Newton's Law - Inspired Tuning of PI Controllers - 2

Welcome back. In the previous class, I began with the underwater vehicle, and we solved the stabilization problem. And what was our observation? That in the presence of the non-vanishing uncertainty, a state which we have defined as a velocity that is not insensitive to the disturbance. So, what is the meaning of non-vanishing disturbance? Those classes of disturbance which are non-zero at the equilibrium point. So, I have to explore some different methodologies to solve that class of problems such that in the presence of non-vanishing disturbance, I can force $v = 0$, irrespective of any bounded disturbance or uncertainty. That is our goal.

And how do we solve that problem? So, in order to solve that problem, we have to incorporate more information. So, in the previous class, all control methodology is just based on the information of the present. In this lecture, I am also going to incorporate information from the past, and that kind of controller is called proportional integral control; 70 percent of industrial plants are based on the PI controller. So, in this lecture, we are going to look at why the industry relies so much on the PI controller and what its limitations are.

Let us begin today's lecture. So, the purpose of the discussion is to understand the tuning of the PI-based controller, and again I am going to utilize the same framework. I have already told you that most of the concepts you can understand using Newton's law. After that, I am also going to explore the limitations of the PI controller. In other words, there is no control; you can talk about PID controllers, adaptive controllers, sliding mode controllers, or optimal controllers.

So, every control has some form of limitations. And due to that, it is better to learn or explore different forms of the controller. So, we are also going to learn the different variants of the PI controller. So, what is our outcome? Now, we can able to give guarantee that if I have some kind of system, then how to design control, what kind of disturbance they are going to tackle. And after that, we are also going to understand the variant of the PI controller.

So, I am going to start with exactly the same plant. I have already told you that we are going to start from the same system, which is the underwater vehicle, and I have already designed the feedforward control. So, our system looks like this. So, here is the first order system, and I have told you that understanding the first order system is very easy because I can represent the first order system using this kind of real line. It means that whatever velocity comes from this particular differential equation means the solution of this equation that is going to lie somewhere on this straight line.

And we have actually updated our control. So, this is the present information, and this information is nothing but the information of the past. And how do we represent past information? So, using the integral. Now, I am assuming that $d$is non-vanishing, but it is a constant. So, if $d \neq 0$and if that is constant, one can assume that its derivative is equal to 0.

Another assumption people can make is that d is actually non-vanishing, but not constant. So, after that, after substituting control, this is the closed loop system. And you can see here that this differential equation is no longer a differential equation. Now, this is a differential integral equation. And whenever integral term comes into picture, it is possible to show that order of the system is going to increase by 1.

So, how do you show that? I have actually defined this as some fictitious state, that is z, and I am assuming that $d$is differentiable. If d is not differentiable, then a different treatment is required. So, in this particular lecture, I am assuming that d is differentiable. So, I can able to represent system like this. Now, you can see that the first-order system is converted into a second-order system, and what is the physical interpretation of a second-order system? Now, I have two axes.

So, this axis is already v. So, I have one extra axis, z, and suppose I have some kind of initial condition here. So, now our situation is little bit difficult. In the one-dimensional case, everything is very easy. I can move either in this direction, that direction, or stay here, but in two dimensions, that is not the case.

I have infinitely many choices if I start here. Now you can see that I have two different cases. I am assuming that disturbance, whatever differentiable disturbance it is, is constant. Even if piecewise constant, the assumption is also okay. So, if you differentiate that, so even if that constant disturbance is not known to So, now that will not present.

It means that in transform frame, our system is fully independent or fully insensitive with respect to the constant disturbance, even if that constant disturbance is unknown to us. So, due to that reason, we can comment that PI controller, if some kind of constant disturbance will act on the system. So, the PI controller is insensitive with respect to that particular class of disturbance. But now, tuning this gain k_p and k_i is necessary because we are no longer in one dimension. So, one-dimensional tuning was very easy.

What is our basic philosophy? Our basic philosophy is very simple. We are assuming that $v$is here; we are calculating $\dot{v}$, and in order to move in this direction, we are assuming that $v$is positive; then $\dot{v}$ is negative. If v is negative, then $\dot{v}$ is positive; however, that is not the case here. Now, suppose you have a time-varying disturbance, not a constant one; then what happens? Now, our system, first order system is converted into second order on certain system. So, the design of the gains is another challenge, but what kind of insight will we get from here that the PI controller is insensitive with respect to any constant non-vanishing disturbance? That kind of message is basically what this slide is going to give you.

Now, I have simulated So, I have taken the same system like this, and the constant disturbance I have kept here, and you can see that our initial condition is here. So, I have perfectly converged to the equilibrium point. It means that if I apply PI controller, our system is become insensitive after some time, and you can see behavior from here. This disturbance, you can see that this is a differentiable disturbance, but that is not equal to 0. So, you can see the behavior.

It means that the PI controller is not suitable for the time-varying disturbance. And those who are working towards the practical problem, you might have observed that if you have a constant offset, then using a PI controller, you can compensate for it. So, why are you able to do that? So, this is actually the explanation of that particular phenomenon. Now, let us come to the gain design phase. So, again I am going to apply Newton's law to design the gain.

So, how do we do that? Now, since we are in two-dimensional space, I have already told you that I have several ways to evolve from here. So, for that reason, what am I going to do? I am going to introduce one transformation, and whenever you are going to introduce some kind of transformation, you have to be a little bit careful because, using transformation, I am moving from the $z$and v coordinate frame to the x_1 and x_2 coordinate frame. So, after analysis, you have to again return back. It means that whatever transformation you are going to utilize, that should be invertible. It means that there should be a one-to-one mapping.

In this particular place, I am assuming that this transformation T, whatever its entry, is nothing but some kind of matrix, because here I have R and this is, so I have 2×2. So, here I have defined a transformation that is 2×2 and this coordinate system is 2×2. So, you can see that the dimension will match and what our objective is. Our objective to decouple the system. It means that after doing this transformation, either the initial condition will lie on this line or this line, and our common intersection point is this; that is our objective.

So, because I have to move here, this point is exactly mapped to this particular point. So, once I am able to do this, you can see that I can apply Newton's law in this axis as well as in this axis. So, how do you do that? It is possible to show that some systems look a little bit complicated in their original coordinate frame, but if you apply some concepts like eigenvalues and eigenvectors, particularly for the case of linear systems, then you will get a fully decoupled system. And here, both k_p and k_i are in our hands. I am able to design the gains of the proportional as well as integral controller.

And due to that reason, you can always be able to tune some kind of matrix Asuch that its eigenvalue is always strictly negative and different. Different means since I have a 2×2 matrix, so I have two eigenvalues. So, first, the eigenvalues λ_1 and $\lambda_2$are supposed to be different, I assume. Now, you can see here that I have applied transformations like $x = T\eta$; $T$is a 2×2matrix. And if I take the derivative, since I am assuming that each entry of this transformation is somehow constant, due to that reason, if you take the derivative, then the derivative will come here, and after that, I am going to substitute the value of $\dot{\eta}$.

So, $\dot{\eta}$is nothing but η. I hope that you are able to understand how one can represent this system like this. So, this representation is also called a state space representation, and this is an autonomous system. Now, what am I going to do? η, since I know that $x = 2\eta$.

So, $\eta = TAT^{-1}x$. Now, our main aim is to select this T. So, how do we select this T? You can see that applying the transformation, eigenvalue of the matrix is not going to change, and that idea I am going to use. So, I have substituted the eigenvalues in the diagonal position, and now I know that $\lambda_1$and λ_2 are the eigenvalues of A. Therefore, I know Ameans that I will keep the $k_p$and k_i values and try to find those kinds of matrices $A$by substituting the $k_p$and $k_i$values such that $\lambda_1$and $\lambda_2$are both strictly negative. Why? You can see here that once you substitute TAT^{-1} in the previous equation, now, in the transformed coordinate frame, both equations are fully decoupled.

So, now, our system is $\dot{x}_1 = \lambda_1 x_1$. So, you can apply Newton's law. If you start from here, if $x_1$is positive, and if $\lambda_1$is negative, then you can only move in this direction. So, in this way, analytically, you can able to tune the PI controller. Let us consider the case where you have a time-varying disturbance that is not constant.

So, in that situation, $\dot{d}$also comes into the picture, and in this case, you can see that if I apply the same transformation, exactly the same transformation, then what happens? I have TAT^{-1}; after that, I have $\dot{x}$, which is $x_1$and x_2, and after that, I have nothing but some kind of matrix like TB. So, $TB$is represented by $b_1$and b_2. So, now this system, if you design $\lambda_1$and $\lambda_2$again same like the previous methodology, you can see here that this system is affected by the disturbance. Now, you can here apply the concept of uniform and ultimate bounds, whatever we learned in the previous class. So, what is our conclusion? If you have a non-vanishing constant disturbance, a PI controller is fully able to remove that kind of disturbance, meaning the behavior of the whole system is insensitive with respect to some kind of time-varying constant disturbance.

But if some kind of time-varying disturbance comes into the picture, then we will get some behavior like uniform ultimate bounded behavior. So, the PI controller is unable to compensate, or the PI controller is not insensitive with respect to the time-varying disturbance. That was the conclusion. Now, it is good to look into the variants of the PI controller. So, this is the very popular variant we are going to learn in this particular course.

So, I am describing this as a non-linear PI controller. Why am I giving this example? By using this example, you can understand that information, whether present or past, is sometimes something I can capture exactly, and after that, I will scale it or sometimes you can represent it in a different way, in some scaling manner. So, suppose that if I have information about x, then I have scaled information like this. Similarly, I have scaled information like this in the case of. So, what kind of example are we looking for? Suppose that I have a system like $\dot{x} = u + d(t)$.

So, similar to the particular problem in the case of the underwater vehicle, that is exactly equivalent to this. So, there is no difference between $x$and v. Both $x$and $v$belong to the real numbers, which means one dimension. After that, now you can substitute the control, then your dynamics looks like this. Here again I am assuming that because we are already well aware that if this disturbance is differentiable, then I can do the coordinate transformation.

If this is not differentiable, then that disturbance start acting here and that will create some kind of difficulty. We will see this kind of case when I deal with the sliding mode control. So, here you can see that now whatever system is similar to the classical PI controller, I have the exact same system, but now this information and this information are a little bit different. So, how do you design $k_1$and k_2? So, again here you can see that this control is now able to be insensitive with respect to any time-varying disturbance also. So, those kinds of things I am going to establish.

It means that classical PI is only able to take care of the differentiable constant disturbance, but this controller is also robust with respect to or insensitive to some kind of differentiable constant or non-constant disturbance. And in the later part of this course, we are also going to see that $x = 0$ and $z = 0$ both infinitely; that is our extra advantage. So, the designing methodology is exactly the same. So, again our observation is exactly the same: if d is constant, then our system is fully sensitive with respect to that perturbation. But if $d$is time-varying, then this kind of closed-loop dynamics we will get.

And again using the same Newton's law, I am going to tune the controller. So, what is our methodology? The methodology is exactly the same. Here you can see that this controller in sliding mode control is also called a super twisting controller. So, that part you are going to explore whenever I teach you the higher order sliding mode control. So, I am just going to give you the feeling of how sliding mode control is just a generalization of the classical PI controller.

For that reason, I have selected this example. You can see that here I have used this kind of representation. What is the meaning of this kind of representation? So, meaning of this kind of representation that we are going to consider all possible value of $\dot{d}$. Since $d(t)$is not known, we only know the bound, which means that $\dot{d}(t)$can take any value between $d_0$and $-d_0$, and if we incorporate each and every condition, this representation comes into the picture. So, this is called a differential inclusion.

Differential inclusion. And what is the meaning of differential inclusion? You can see that in this particular representation, I have infinitely many differential equations here. Now I am going to tune the gain. So, in order to tune the gain, what is our basic philosophy? Either we try to decouple the system, or at least we are trying to partially decouple it. A first-order system is comfortable for us because I can easily apply Newton's philosophy. So I am going to apply again, the same way, exactly the same way I am going to apply this transformation, and at that time you can see that only I will get the partial decoupling.

This term is present, but this term is not going to harm us. Why? Because this term is always positive. Okay, but I have to remove $x = 0$because $x^{-1/2}$comes into the picture. So, at $x = 0$, that is not defined. Okay, but that point is completely fine because if $x = 0$, $z = 0$.

Then we are already at the equilibrium point. So, that point is not going to disturb us, and for that reason, we are able to remove that point. But the only difficulty is that if $x = 0$, then $z \neq 0$. So, that is the kind of difficulty we have. So, let us try to see how to tackle this case. So, in this particular case, one more important thing is that this $\nu$is nothing but the x, you can assume.

Now, in the exact same way, I am going to proceed, but you can see here that the coordinate transformation first considers the coordinate system $x^{1/2}$ sign(x). So, now if you differentiate, then the differentiation of $x^{1/2}$ sign(x)is nothing but $x^{-1/2}\, \dot{x}/2$. Please do it by yourself, and after that, you can see that the exact same kind of linear system with this kind of non-linearity comes into the picture. And if you apply the transformation, the transformation is exactly the same, and this is just a scaling term.

So, forget about this for time being. And after that, you can again see that if I maintain the eigenvalue of Astrictly negative, then I will get this kind of partially decoupled system. And now you can see, I am able to write this as $x^{-1/2}$and $\lambda \times 1$. And if I tune $\lambda < 0$, then what happens? You can just come here, and after that, if you are here, you will converge in this way. So, in this way you can tune the non-linear PI controller.

using Newton's law. Now, I haven't analyzed when $x = 0$, but $z \neq 0$. So, try to observe the behavior. So, it is possible to show that if $x = 0$, but $z \neq 0$, it means that if $x = 0$and $z \neq 0$, I am on this axis, anywhere on this axis, but you can see this is called the vector field. So, $\dot{z}$is some kind of right-hand side, which is some kind of constant quantity $-k_2 + d_0$in the worst case. So, that is not going to stay here, and finally, they will converge here.

It means that you just have to take care of the case when $x \neq 0$, and whenever $x = 0$, then the trajectory will automatically converge here. So, that point is not going to harm our analysis. In the worst case, if there is a disturbance, and if the disturbance is time-varying, then you will be able to design the matrix Aor the eigenvalue of the matrix based on this particular analysis. And I have already told you that whenever you move from a classical PI controller to a super twisting or non-linear PI controller, you have an extra advantage.

What kind of extra advantage you have? that both state variable. So, I have two state variables in this particular representation; you can see here that $x = 0$and $z = 0$both in the infinite term. And if I am claiming that $x$and z, and equivalently $\dot{x}$and $\dot{z}$, have equivalent values of $\dot{z} = 0$, it means that this guy should perfectly track $\dot{d}$. So, somehow this particular algorithm is also give you the information of the disturbance. So, those kinds of things we have actually represented here. You can see that k sign(x)means the equivalent value of this is exactly equal to $\dot{d}(t)$; then only can I claim that $z = 0$.

Now, it is good to simulate the first order system by taking this kind of control. So, this control has a name known as smooth PI. So, if you are working with an industrial plant, then you cannot apply some kind of complicated term here, because whenever you are going to do implementation using some kind of macro processor, then $x^{1/2}$is a little bit difficult to implement, but this controller you can easily implement. How do you tune the gain? That you are able to learn from the internet.

So, we have already published one paper in this area. So, it is time to conclude today's session. So, what we have seen is that the PI controller is very, very famous. So, using Newton's law again, you can be able to tune that. And what is philosophy? You should convert a second-order system into a decoupled or partially decoupled first-order system.

With the help of the notion of eigenvalues and eigenvectors. I hope that you are already aware of the eigenvalue and eigenvector. We have also explored some other variants. Why are we basically worrying about the other variant? Because most of the practical problems are affected by the time-varying uncertainty of disturbances, we have to take care of that. So, what is our next step now? Our next step is to explore something more because of what we have actually observed in the case of the PI controller. If the disturbance is not differentiable or time-varying, then classical PI is not able to take care of it.

If it is differentiable and bounded, then non-linear PI is able to take care of that, but suppose the disturbance is not differentiable. So, how do you do that? So, those kinds of things we are going to look into in the subsequent slide, and I have already told you that the PI controller and its variant are quite famous in the industry. If you are working with any practical system using this kind of philosophy, Newton-based philosophy, please try to design the control for your problem. So, thank you.