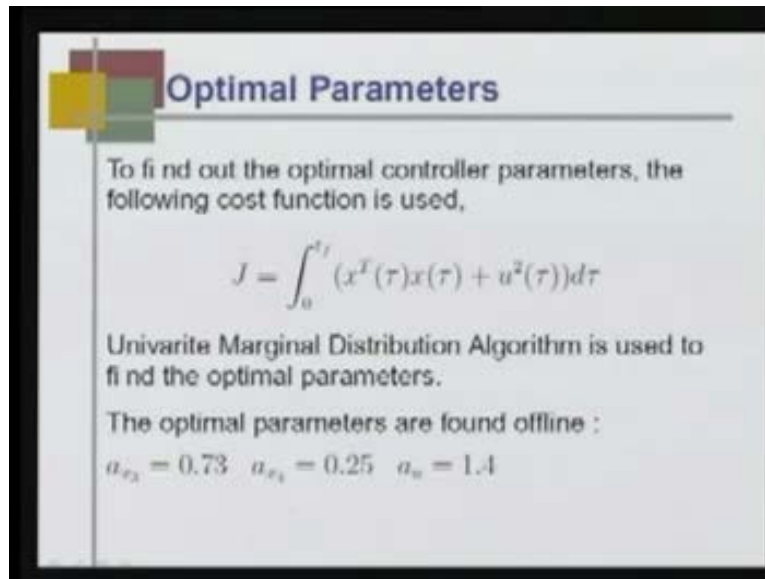
$$= \frac{(a_n) e^{-(x-a_p)^2} e^{-(x+a_n)^2} - (a_p) e^{-(x-a_p)^2} e^{-(x+a_n)^2}}{e^{-(x-a_p)^2} e^{-(x+a_n)^2} (a_n + a_p)}$$

$$= \frac{(a_n - a_p)}{(a_n + a_p)}$$

The parameters a_{p3} , a_{p2} and a_n can be found out using any optimization technique.

So there are 4 and finally, you get the value by taking $\mu_p \cdot x$ is e to the power minus x minus x whole square in μ and x is e to the power minus x plus x whole square, you can write u to this function. This is your control law. This is nothing to do because we have taken Gaussian function. Finally you can simplify this u , but you also can take any other function triangular function or whatever function you want to like, but if you look at this control action what are the parameters that you do not know, this one is a_u , another is a_{x3} , another is a_{x4} , all other things x_3 x_4 , these quantities that you are getting feedback from the system. So these three parameters they have to be found and we can do any kind of optimization algorithm.

(Refer Slide Time 34:29)



Optimal Parameters

To find out the optimal controller parameters, the following cost function is used,

$$J = \int_0^{t_f} (x^T(\tau)x(\tau) + u^2(\tau))d\tau$$

Univariate Marginal Distribution Algorithm is used to find the optimal parameters.

The optimal parameters are found offline :

$$a_{e_3} = 0.73 \quad a_{e_4} = 0.25 \quad a_u = 1.4$$

And particularly we have used genetic algorithm, particularly univariate marginal distribution algorithm. We have already described in the previous classes what univariate marginal distribution algorithm is.

This has been used to minimize this cost function and by minimizing that, the optimal parameters are found to be the mean of the linguistic variable for x_3 is 0.73 and for x_4 is 0.25, this either positive or negative. So this is the mean and u is control action singleton is 1.4. This was done, so this design is over, now we can simulate the system which will show the results at the end of the section.

(Refer Slide Time 35:31)

Rule base: Fuzzy state feedback controller

Assume that u has a form $u = -k_1x_3 - k_2x_4$. Putting the expression of u in $\dot{V} = x_3x_4 + x_4u$,

$$\dot{V} = x_3x_4 + x_4(-k_1x_3 - k_2x_4) = (1-k_1)x_3x_4 - k_2x_4^2$$

To make $\dot{V}$ negative, the following rule base is formed

- IF x_3 is *negative* _{x_3} AND x_4 is *negative* _{x_4} , THEN u is $u = -k_{1_{neg}}x_3 - k_{2_{neg}}x_4$ where $k_{1_{neg}} \geq 1, k_{2_{neg}} > 0$
- IF x_3 is *positive* _{x_3} AND x_4 is *positive* _{x_4} , THEN u is $u = -k_{1_{pos}}x_3 - k_{2_{pos}}x_4$ where $k_{1_{pos}} \geq 1, k_{2_{pos}} > 0$
- IF x_3 is *negative* _{x_3} AND x_4 is *positive* _{x_4} , THEN u is $u = -k_{1_{neg}}x_3 - k_{2_{pos}}x_4$ where $k_{1_{neg}} \leq 1, k_{2_{pos}} > 0$
- IF x_3 is *positive* _{x_3} AND x_4 is *negative* _{x_4} , THEN u is $u = -k_{1_{pos}}x_3 - k_{2_{neg}}x_4$ where $k_{1_{pos}} \leq 1, k_{2_{neg}} > 0$

But here, before that I will derive again another kind of rule base for the same system. The u instead of earlier case u was simply a linguistic variable, but now I am assuming this is a state variable technique, where u is minus k_1x_3 minus k_2x_4 . Of course this is just to simplify but actually I should have taken all the x_1x_2 I am not taking, just to simplify because if I am able to control using only two variables, I can only expect that using x_1 and x_2 I can improve the performance. So putting expression of u in $\dot{V}$ is already we have found out this earlier because we considering same RTAC system.

So for RTAC system giving the physical principles $\dot{V}$ qualitatively comes out to be x_3x_4 plus x_4u . Now this $\dot{V}$ can be written as x_3x_4 and then x_4u is this quantity which is minus k_1x_3 minus k_2x_4 , and this can be again simplified, this is $1 - k_1$ into x_3x_4 minus k_2x_4 whole square. So how do I now derive the rule base for this kind of structure?

I just have to make sure that I have to select k_1 and k_2 in such a way that this quantity is negative definite. This is what, if x_3 is negative and x_4 is negative then u is, you see that x_3 is negative and x_4 is negative, so this quantity is positive. Since x_4 is negative, the x_4 square is positive. All that I have to make sure that $1 - k_1$ has to be negative as well as k_2 is also a positive. So k_2 is positive and $1 - k_1$, because this is positive, this has to be negative, $1 - k_1$ is negative means k_1 is greater than 1. This is my first rule.

The second rule is if x_3 is positive and x_4 is positive, this quantity is positive, this quantity is positive, to make it again negative definite, again I will have the same principle whatever I got here I must be that, k_{2pp} greater than 0, k_{1pp} is greater than equal to 1. Same condition, only this nn stands for the rule base when x_3 is negative and x_4 is negative and here pp stands for x_3 is positive and x_4 is positive.

In the second rule you saw that, the same condition came but these are different parameters you have to remember k_{1pp} . If x_3 is positive and x_4 is positive then u is equal to minus $k_{1pp} x_3$ minus $k_{2pp} x_4$ where I showed you that this has to be positive and k_2 has to be positive.

Now let us go to the third one, x_3 is negative AND x_4 is positive THEN u is minus $k_{1np} x_3$ minus $k_{2np} x_4$. How do I find out what is k_1 and k_2 ? It is simple; k_{1np} you see that, x_3 is negative and x_4 is positive here, that makes this is negative. This is negative means this has to be positive, 1 minus k_1 has to be positive, so k_1 should be less than 1. That is k_{1np} is less than 1 and here this has to be again positive quantity $k_2 x_4$ because this is negative is already there, this x_4 is positive so this is always positive whether x_4 is negative or positive. To make it positive k_2 also has to be positive, so this is what k_{2np} this has to be positive.

Similarly, if x_3 is positive and x_4 is negative, again this is negative, this has to be again positive, and again k_1 should be less than 1. So k_{1pn} is less than 1 and k_{2pn} has to be again here this is positive, so k_2 has to be always positive. You can easily see that this k_2 always in every case is positive, only in this case, the first one it is positive and the second that is negative. What you got an idea that, what should be this value k_1 and k_2 , at least the magnitude?

Now **heuristically** you have to tune these values what should be k_{1nn} k_{2nn} k_{1pp} k_{2pp} k_{1np} k_{2np} and k_{1pn} and k_{2pn} heuristically, or I can use the genetic algorithm or we have done that univariate marginal distribution algorithm to optimize this parameter.

You see that considering the same membership function and using product for AND inference and center of gravity method for defuzzification, the fuzzy controller is given is

like this. Since μ_p is this and μ_n is this, which we have already done in the previous case, we can write u as like this, we have a controller and by simplifying that we have actually ten parameters to identify.

(Refer Slide Time 41:48)

Fuzzy control law

After simplification, the fuzzy control law becomes:

$$u = \frac{(-k_{1_{pos}}x_2 - k_{2_{pos}}x_1)e^{-(2\alpha_{x_2} + \alpha_{x_1})x_3} + (-k_{1_{neg}}x_2 - k_{2_{neg}}x_1)e^{(2\alpha_{x_2} + \alpha_{x_1})x_3}}{(e^{-2\alpha_{x_2}x_3} + e^{2\alpha_{x_2}x_3})(e^{-\alpha_{x_1}x_3} + e^{\alpha_{x_1}x_3})}$$

$$= \frac{(-k_{1_{pos}}x_2 - k_{2_{pos}}x_1)(e^{-2\alpha_{x_2}x_3} - 2\alpha_{x_2}x_3 + \alpha_{x_2}^2x_3^2) + (-k_{1_{neg}}x_2 - k_{2_{neg}}x_1)(e^{2\alpha_{x_2}x_3} - 2\alpha_{x_2}x_3 + \alpha_{x_2}^2x_3^2)}{(e^{-2\alpha_{x_2}x_3} + e^{2\alpha_{x_2}x_3})(e^{-\alpha_{x_1}x_3} + e^{\alpha_{x_1}x_3})}$$

The optimal parameters are found as:

$$\alpha_{x_2} = 0.25, \alpha_{x_1} = 0.12,$$

$$k_{1_{pos}} = 1.2, k_{2_{pos}} = 0.35,$$

$$k_{1_{neg}} = 1.4, k_{2_{neg}} = 0.2,$$

$$k_{1_{pos}} = 0.6, k_{2_{neg}} = 0.4,$$

$$k_{x_1} = 0.8, k_{x_2} = 0.25$$

Two parameters x_3x_4 , the mean for linguistic variable x_3 positive, for x_3 positive and x_3 negative the mean is x_3 and actually in case of one case mean is positive and in one case mean is negative. That is what we are doing that when it is negative, mean is negative, so the sign changes that is why you saw that here.

(Refer Slide Time 42:30)

Inference Mechanism and Defuzzification

Considering the same membership functions and using "product for AND" inference and "center of gravity" method for defuzzification the fuzzy controller is given as:

$$\mu = \frac{\mu_p(x_1)\mu_n(x_2)(-k_{1_{\text{low}}}x_3 - k_{2_{\text{low}}}x_4) + \mu_p(x_1)\mu_p(x_2)(-k_{1_{\text{up}}}x_3 - k_{2_{\text{up}}}x_4)}{\mu_p(x_1)\mu_n(x_2) + \mu_p(x_1)\mu_p(x_2) + \mu_n(x_1)\mu_p(x_2) + \mu_n(x_1)\mu_n(x_2)}$$

$$= \frac{\mu_p(x_1)\mu_p(x_2)(-k_{1_{\text{up}}}x_3 - k_{2_{\text{up}}}x_4) + \mu_n(x_1)\mu_n(x_2)(-k_{1_{\text{low}}}x_3 - k_{2_{\text{low}}}x_4)}{\mu_p(x_1)\mu_p(x_2) + \mu_n(x_1)\mu_p(x_2) + \mu_p(x_1)\mu_n(x_2) + \mu_n(x_1)\mu_n(x_2)}$$

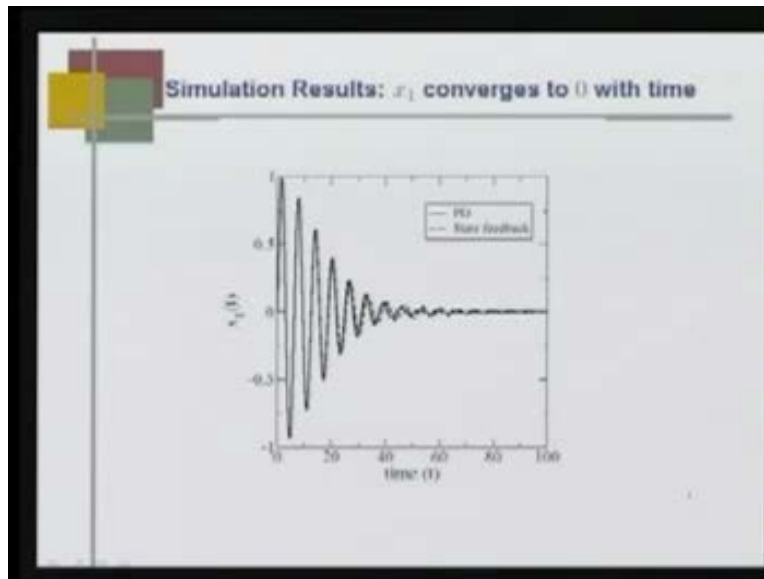
Since $\mu_p(x) = e^{-(x-a_p)^2}$ and $\mu_n(x) = e^{-(x+a_n)^2}$, we can write

$$\mu = \frac{e^{-(x_1-a_p)^2}e^{-(x_2-a_p)^2}(-k_{1_{\text{up}}}x_3 - k_{2_{\text{up}}}x_4) + e^{-(x_1+a_n)^2}e^{-(x_2+a_n)^2}(-k_{1_{\text{low}}}x_3 - k_{2_{\text{low}}}x_4)}{e^{-(x_1-a_p)^2}e^{-(x_2-a_p)^2} + e^{-(x_1-a_p)^2}e^{-(x_2+a_n)^2} + e^{-(x_1+a_n)^2}e^{-(x_2-a_p)^2} + e^{-(x_1+a_n)^2}e^{-(x_2+a_n)^2}}$$

$$= \frac{e^{-(x_1-a_p)^2}e^{-(x_2-a_p)^2}(-k_{1_{\text{up}}}x_3 - k_{2_{\text{up}}}x_4) + e^{-(x_1+a_n)^2}e^{-(x_2+a_n)^2}(-k_{1_{\text{low}}}x_3 - k_{2_{\text{low}}}x_4)}{e^{-(x_1-a_p)^2}e^{-(x_2-a_p)^2} + e^{-(x_1-a_p)^2}e^{-(x_2+a_n)^2} + e^{-(x_1+a_n)^2}e^{-(x_2-a_p)^2} + e^{-(x_1+a_n)^2}e^{-(x_2+a_n)^2}}$$

In this case this is positive x when it is negative mean is negative, so it is positive quantity and mean is positive here in case of positive, so it is negative here. So ax_3 is 0.25 here, doing optimization using univariate marginal distribution ax_3 is 0.25 and ax_4 is 0.12 and the parameters, the controller parameters that is k_1 and k_2 for all the different fuzzy rules, four rules are there and for each rule we have two parameters k_1 and k_2 and we have got these values 1.1, 1.4, 0.6, 0.8, 0.35, 0.2, 0.4, 0.25. These parameters we derived using univariate marginal distribution algorithm.

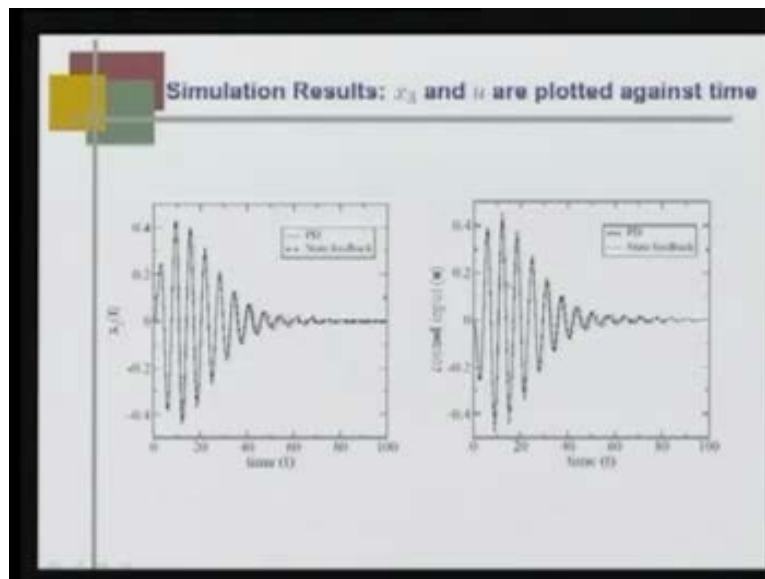
(Refer Slide Time 43:29)



You see that simulation results x_1 converges to 0 with time, what we started with we gave some initial disturbance and using the initial disturbance we are showing now, here x_1 , x_1 is the position of that cart, you give some little disturbance to the cart finally it should come to the rest position.

What is the meaning of PD? This is the FLC controller where we have used the PD type of rule base that is if x_3 is positive and x_4 is positive then u is negative, in that kind of rule base. This is PD; it is actually the rule base that we formulated. That is, we are denoting we are saying this is PD and this we are saying state feedback because you see that u is a function of states. By doing that you see that, we have got the response that, this is in second unit, after 40 seconds almost the oscillation comes to an end.

(Refer Slide Time 44:58)



And here x_3 is plotted which is the angular position, that also should come to, because the cart was disturbed, although the pendulum was in the beginning initial position, but it will start oscillating and that oscillation will gradually slow down and it equals in same amount of time this is reduced to 0, both the PD type of rule base and state feedback type of rule base and this is the control action and this is control action you see that the control action is also fairly smooth both for PD and as well as state feedback.

(Refer Slide Time 45:50)

Tracking Controller

The control objective is to find a u such that the actuator angle will follow any given desired trajectory.

Let $V = \frac{1}{2}(e_1^2 + e_2^2 + e_3^2 + e_4^2)$ be the Lyapunov function candidate, where $e_i = x_i^d - x_i$, x_i is i^{th} state and x_i^d is the corresponding desired response. Since $e_i = x_i^d - x_i = x_{i+1}^d - x_{i+1} = e_{i+1}$ for $i = 1, 3$, we can write,

$$\dot{V} = e_1 e_2 + e_2 e_3 + e_3 e_4 + e_4 e_1$$

Again $\dot{x}_2 \approx -x_1$ and $\dot{x}_4 \approx u$. Thus one can simplify the above expression to:

$$\dot{V} \approx e_3 e_1 + e_4 e_1$$

where $v = \dot{x}_1^d - u$.

Saying that what we did is the regulation that is given the RTAC system we gave some disturbance and we showed that this controller is able to stabilize the system. Now the tracking controller that we would like that our pendulum should track it as a trajectory. For tracking a desired trajectory, what we would like to have is that instead of the Lyapunov function which we had half x_1 square plus x_2 square plus x_3 square plus x_4 square for regulation problem, now we will consider e_1 square plus e_2 square plus e_3 square plus e_4 square be the Lyapunov function where e_r is desired state trajectory minus the actual state trajectory. So x_i is the higher state and x_{id} is the corresponding desired state trajectory.

The control objective is to find a control action such that the actuator angle will follow any given desired trajectory. Hence the Lyapunov function has been changed from x_1 square plus x_2 square plus x_3 square plus x_4 square to in this case e_1 square plus e_2 square plus e_3 square plus e_4 square. I hope this is clear to you.

Now in the same manner we can also show $\dot{e}_i$ is $\dot{x}_i$ minus $\dot{x}_{id}$ which is same as $\dot{x}_i$ plus 1 d minus $\dot{x}_{id}$ plus 1 which is $\dot{e}_i$ plus 1. What is $\dot{x}_{id}$ plus 1? Because we have the parameters x_1, x_2, x_3, x_4 , so when I say $\dot{x}_{id}$ plus d, when I say $\dot{x}_i$ is this is $\dot{x}_1$, $\dot{x}_1$ dot d is same as $\dot{x}_2$ d. Obviously, because x_2 and x_1 the relation is $\dot{x}_2$ is $\dot{x}_1$ dot. By that principle you are able to see that $\dot{x}_i$ dot d is $\dot{x}_i$ plus 1 d which I am writing here and similarly also I can write $\dot{x}_3$ dot d is $\dot{x}_4$ d, $\dot{x}_3$ because $\dot{x}_3$ is $\dot{x}_4$. So obviously $\dot{x}_3$ dot d is desired is $\dot{x}_4$ desired.

(Refer Slide Time 48:44)

Tracking Controller

The control objective is to find a u such that the actuator angle will follow any given desired trajectory.

Let $V = \frac{1}{2}(e_1^2 + e_2^2 + e_3^2 + e_4^2)$ be the Lyapunov function candidate, where $e_i = x_i^d - x_i$, x_i is i^{th} state and x_i^d is the corresponding desired response. Since $e_i = x_i^d - x_i = x_{i+1}^d - x_{i+1} = e_{i+1}$ for $i = 1, 3$, we can write,

$$\dot{V} = e_1 \dot{e}_2 + e_2 \dot{e}_2 + e_3 \dot{e}_4 + e_4 \dot{e}_4$$

Again $\dot{x}_2 \approx -x_1$ and $\dot{x}_4 \approx u$. Thus one can simplify the above expression to:

$$\dot{V} \approx e_3 e_4 + e_4 \dot{V}$$

where $v = \dot{x}_4^d - u$

Handwritten notes on the right side of the slide:

$$\begin{aligned} \dot{x}_2 &= -x_1 & \dot{x}_3 &= \dot{x}_4 \\ \dot{x}_4 &= u & \dot{x}_1 &= \dot{x}_2 \end{aligned}$$

Handwritten box containing:

$$\dot{x}_4^d = \dot{x}_4^d$$

For i equal to 1 and 3, that is what I have done, i equal to 1 and 3 and differentiating V we had $\dot{V}$ is $e_1 \dot{e}_2$ plus actually this is $e_1 \dot{e}_1$ dot $e_2 \dot{e}_2$ dot $e_3 \dot{e}_3$ dot $e_4 \dot{e}_4$ dot. We already know what is $\dot{e}_2$ is $\dot{e}_1$ dot and this is retained and $\dot{e}_4$ is $\dot{e}_3$ dot. This is our actual expression of $\dot{e}_1$ dot.

Just like this again $\dot{x}_2$ dot is approximately minus x_1 and $\dot{x}_4$ dot is approximately u , thus in the same similar logic we can simplify this $\dot{V}$ dot to be qualitatively $e_3 e_4$ plus because qualitatively this will cancel out this because $\dot{e}_2$ dot would be minus of e_1 . This will be cancelling out qualitatively, what is remained is $e_3 e_4$ plus e_4 and $\dot{e}_4$ dot we see that $\dot{x}_4$ dot is directly proportional to u . We can write that this is $\dot{e}_4$ and $\dot{e}_4$ dot is V where V is $\dot{x}_4$ dot minus u .

This is very simple because $\dot{e}_4$ dot is $\dot{x}_4$ dot desired minus $\dot{x}_4$ dot and since this is $\dot{x}_4$ dot I put this $\dot{x}_4$ dot to be u , I can replace that by u and $\dot{x}_4$ dot minus u I can write V and that V is $\dot{x}_4$ dot minus u .

What we have to do? I can create a rule base from this Lyapunov function assuming V to be my control input and from V if I know V then I know u , actual control action, this is pseudo control action, my actual control action would be $\dot{x}_4$ dot minus u .

(Refer Slide Time 51:33)

Tracking Controller

- Regulation : $\dot{V} \approx x_3x_4 + x_4u$
- Tracking : $\dot{V} \approx e_3e_4 + e_4v$

Thus v can be designed in the same way as that of u in regulation problem.

- Tracking Controller - PD : $u = \dot{x}_4^d - v$ where

$$v = \frac{(k_{e3})[e^{-(2\alpha_3+3+2\alpha_4)s}] + k_{e4}(2\alpha_3+3+2\alpha_4+4)]}{(e^{-2\alpha_3+3} + \alpha_3 e^{2\alpha_3+3})[(e^{-2\alpha_4+3} + \alpha_4 e^{2\alpha_4+3})]}$$
- Tracking Controller - state feedback : $u = \dot{x}_4^d - v$ where

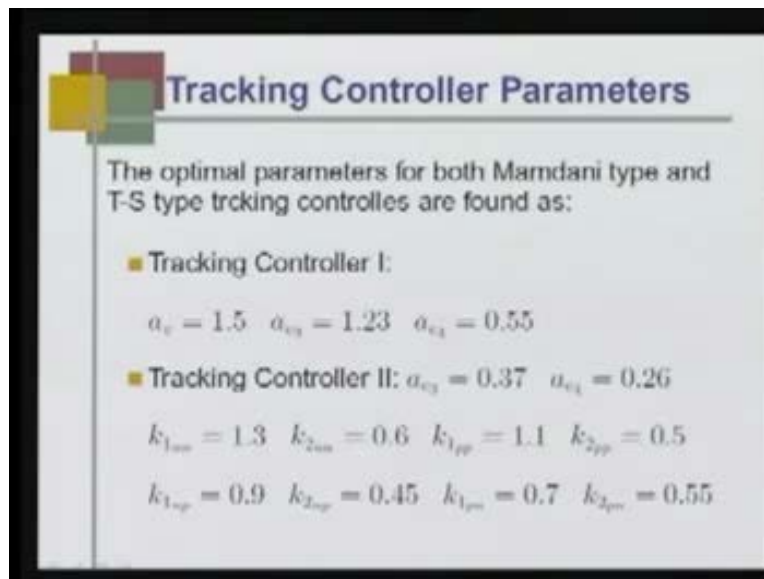
$$v = \frac{(-k_{1,reg}e_3 - k_{2,reg}e_4)[e^{-(2\alpha_3+3+2\alpha_4)s}] + (-k_{1,reg}e_3 - k_{2,reg}e_4)[e^{-(2\alpha_3+3+2\alpha_4)s}]}{(e^{-2\alpha_3+3} + \alpha_3 e^{2\alpha_3+3})(e^{-2\alpha_4+3} + \alpha_4 e^{2\alpha_4+3})} + \frac{(-k_{1,reg}e_3 - k_{2,reg}e_4)[e^{-(2\alpha_3+3+2\alpha_4)s}] + (-k_{1,reg}e_3 - k_{2,reg}e_4)[e^{-(2\alpha_3+3+2\alpha_4)s}]}{(e^{-2\alpha_3+3} + \alpha_3 e^{2\alpha_3+3})(e^{-2\alpha_4+3} + \alpha_4 e^{2\alpha_4+3})}$$

In general what we learnt is that for regulation the qualitative equation for right derivative of Lyapunov function x_3x_4 plus x_4u for tracking you found out this to be $x_4\dot{V}$ is qualitatively equal to e_3e_4 plus e_4 into V where V is $\dot{x}_4$ minus $\dot{x}_4^d$ minus u and u is the actual control action and V is the actual control action.

This V can be designed the same way as that u in the regulation problem; I will not be going in detail because this and this has the same structure. The rule base either using the PD type of controller or the state feedback controller the two rule base will have same form, only thing that I am computing V . After I compute V , V will have the same structure that we earlier derived the same way but given V , I find out u to be $\dot{x}_4^d$ minus V .

Doing that and also the parameters in the first case will have three parameters and in the second case similarly ten parameters that have to be optimized.

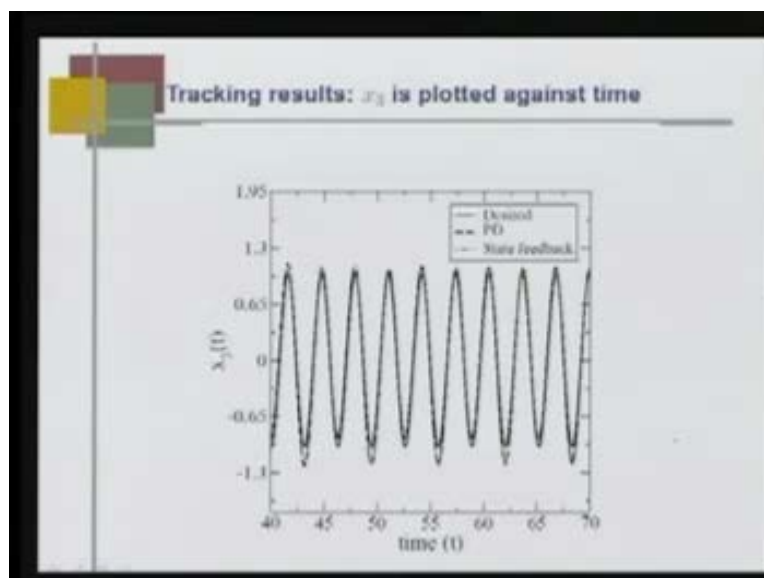
(Refer Slide Time 53:11)



By optimizing these parameters, we get this is a_v not a_u , a_v is 1.5, a_{e3} 1.23, a_{e4} is 0.55 and for this tracking controller two you have these parameters .37, .26 and these are the parameters for the state feedback controller for four rules.

So k_{1nn} is 1.3 and k_{1np} is .9, k_{2nn} is .6, k_{2np} is .45 and so on.

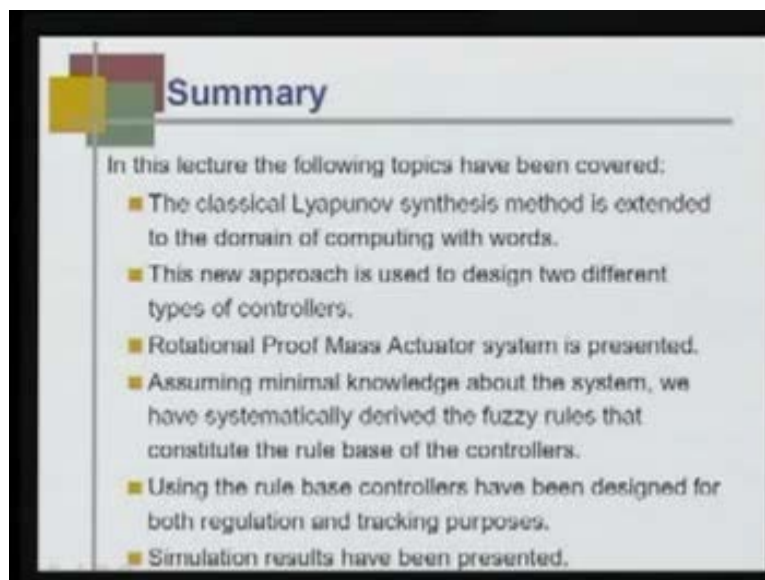
(Refer Slide Time 53:52)



Given that, the tracking result is x_3 is plotted against time desired PD and state feedback give the same result here for the tracking. Tracking is not plotted because you see that the rule base is very small that we have taken and also we have in this state feedback control are also we have rejected in the state feedback the input of x_1 and x_3 the state feedback for x_1 and x_2 .

The tracking here is not right, some loss here otherwise tracking is properly and the control action you can see both PD and state feedback they behave almost similarly.

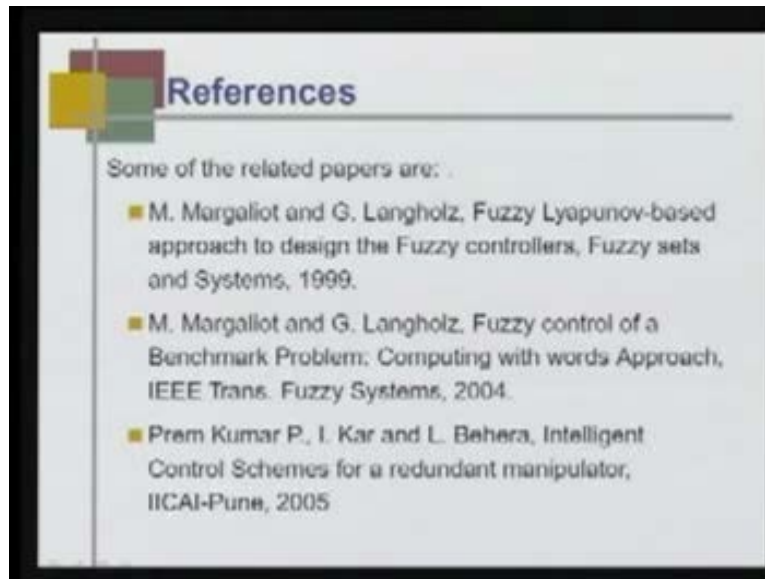
(Refer Slide Time 54:48)



Finally the summary, in this lecture following topics have been covered a classical Lyapunov synthesis method is extended to the domain of computing with words that means qualitatively how can we use a rate derivative of the Lyapunov function in such a way that the rule base can be generated. This new approach is used to design two different types of controller actually, the two different types of rule base, controller is same fuzzy logic controller mamdani type.

Rotational proof mass actuator system is presented for simulation. Assuming minimal knowledge about the system we have systematically derived fuzzy rules that constitute the rule base the controller. Using rule base controller we have design for both regulation tracking purposes. Simulation results are presented.

(Refer Slide Time 55:44)



References in this case are some of the related papers are you can easily see Fuzzy Lyapunov based approach to design the fuzzy controllers, Fuzzy sets and system in 1999 by Margaliot and Langholz and same author also have a paper on fuzzy control of a Benchmark problem; computing with words approach, IEEE transaction Fuzzy Systems, 2004.

And also we have a paper some work on the directional intelligent controls schemes for a redundant manipulator presented in a conference held in Pune in 2005.

Thank you very much.