Sliding Mode Control and Applications

Dr. Shyam Kamal
Department of Electrical Engineering
IIT(BHU) Varanasi

Week-04

Lecture-18

Welcome back. In the previous class, I was talking about the integral sliding mode control, and I had just taken the disturbances that enter through the control channel. We have already defined that the class of disturbances is called matched uncertainty or disturbances, and what our observations were is that. our trajectory is insensitive with respect to any matched uncertainty. Just a few assumptions we have that should be bounded. But if you see the realistic situation, at that time you will be able to see that several classes of uncertainty will not enter through the control channel.

And due to that reason, we have to extend integral sliding mode control for the unmatched uncertainty as well. It means that those classes of uncertainty which will not enter through the control channel, and then I am going to redesign the sliding mode control such that at least unmatched uncertainty will be minimized. Here, I am going to clarify that, based on my knowledge, it is not possible to fully compensate for any classes of unmatched uncertainty. Obviously, if unmatched uncertainty has some kind of well-behavedness, then we can be able to do it.

So, in this class, I am going to consider those classes of unmatched uncertainty, which are also bounded. And then we are going to redesign the integral sliding mode control. So, the purpose of the discussion is that we have already seen in the previous class that in classical sliding mode control, we basically have two phases. So, suppose this is sliding surface, so this is generalized sliding surface in two-dimensional plane x_1 and x_2 and we are going to start anywhere from this state space and then our first task to converge towards the equilibrium point and that phase is called the reaching phase and after that I have to maintain trajectory along this particular surface and this is called the sliding phase and what kind of beauty comes into picture if I am going to consider integral sliding mode control. It is possible to show that I am able to start sliding from the initial time onward, which means $t = 0$.

It means that this phase is not basically present, and what is our basic motivation? You can design any nominal control and use that nominal control to remove all kinds of disturbances; whatever the nominal solution, that solution itself is a trajectory. Now, by adding some extra terms based on sliding mode control, I am going to compensate for that uncertainty, and I will explain why exact compensation of uncertainty occurs during the mesh disturbance, because we know that if this is the sliding surface, then $\sigma = 0$ and $\dot{\sigma}$, their average, also equals 0. So, in order to maintain an average equal to 0, and to slide along this surface, it is possible to show that the equivalent value of the discontinuous term is exactly equal to the unmatched uncertainty. And in this way, I can able to maintain the trajectory along this particular sliding surface and then we will reach to our objective. But here, now we are also going to actually consider unmatched uncertainty.

And it is possible to show that in the presence of unmatched uncertainty, if you do not design your gain properly. So, matched uncertainty is compensated for, but unmatched uncertainty is further amplified. your overall performance become more poor than the classical robust control performance and due to that reason, now what I am going to do, I am going to combine the integral sliding mode control with any robust control technique and here I am going to design Robust control based on some method of optimization. So, now I hope that you are able to understand the philosophy. Our control is now containing two part.

One part is obviously integral sliding mode control that is exactly the same as the previous class. Now, during sliding, what am I going to do? I am going to assume that average control is being applied to the system. And then whatever mismatch comes into the picture, I am going to compensate for that mismatch using some standard robust control technique. So, here you are free to apply H_2, H_∞, or any kind of optimal control. I am going to start with exactly the same system, but here this system is different with respect to previous class system, because here you can see that now this uncertainty is not only match can be any general class of uncertainty either that will enter through control channel or not enter through the control channel.

Again, the assumptions are exactly the same. I am assuming that rank of this matrix B through which control is going to enter that is m and this uncertainty is again bounded for all t and x. Here, number of state again is n and control is m. So, somehow our problem is actually multi-input, multi-output problem where output is all state variable. Now, what I am going to do, since this uncertainty or perturbation contains both the matched and unmatched uncertainty.

So, in order to design sliding mode control or integral sliding mode control, what am I going to do? I am going to separate these two classes of uncertainty somehow. Why? We are trying to separate because we already know that integral sliding mode control or any class of sliding mode control is insensitive with respect to any bounded matched uncertainty. So, that is fully compensated and due to that reason I need some kind of way such that I can able to express uncertainty as a matched uncertainty as well as unmatched uncertainty. So, use here for the unmatched uncertainty. Now, if you have some knowledge of the linear algebra, then you can easily able to understand the further development.

In order to support the further development, I am also going to take a very, very simple example. So, even if you are not comfortable with linear algebra, you can understand how to proceed. So, I know that our control matrix that belongs to $\mathbb{R}^m$ and this is $\mathbb{R}^{m\times n}$ because I have termed like if you see the previous class, previous lecture, then you can see that here also I am assuming that B belongs to $m \times n$, $n \times m$ and here similar kind of assumption here. So, basically, this is nothing but an $n \times m$ matrix. Now, since this is an $n \times m$ matrix, it is not a square matrix.

So, obviously, if I do not have any square matrix, then I cannot calculate the inverse. So, for that, I have to take the help of the pseudo inverse, okay. I have defined that as a left inverse. So, some places in literature people are telling pseudo inverse, some place people are telling that as a left inverse. So, now you can see here that our original matrix is B, and this is called $B^{\perp}$, and the original dimension is $n\times m$, and here $B^{\perp}$ is containing the dimension $n \times (n - m)$.

So, now we are assuming that this particular matrix has full rank. I hope that you are familiar with the definition of rank, and I am assuming that the column of this matrix will span the null space of this $B^{\top}$. So, we have matrix B. I will calculate $B^{\top}$, and after that, I am going to design this matrix, which is $B^{\perp}$, and I am going to satisfy this criteria. What does it mean? The meaning of this is that this $B^{\perp}$ is going to fall in the null space of the $B^{\top}$.

And why are we doing this? Because our ultimate goal to separate uncertainty into two parts, matched part or unmatched part. And here, B^{+} represents the left inverse, also called the pseudo-inverse, and that is actually defined like this. Now, what I am going to do, I am going to construct some kind of matrix P that is containing the pseudo inverse and this is the perpendicular of the pseudo inverse. And P^{-1}, you can see that it is B or $B^{\top}$, and I have to design or select this $B^{\perp}$ in such a way that $P^{-1}P$ should be the identity. So, I have to actually maintain that.

If we are able to do this, then it is possible to show that our uncertainty is actually separated into matched or unmatched. So, you just treat this as an algorithm. So, whenever you have some kind of system with a B matrix, you can now apply this kind of algorithm. So, let us try to see an example. So, I have taken a very, very simple example.

I am assuming that $x \in \mathbb{R}^3$. So, I have a three-dimensional system. I am assuming that $u \in \mathbb{R}^2$, and due to that reason, this B becomes a 3×2 matrix. And so, here you can see that this is 3×2 matrix and if you see individual column vector, this is called column vector. So, it has three entries. What is the meaning of three entry? It means that I am in three-dimensional space. So, now if you are in three-dimensional space, in order to expand everywhere inside this space, you need at least three vectors. And I hope that you are already familiar with this concept, such that if I have some vector like this: $(0, 1, 0)$, and after that, $\gamma(0, 0, 1)$. So, using the α, β, γ, I can actually expand anywhere in this three-dimensional space. But what happens here? I have just two vectors in three-dimensional space.

How do you actually calculate dimensions? Just try to see the number of entries inside one column vector. So that will give you dimension. And now at least three vectors, three linearly independent vectors. How can you check if vectors are linearly independent? You can put α, β, and γ multiplied with this equal to 0. And if you have solution α, β, and γ all equal to 0, then that is called linearly independent.

So, these two vectors in three-dimensional space are linearly independent. It means that this B has somehow attained full rank. Now, you can see our aim to calculate the $B^{\perp}$ pseudo-inverse and P, and after that, I have to verify this. So, how do we proceed? First, I have to calculate the transpose. So, our original matrix is

$$B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}.$$

So, this is the transpose:

$$B^{\top} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

Now, regarding the meaning of null space, I have to select a vector v such that if I multiply it by this vector, it will give us 0. And v is nothing but

$$v = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}.$$

So, solution is $v_1 = 0$ and $v_2 = 0$ and v_3 is non-zero. So, this is the basis vector using which I can expand the null space. And here you can take any vector.

So, for simplicity, I have selected 1. So, in this way, I have constructed $B^{\perp}$. So, in order to track this null space, obviously, the null space also has dimension 3 and v_3; you can select anything. Now, you have to calculate the left inverse, which is also called the pseudo-inverse, and this is the formula for that:

$$B^{+} = (B^{\top}B)^{-1}B^{\top}.$$

So, you can just substitute inside this. So, you can see that $B^{\top}B$ is now some kind of a square matrix and this is an identity and due to that reason their inverse is actually exactly same like the $B^{\top}B$. And in this way I can able to calculate the left inverse or pseudo-inverse and I will get this.

Once you get this, what do you have to do now? You have to calculate the B and the perpendicular of B and how to define the pseudo-inverse. Again, you can substitute $B_{\perp}^{+}$ everywhere. And after that, you can substitute their value.

So, at this time, this is 1, and in this way, I will get $B_{\perp}^{+}$ that equal to this vector. Now, what am I going to do? In this way, I am going to construct this P matrix. So, the P matrix is B^{+}. So, I have substituted here, and after that, $B_{\perp}^{+}$ what I have substituted here. So, in this way I can able to calculate P^{-1} and P^{-1} and P is exactly the same and you can see here easily you can able to verify here that

$$BB^{+} + B^{\perp}(B^{\perp})^{+} = I.$$

So, everything is matched. In this way, I can separate the matched uncertainty or unmatched uncertainty. So, a similar kind of theory I am going to apply whenever I have a practical system. Now, since I know how to separate matched uncertainty and unmatched uncertainty, matched uncertainty can be represented by this, unmatched uncertainty can be represented by this, because we have already seen that I have this kind of thing and dimension is going to match, so I am in the generalized case. So, now, I also know that B, B^{+}, and $B^{\top}B^{+}$. So, both are identities, and due to that reason, I have this kind of simpler structure.

So, now I know that this is matched uncertainty and this is actually unmatched uncertainty. Now, again, I am going to design control in two parts. So, this is nominal control, and this is actually switching-based control or sliding-based control, and obviously, this control is discontinuous in nature. And how do you define the integral sliding manifold? I have already told you that you can remove all kinds of uncertainty, and then you can apply the control, nominal control. Nominal control can be designed by PID, whatever is optimal, whatever you want.

Nowadays, one fancy trend people are designing by data driven based, you can able to do. Based on machine learning, you can be able to do. You can do reinforcement learning-based. Every possibility is here. Now, only important factor here that how to design this projection matrix.

In the previous class, I already told you that you can select $G = B^{\top}$. But here, since you have unmatched uncertainty as well, you have to actually select G in a specific way so that you can minimize the unmatched uncertainty. Here, I am also again commenting that you cannot able to nullify fully the any class of unmatched uncertainty, but just you are able to minimize it. And due to that reason, I am going to apply realistic approach that I am going to minimize it. Control, switching-based control is exactly the same as the previous one, but here you can see that the control gain has to be kept a little bit higher because I have matched uncertainty as well as unmatched uncertainty; this is the upper bound of the matched uncertainty.

Now, I have designed a sliding surface like this, so the philosophy is exactly the same: if you have designed the sliding surface, you can calculate the rate of change and then try to maintain the rate of change such that $S = 0$. So, the same philosophy I am going to adopt here. So, after the calculation of $\dot{S}$, in this expression, you can see that $\dot{x}$ will come here, this term will cancel out, this becomes 0, and then this integral, so t is substituted here, and then I will get this kind of expression.

Now, suppose that I have considered the generalized case; it might be possible in some cases if I have B and B^+ that is not equal to the identity. So, I am always considering the generalized case; by construction, sometimes you are able to get identity, and sometimes you are not able to get identity.

So, that case also you can able to consider. Now, what am I going to do? I am going to rearrange the terms. Whatever term is canceled, I am going to cancel it. And after that, now I have this kind of structure. And I have told you that philosophy is like this. Since all plant is almost the low pass filter, so during sliding, it is possible to show that some kind of equivalent control comes into picture and how basically this control one can able to drive by putting $\dot{S} = 0$ and this control is responsible for maintaining the sliding surface.

So, that kind of result is actually shown by Professor Utkin. So, Professor Utkin is no more, but he has contributed a lot to the direction of sliding mode control. So, what I am going to do is, since I know that during sliding, if we are exactly on the sliding surface, then this control is required. And due to that reason, what I am going to do, I am going to substitute this control, equivalent control in this dynamics.

Once I substitute, I have this kind of dynamics. And if you look carefully, you will see that if this term is not present, then our system is governed by nominal control. So, what is our objective? Now, our objective is to minimize this term.

Okay. Due to that reason, I have defined this term as a separate variable. Okay. And now, how do we minimize this term? You can see that this is coming from the system.

So, I cannot alter. Okay. The identity matrix is, obviously, a constant one. So, only one guy with which we can able to play, that is the projection matrix G. Okay. So, now what is our ultimate goal? how to select this G such that I can able to minimize this amount of the perturbation. So, original perturbation is obviously, $d\phi_u$ because what happens that during sliding $\phi_{\text{eq}} = d\phi_u$.

So, that kind of things I have to take care. What I am going to do is define this as a $G^\star$ that is nothing but $B^\top$, and that is given like this. Now, what we are saying is that if I select $G = B^\star$, and B, $B^\top$, and B^+ are the matrices that minimize the norm ϕ equivalent. So, that problem is equivalent to this problem because ultimately I have to minimize this. So, now you can see why if I select $G = B^\top$, then I will get here *arg* means what kind of G I will select such that. So, this is the independent variable, and these whole things are independent variables; that is the meaning, such that I will get the minimum value of this whole expression.

That is some class of optimization problems. So, what am I going to define now? I am going to define κ; this whole term I will define as a κ. So, now I have ϕ_u; ϕ I am going to multiply, which is nothing but $B\kappa$. And after that, now I have to define this $K^\star$; this is nothing but the optimal value such that this expression has the minimum value. And when this is the minimum, you can easily see that the norm is minimum when I am able to select $K^\star$, which is B^{-1}. So, B is actually not a square matrix, and for that reason, I have to calculate the pseudo inverse of this.

So, if I select like this, then this value is the minimal, easily one can able to see and due to that reason by taking $G = B^\top$, we can able to get the optimal value. So, the main theme of this particular proposition or theorem is that if you select G as $B^\top$, in the previous class I also told you that whenever you are designing integral control, the best choice is $G = B^\top$, because that is the optimal choice in any class of disturbance, even if I have no unmatched uncertainty. So, it is better to select this option. It is also possible to show that even if you select $G = B^+$, then this is minimal.

So, you are free to select any one of them. So, I will always suggest that you can select $G = B^\top$. So, what am I going to do? I am again going to take exactly the same example as in the previous class. I have linearized this model. This time I am going to consider both kinds of uncertainty: matched uncertainty as well as unmatched uncertainty. Now, what I am going to do, since I have only one control input, so now this is the B matrix, so again I will actually separate B matrix like this such that I can able to get matched uncertainty and unmatched uncertainty.

So, one of the choices of $B^\top$ is that it is possible to show it like this. You can easily do calculations. I have already shown you that this calculation is not unique. Why? Because null vector people can decide by themselves and due to that reason selection may or may not be same. Now, ϕ_m can be given like this, and ϕ_u, uncertainty part that is one can able to show now that is given like this.

And now, I will design the control again based on the Ackermann formula. $G^\star$ I am going to select; that is, nothing but the $B^\top$, and then I will design the control. You can see that in unmatched uncertainty, I also have satisfactory performance. You can see that control is not switching at very high speed. Why? Because of unmatched uncertainty, I am not able to achieve exact sliding.

So, you can see a similar kind of thing here. What is the meaning of exact sliding? That $\sigma = 0$, but $\sigma \neq 0$ due to perturbation; they are actually oscillating like this, and for that reason, the control behavior is not like this. And this is the MATLAB code, you can directly copy this code and actually run in MATLAB environment and you can able to exactly retrace the result. Now, here this control you can able to change by another control also. So, after that, everything is just the coding of whatever development I have done.

Now it is time to conclude. This lecture. So, what we have seen is that integral sliding mode control can also handle the bounded unmatched uncertainty. And obviously, if there is no unmatched uncertainty, then we can able to see sliding from initial point onward and due to that reason, any safety critical application, because robotics, now robotics is also used for surgery purpose for a space mission several places. And due to that reason, sliding mode control is one of the very good choices for that class of problems. And obviously, integral sliding mode is more useful because people have already known the performance of classical control.

So, now there is no need to change the control. What we are suggesting is that you can add one extra loop, and you can design that extra loop based on sliding mode control, specifically integral sliding mode control. What is the basic philosophy? Suppose you have already applied some kind of control. Now, if you need some kind of robustness or insensitivity, then you can just add one more control loop that is based on integral sliding mode control. So, with this remark, I will end this class. Thank you very much.