Proceedings of the 10th International Conference of Control Systems, and Robotics (CDSR'23)
Ottawa, Canada – June 01 - 03, 2023
Paper No. 201
DOI: 10.11159/cdsr23.201

Model-Free Sliding Mode Control in the Lateral and Direction Dynamics of an Aircraft

Nicholas Hutson[1], Agamemnon Crassidis[2]
[1,2]Rochester Institute of Technology
1 Lomb Memorial Drive, Rochester, New York, 14623, USA
ndh4970@rit.edu; alceme@rit.edu

Abstract **-** Traditional control methods require extensive tuning or a derivation of a system model making them increasingly antiquated for use on new, more complex systems. Sliding Mode Control has emerged as a more effective alternative as a control method that can directly handle nonlinear systems with increased robustness while guaranteeing stability. However, it is still limited by the need for a system model for the derivation of the controller form. This work proposes a new model-free control method based on Sliding Mode Control referred to as Model-Free Sliding Mode Control where the form of the controller is only dependent on system order, state measurements, and previous control inputs. Lyapunov's stability theorem is used to ensure global asymptotic stability and a boundary layer is incorporated to reduce chattering. The model-free properties of the controller are enabled by a least-squares online parameter estimation method used to estimate the control input influence gain matrix of the system directly. The estimation method incorporated exponential forgetting to only include updated data for parameter estimation, increasing the speed of convergence. Another addition was a bounded gain forgetting factor to ensure that the magnitude of the control input influence gain was upper bounded. The performance of this controller was simulated on a single-input, single-output second order example system. It was also implemented to control a first order example system with a shaped input characterizing aircraft roll dynamics. The controller proved to exhibit outstanding tracking performance, convergence of estimated parameters, acceptable control input, and robustness to parameter uncertainty in all cases.

Keywords**:** Nonlinear control, model-free, aircraft, robust control, artificial intelligence

1. Introduction

Interest in the advancement of control system design has proliferated as systems have become more complex. Technological development has allowed for the use of much smaller and faster computers which have made the use of advanced control systems feasible for a wide variety of applications. Nonlinear control systems such as Sliding Mode Control (SMC) have been developed as more effective alternatives for when an accurate linear approximation of the system's dynamics cannot be derived in higher-order nonlinear systems. SMC provides a greater robustness in the face of modeling uncertainty and external disturbances and theoretically has perfect tracking performance. SMC still has inherent drawbacks such as chattering which causes high controller effort. Traditional SMC also requires a mathematical reference or plant model for the derivation of the control law. In modern times, it is increasingly difficult to derive a mathematical model of systems as they become more complex leading to higher parametric uncertainty.

Thus, a huge benefit is realized in the development a controller that is not dependent on a system model and can be generalized to all systems. Sariful and Crassidis [1] proposed a Model-Free Sliding Mode Controller (MFSMC) developed from the work of Reis and Crassidis [2] including a least-squares online parameter estimation law to estimate the increment to the switching gain in a time-varying boundary layer. The updated method only required knowledge of the system order, state measurements, and the previous control inputs making it truly model-free. However, this method assumed a unitary input influence gain which is not true for most systems. The method developed here uses the least-squares online parameter estimation method proposed by authors Sariful and Crassidis [1] to estimate the control input influence gain in real-time in place of the increment to the switching gain while guaranteeing convergence of the estimated input influence gain. This is in an effort to neutralize the degraded handling qualities observed when Stephens [3] implemented the Sariful and Crassidis [1] method in the longitudinal axis for pitch rate control of the Calspan Variable Stability System (VSS) Learjet. Thus, this method is also completely model-free. This method was applied to the control of a second-order single-input, single-output (SISO) example system as well as a single-order SISO example system with a shaped step input mimicking aircraft roll dynamics as pre-requisite to lateral-directional aircraft control. The controller provided great tracking performance in both cases.

2.1 Model-Free Control System Law

The MFSMC system for a second-order SISO system was derived with the following steps:

The following discrete-time measurement model is assumed:

$$\ddot{x} = \ddot{x} + bu - bu_{k-1} - bu + bu_{k-1} \quad (2.1)$$

where $\ddot{x}$ is the measured acceleration, u is the control system input, u_{k-1} is the previous value of the control system input, and b is the control input influence gain that will be estimated. This equation can be re-written as:

$$\ddot{x} = \ddot{x} + bu - bu_{k-1} + \varepsilon(u) \quad (2.2)$$

where $\varepsilon(u)$ is the estimation error in the input influence gain defined by:

$$\varepsilon(u) = b(u_{k-1} - u) \quad (2.3)$$

$\varepsilon(u)$ is assumed to be bounded by a known function, E, such that:

$$|\hat{\varepsilon}(u) - \varepsilon(u)| \leq E \quad (2.4)$$

where $\hat{\varepsilon}(u)$ is the estimated error in the input influence gain estimation assumed to be defined by:

$$\hat{\varepsilon}(u) = \hat{b}(u_{k-2} - u_{k-1}) \quad (2.5)$$

and the actual error is bounded by the function:

$$(1 - \sigma_l)\hat{\varepsilon}(u) \leq \varepsilon(u) \leq (1 + \sigma_u)\hat{\varepsilon}(u) \quad (2.6)$$

where σ_l and σ_u are the lower and upper defined bounds.

The sliding surface for a second-order system is defined as:

$$s = \dot{x} - \dot{x}_d + \lambda(x - x_d) \quad (2.7)$$

where x and $\dot{x}$ are the system states to be measured and x_d and $\dot{x}_d$ are the desired states to be tracked. Taking the derivative of the sliding surface results in:

$$\dot{s} = \ddot{x} - \ddot{x}_d + \lambda(\dot{x} - \dot{x}_d) \quad (2.8)$$

Plugging in Equation 2.2 and setting $\dot{s}$ equal to zero ensures that the state error trajectories do not move once they reach the sliding surface and gives us:

$$\dot{s} = [\ddot{x} + bu - bu_{k-1} + \varepsilon(u)] - \ddot{x}_d + \lambda(\dot{x} - \dot{x}_d) = 0 \quad (2.9)$$

The best estimate for the control input, $\hat{u}$, to maintain $\dot{s}$ equal to zero is therefore:

$$\hat{u} = \hat{b}^{-1}[-(\ddot{x} - \ddot{x}_d) - \lambda(\dot{x} - \dot{x}_d) - \hat{\varepsilon}(u)] + u_{k-1} \quad (2.10)$$

and adding a discontinuous term to satisfy the sliding condition gives us:

$$u = b^{-1}[-(\ddot{x} - \ddot{x}_d) - \lambda(\dot{x} - \dot{x}_d) - \varepsilon(u) - \eta sgn(s)] + u_{k-1} \quad (2.11)$$

where η is a positive constant. To check if the controller form is correct, Lyapunov's stability theorem is used and a positive definite "energy-like" function is defined as:

$$V = \frac{1}{2}s^2 \geq 0 \quad (2.12)$$

Taking the derivative results in:

$$\dot{V} = s\dot{s} \leq 0 \quad (2.13)$$

Substituting the assumed measurement model based on Equation 2.9 gives us:

$$\dot{V} = s\big[[\ddot{x} + \hat{b}u - \hat{b}u_{k-1} + \hat{\varepsilon}(u)] - \ddot{x}_d + \lambda(\dot{x} - \dot{x}_d)\big] \quad (2.14)$$

Substituting Equation 2.11 gives us:

$$\dot{V} = s\big[[\ddot{x} + \hat{b}\{\hat{b}^{-1}[-(\ddot{x} - \ddot{x}_d) - \lambda(\dot{x} - \dot{x}_d) - \hat{\varepsilon}(u) - \eta sgn(s)] + u_{k-1}\} - \hat{b}u_{k-1} + \hat{\varepsilon}(u)] - \ddot{x}_d + \lambda(\dot{x} - \dot{x}_d)\big] \quad (2.15)$$

Rearranging this equation can give us:

$$\dot{V} = s\big[\ddot{x} - (\dot{x} - \dot{x}_d) - \ddot{x}_d - \lambda(\dot{x} - \dot{x}_d) + \lambda(\dot{x} - \dot{x}_d) - \hat{\varepsilon}(u) + \hat{\varepsilon}(u) - \eta sgn(s) + \hat{b}u_{k-1} - \hat{b}u_{k-1}\big] \quad (2.16)$$

which results in:

$$\dot{V} = -s\eta sgn(s) \quad (2.17)$$

which can be re-written as:

$$\dot{V} = -\eta|s| \quad (2.18)$$

This verifies the sliding condition. Therefore, η can be replaced with the system gain, K, in Equation 2.11:

$$u = \hat{b}^{-1}[-(\ddot{x} - \ddot{x}_d) - \lambda(\dot{x} - \dot{x}_d) - \hat{\varepsilon}(u) - Ksgn(s)] + u_{k-1} \quad (2.19)$$

From the definition of $\dot{V}$, we can define the sliding condition as:

$$s\dot{s} \leq -\eta|s| \tag{2.20}$$

Substituting in Equation 2.9 gives us:

$$s\left[\left(\ddot{x} + bu - bu_{k-1} + \varepsilon(u)\right) - \ddot{x}_d + \lambda(\dot{x} - \dot{x}_d)\right] \leq -\eta|s| \tag{2.21}$$

Substituting in the best estimate of the control input, $\hat{u}$, for u gives us:

$$s\left[\left(\ddot{x} + b\{\hat{b}^{-1}[-(\ddot{x} - \ddot{x}_d) - \lambda(\dot{x} - \dot{x}_d) - \hat{\varepsilon}(u) - Ksgn(s)] + u_{k-1}\} - bu_{k-1} + \varepsilon(u)\right) - \ddot{x}_d + \lambda(\dot{x} - \dot{x}_d)\right] \leq -\eta|s| \tag{2.22}$$

Rearranging to isolate $K|s|$ gives us:

$$K|s| \geq s\left[(\ddot{x} - \ddot{x}_d)(\hat{b}b^{-1} - 1) + \lambda(\dot{x} - \dot{x}_d)(\hat{b}b^{-1} - 1) + (\hat{b}b^{-1} - 1)\hat{\varepsilon}(u) + \hat{b}b^{-1}[\varepsilon(u) - \hat{\varepsilon}(u)]\right] + \hat{b}b^{-1}\eta|s| \tag{2.23}$$

The most conservative estimate for the upper bound of $\varepsilon(u)$ is defined as:

$$\varepsilon(u) = (1 + \sigma_u)\hat{\varepsilon}(u) \tag{2.24}$$

Therefore, the $\varepsilon(u) - \hat{\varepsilon}(u)$ term can be redefined as:

$$\varepsilon(u) - \hat{\varepsilon}(u) = (1 + \sigma_u)\hat{\varepsilon}(u) - \hat{\varepsilon}(u) = \hat{\varepsilon}(u) + \sigma_u\hat{\varepsilon}(u) - \hat{\varepsilon}(u) = \sigma_u\hat{\varepsilon}(u) \tag{2.25}$$

Plugging this into Equation 2.23 gives us:

$$K|s| \geq s\left[(\ddot{x} - \ddot{x}_d)(\hat{b}b^{-1} - 1) + \lambda(\dot{x} - \dot{x}_d)(\hat{b}b^{-1} - 1) + (\hat{b}b^{-1} - 1)\hat{\varepsilon}(u) + \hat{b}b^{-1}\sigma_u\hat{\varepsilon}(u)\right] + \hat{b}b^{-1}\eta|s| \tag{2.26}$$

And rearranging this gives us:

$$K|s| \geq s\left[(\ddot{x} - \ddot{x}_d)(\hat{b}b^{-1} - 1) + \lambda(\dot{x} - \dot{x}_d)(\hat{b}b^{-1} - 1) + (\hat{b}b^{-1}[1 + \sigma_u] - 1)\hat{\varepsilon}(u)\right] + \hat{b}b^{-1}\eta|s| \tag{2.27}$$

Next, we define:

$$\hat{b} = \sqrt{b_{low}b_{upp}} \tag{2.28}$$

$$\beta = \sqrt{b_{low}/b_{upp}} = \hat{b}b^{-1} \tag{2.29}$$

Plugging these into Equation 2.27 gives us:

$$K|s| \geq s[(\ddot{x} - \ddot{x}_d)(\beta - 1) + \lambda(\dot{x} - \dot{x}_d)(\beta - 1) + (\beta[1 + \sigma_u] - 1)\hat{\varepsilon}(u)] + \beta\eta|s| \tag{2.30}$$

With the final equation for the system gain being:

$$K \geq |(\beta - 1)||(\ddot{x} - \ddot{x}_d)| + |(\beta - 1)||\lambda(\dot{x} - \dot{x}_d)| + |(\beta[1 + \sigma_u] - 1)||\hat{\varepsilon}(u)| + \beta\eta \tag{2.31}$$

In summary:

$$u = \hat{b}^{-1}[-(\ddot{x} - \ddot{x}_d) - \lambda(\dot{x} - \dot{x}_d) - Ksgn(s)] + 2u_{k-1} - u_{k-2}$$

$$K \geq |(\beta - 1)||(\ddot{x} - \ddot{x}_d)| + |(\beta - 1)||\lambda(\dot{x} - \dot{x}_d)| + |(\beta[1 + \sigma_u] - 1)||\hat{b}(u_{k-2} - u_{k-1})| + \beta\eta$$

If including a boundary layer:

$$u = \hat{b}^{-1}\left[-(\ddot{x} - \ddot{x}_d) - \lambda(\dot{x} - \dot{x}_d) - Ksgn(\frac{s}{\phi})\right] + 2u_{k-1} - u_{k-2}$$

$$K \geq |(\beta - 1)||(\ddot{x} - \ddot{x}_d)| + |(\beta - 1)||\lambda(\dot{x} - \dot{x}_d)| + |(\beta[1 + \sigma_u] - 1)||\hat{b}(u_{k-2} - u_{k-1})| + \beta\eta$$

$$K_d = |(\beta[1 + \sigma_u] - 1)||\hat{b}(u_{k-2} - u_{k-1})| + \beta_d\eta$$

$$K_d \leq \frac{\lambda\phi}{\beta_d} \Rightarrow \dot{\phi} + \lambda\phi = \beta_d K_d \qquad \overline{K} = K - \frac{\dot{\phi}}{\beta}$$

$$K_d \geq \frac{\lambda\phi}{\beta_d} \Rightarrow \dot{\phi} + \frac{\lambda\phi}{\beta_d^2} = \frac{K_d}{\beta_d} \qquad \overline{K} = K - \beta\dot{\phi}$$

$$\phi(0) = \frac{\beta_d K_d(0)}{\lambda}$$

2.2 Online Parameter Estimation Methods

The basis of parameter estimation is extracting parameter values from measurable system outputs. A general model for parameter estimation can be defined as (Slotine and Li [4]):

$$y(kT) = W(kT)b(kT) \tag{2.32}$$

where b is the parameter to be estimated, which in our case is the input influence gain, vector y includes the outputs from the system used for estimation, and W is a signal matrix. This model is only valid for discrete time so T is the sampling time and k is zero or a positive integer. The predicted system output, $\hat{y}$, at time kT, can be defined as:

$$\hat{y}(kT) = W(kT)\hat{b}(kT) \tag{2.33}$$

where $\hat{b}$ is the predicted parameter at time t. Online estimation is based around the fact that the value of $\hat{b}$ is found recursively. In other words, it is updated every time there is a new set of data for y and W. The instantaneous prediction error, e_1, can then be defined as:

$$e_1 = \hat{y}(kT) - y(kT) \tag{2.34}$$

Or, plugging in Equations 2.32 and 2.33:

$$e_1 = W\hat{b}(kT) - Wb(kT) = W\tilde{b} \tag{2.35}$$

2.2.1 Standard Least-Squares Estimator

The standard least-squares method has the advantage of averaging out the effects of noise in measurements. This method can be implemented by minimizing the total prediction error with respect to $\hat{b}^T(t)$ (Slotine and Li [4]):

$$J = \int_o^t \left\| y(r) - W(r)\hat{b}(t) \right\|^2 dr \tag{2.36}$$

where the estimated parameter $\hat{b}$ satisfies:

$$\int_o^t (W^T(r)W(r)dr)\hat{b}(t) = \int_o^t W^T ydr \tag{2.37}$$

We can then define:

$$P(t) = [\int_o^t (W^T(r)W(r)dr)]^{-1} \tag{2.38}$$

For computational efficiency, it is better to calculate P, the estimator gain matrix, recursively, so the above equation can be replaced with the following differential equation:

$$\dot{P}^{-1} = W^T(t)W(t) \tag{2.39}$$

This can give us the equation:

$$\frac{d}{dt}[\hat{b}] = -P(t)W^T e_1 \tag{2.40}$$

To be able to update P directly, we use the following identity:

$$\frac{d}{dt}[PP^{-1}] = \dot{P}P^{-1} + P\dot{P}^{-1} = 0 \tag{2.41}$$

to get the equation:

$$\dot{P} = -PW^T WP \tag{2.42}$$

To successfully implement this method, P and $\hat{b}$ must be initialized with finite values. The initial value of P should be as high as possible within noise sensitivity constraints and $\hat{b}$ should be a best guess.

2.2.2 Convergence

Solving the differential Equations 2.39 and 2.40 above and using Equation 2.42 we can show that (Slotine and Li [4]):

$$P^{-1}(t) = P^{-1}(0) + \int_0^t W^T(r)W(r)dr \tag{2.43}$$

$$\frac{d}{dt}[P^{-1}(t)\tilde{b}(t)] = 0 \tag{2.44}$$

From which we can define:

$$\tilde{b}(t) = P(t)P^{-1}(0)\tilde{b}(0) \tag{2.45}$$

If W meets the criteria of:

$$\lambda_{min} \int_0^t W^T W dr \to \infty \text{ as } t \to \infty$$

where λ_{min} is the smallest eigenvalue of W, then the gain matrix converges to zero and the estimated parameters asymptotically converge to their true values. Additionally, for any positive integer value of k:

$$\int_0^{k\delta+\delta} W^T W dr = \sum_{i=0}^{k} \int_{i\delta}^{i\delta+\delta} W^T W dr \geq k\alpha_1 I \tag{2.46}$$

where δ and α_1 are positive constants. Therefore, if W is under persistent excitation, the above equation is satisfied and $P \to 0$ and $\tilde{b} \to 0$. An initial parameter error, $\tilde{b}(0)$, or large initial gain, $P(0)$, can lead to a small parameter error for all time. If the initial gain is chosen such that $P(0) = p_0 I$, then:

$$\tilde{b}(t) = \left[I + p_0 \int_0^t W^T(r)W(r)dr\right]^{-1} \tilde{b}(0) \tag{2.47}$$

2.2.3 Least-Squares with Exponential Forgetting

When estimating time-varying parameters, it is known that past data is generated by past parameter values. Therefore, this past data should be discounted when estimating the current value of parameters. To implement exponential forgetting into least square estimation, the following cost function is defined (Slotine and Li [4]):

$$J = \int_o^t exp[-\int_s^t \lambda(r)dr] \left|\left|y(s) - W(s)\hat{b}(t)\right|\right|^2 ds \tag{2.48}$$

where $\lambda(t) \geq 0$ is the time-varying forgetting factor. The parameter update law stays the same as:

$$\frac{d}{dt}\hat{b} = -P(t)W^T e_1 \tag{2.49}$$

but the gain update law has changed to be:

$$\frac{d}{dt}[P^{-1}(t)] = -\lambda(t)P^{-1} + W^T(t)W(t) \tag{2.50}$$

which can be implemented to directly update P in the form of:

$$\dot{P} = \lambda(t)P - PW^T(t)W(t)P \tag{2.51}$$

This exponential forgetting method improves parameter convergence over the traditional least-squares method by creating exponential convergence of the parameters to their final values. This is done while still guaranteeing asymptotic convergence of estimated parameters.

2.2.4 Control Law Implementation

Including a boundary layer, we define the sliding condition as:

$$s\dot{s} \leq (\dot{\phi} - \eta)|s| \tag{2.52}$$

To ensure the global asymptotic stability of the system, we want to ensure that the sliding condition is met. Thus, we define our instantaneous parameter estimation error based on meeting the sliding condition as shown below:

$$e_1 = (\dot{\phi} - \eta)|s| - s\dot{s} \tag{2.53}$$

The value of e_1 is minimized as the state trajectories reach the boundary layer. A boundary layer closing function is also defined as:

$$|s| \geq \phi \Rightarrow e_1 = (\dot{\phi} - \eta)|s| - s\dot{s} \tag{2.54}$$

$$|s| < \phi \Rightarrow e_1 = -\left[(\dot{\phi} - \eta)|s| - s\dot{s}\right] \tag{2.55}$$

As the state trajectory reaches a point within the boundary layer, this function starts to reduce the size of the boundary layer. Reducing the size of the boundary layer decreases tracking error and controller input over time.

For the implementation of Equation 2.51, the signal matrix, W was defined as:

$$W = |s| \tag{2.56}$$

This was done to ensure that the value of $\hat{b}$ varied while the value of the sliding surface varied over time.

2.2.5 Bounded Gain Forgetting Factor Tuning

The benefit of data forgetting is the ability to track slowly varying parameters, but the gain matrix P can grow unbounded when W is not persistently exciting. It is desirable to tune the forgetting factor such that data forgetting is active when W is persistently exciting and not active when W is not exciting. The magnitude of P shows the excitation level of W. Therefore, the forgetting factor variation can be made dependent on $\|P(t)\|$ (Slotine and Li [4]) such that:

$$\lambda(t) = \lambda_0 \left(1 - \frac{\|P(t)\|}{k_0}\right) \tag{2.57}$$

where λ_0 is the maximum forgetting rate and k_0 is the bound for the gain matrix magnitude and both are positive constants. A higher value of λ_0 leads to faster forgetting but more oscillations in the estimated parameters. A higher value of k_0 updates the parameter estimation values faster but makes the estimator less robust to disturbances in the prediction error. In order for k_0 to be the upper bound of the gain matrix, we choose $\|P(0)\| \leq k_0$.

2.3 Implementation Results

2.3.1 SISO System

The derived control law was implemented on the following nonlinear second-order system:

$$\ddot{x} + 3x\dot{x} + 5x^2 = bu$$

with the desired tracking being:

$$x_d(t) = sin\left(\frac{\pi}{2}t\right)$$

To demonstrate the robustness of this control law, the input influence gain was varied using a sine wave defined as:

$$b(t) = \left(\frac{b_{upp} - b_{low}}{2}\right)\sin(t) + \frac{b_{upp} + b_{low}}{2}$$

where b_{upp} and b_{low} are the upper and lower estimated bounds of the input influence gain respectively. The results of simulating the system with the derived control law for 60 seconds with a sampling of 0.001 seconds are shown below:

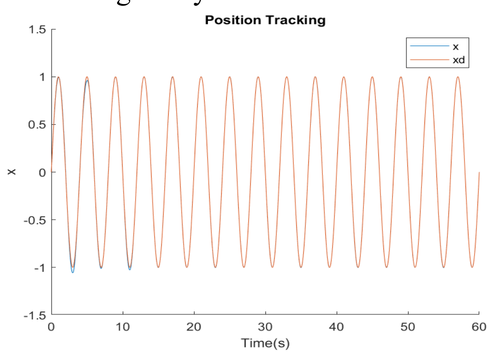

Figure 7: Position Tracking

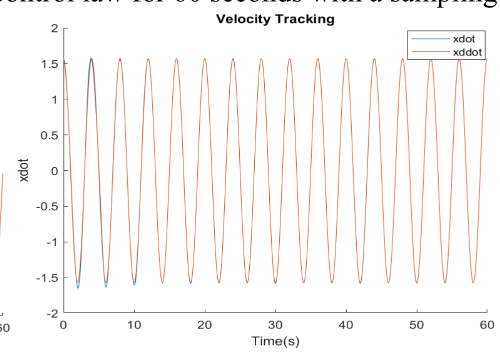

Figure 8: Velocity Tracking

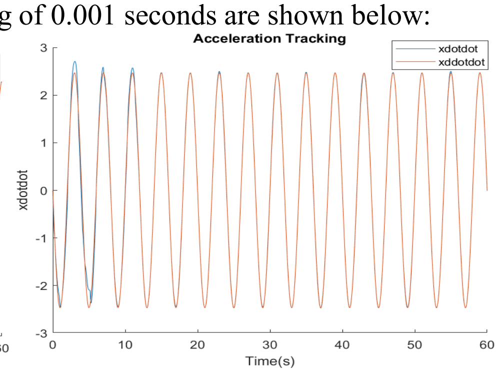

Figure 9: Acceleration Tracking

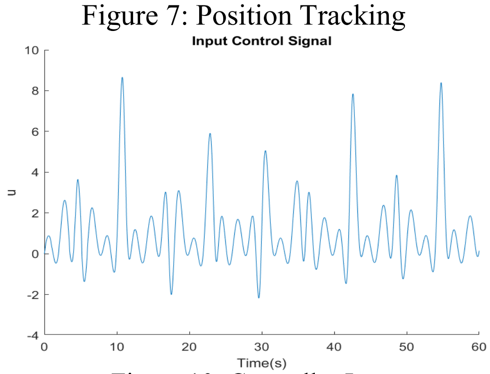

Figure 10: Controller Input

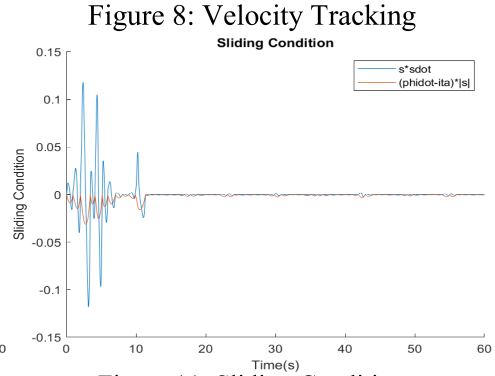

Figure 11: Sliding Condition

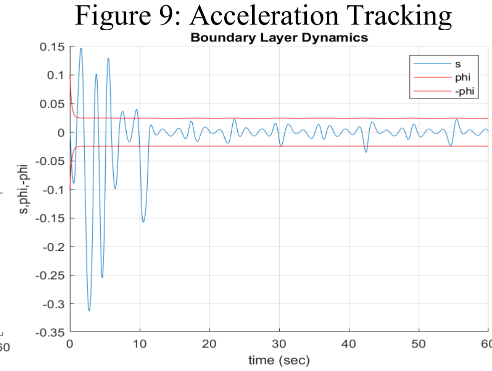

Figure 12: Boundary Layer Dynamics

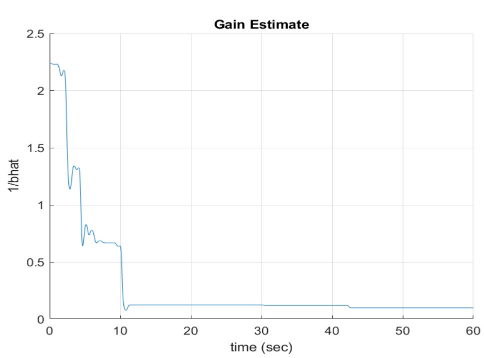

Figure 13: Input Influence Gain Estimate

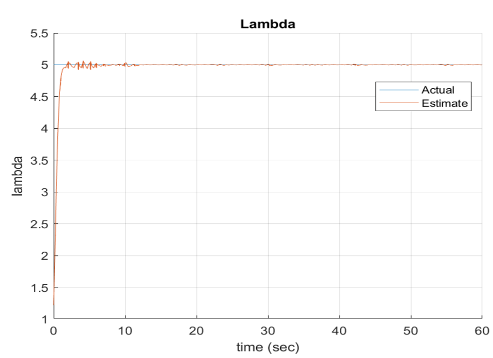

Figure 14: Lambda Estimate

As seen in Figures 7, 8, and 9, the system had excellent tracking. The sliding condition was also met as time went on and the estimate for the input influence gain and λ were both convergent as shown by Figures 11, 13, and 14. This was accomplished with a controller input that was smooth and adequately small as shown in Figure 10.

Note: the control input influence gain value does not need to be known before simulating the system with the control system as the control system is wholly model-free. However, to decrease controller activity, the upper and lower bounds of the input influence gain estimate can be redefined in further simulations to encompass the final estimated value of the input influence gain.

2.3.2 First-Order System

In preparation for the control of roll and yaw rate for an aircraft, the control law was updated to control a first-order SISO system. This included the re-derivation of the control law after Equations 2.7 and 2.8 are updated such that:

$$s = x - x_d + \lambda \int_0^t (x - x_d)dt \tag{2.57}$$

$$\dot{s} = \dot{x} - \dot{x}_d + \lambda(x - x_d) \tag{2.58}$$

This control law was implemented on an example first-order SISO system defined as:

$$\dot{x} + 7x + 5x^3 = bu$$

where the input influence gain, b, was time-varying as described in Section 2.3.1. The desired tracking signal was updated to be a step function with a magnitude of 1 ran through a transfer function. To generate a desired "roll" signal, a first-order transfer function was used such that:

$$\frac{X_d}{U} = \frac{K}{\tau s + 1}$$

where U is the step function input, τ is a roll-mode time constant set to 0.3 seconds, and K is a gain set to 1. The results of simulating the system with the updated control law for 60 seconds with a sampling of 0.001 seconds are shown below:

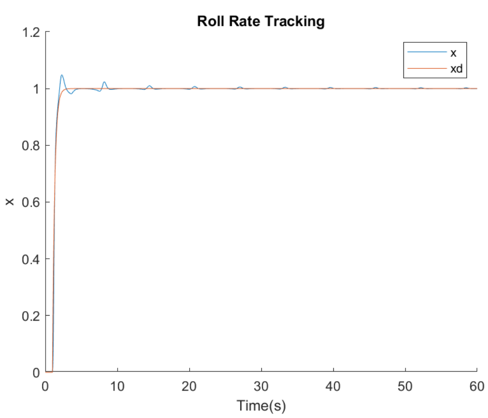

Figure 15: Roll Rate Tracking

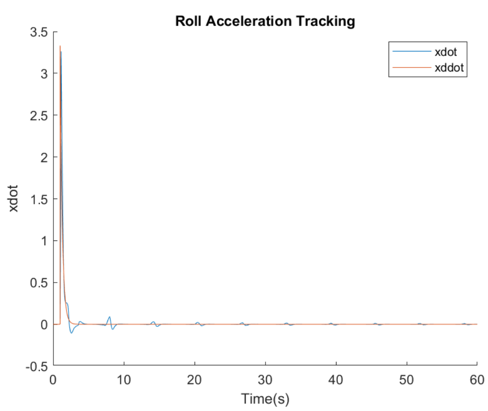

Figure 16: Roll Acceleration Tracking

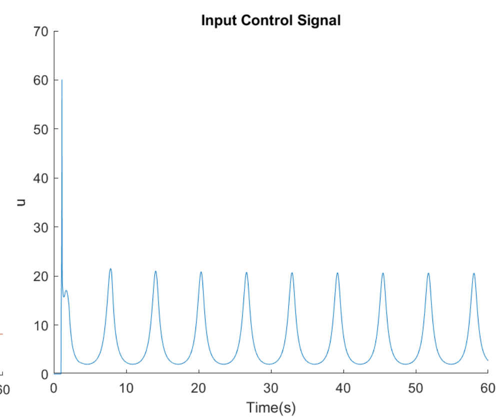

Figure 17: Control Input

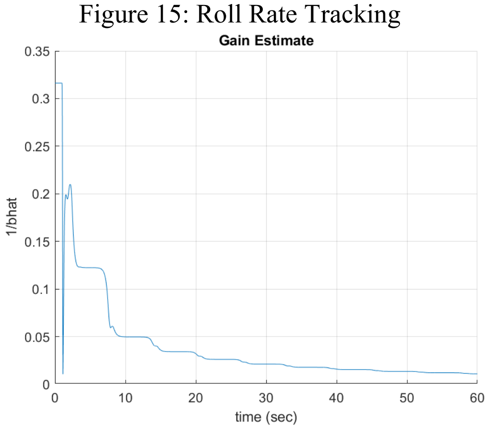

Figure 22: Input Influence Gain Estimate

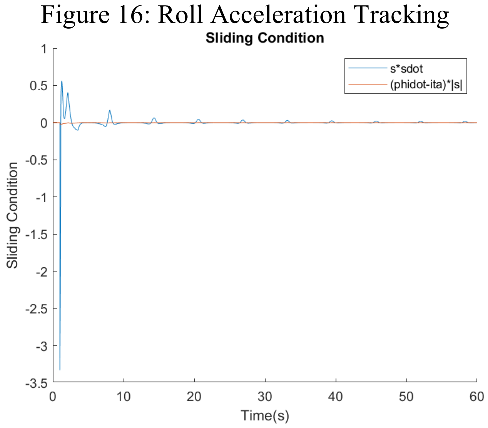

Figure 20: Sliding Condition

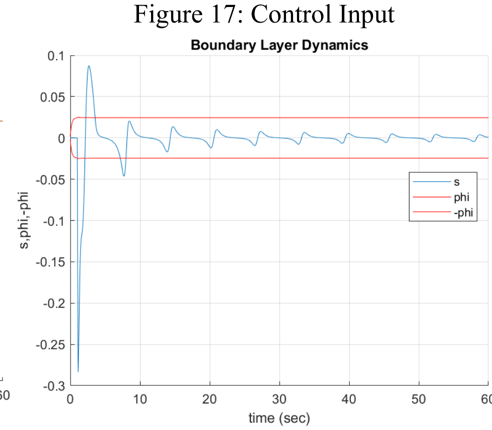

Figure 21: Boundary Layer Dynamics

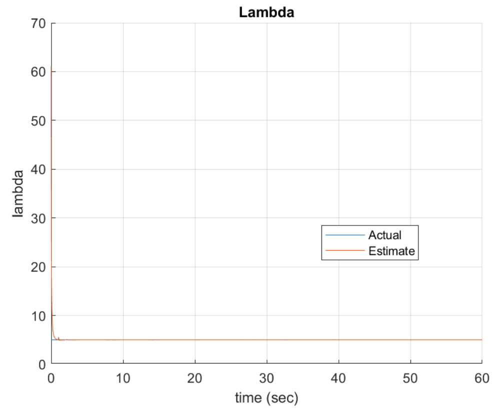

Figure 23: Lambda Estimate

As seen in Figures 15 and 16, the controller still provides excellent tracking of roll rate and acceleration. Again, the sliding condition is met and the estimates for the input influence gain and λ are both convergent while the control input is smooth and adequately small.

4. Conclusion

The majority of existing control methods require extensive development and tuning. This is becoming a problem as nonlinear systems become more complex with advancements in technology. It is becoming time intensive to define a mathematical model of these systems. Oftentimes, the final models are not perfect and have parameter uncertainty. SMC is being applied more widely for its robustness against modeling uncertainty and external disturbances. Thus, SMC saves time and money by eliminating most of the development and tuning needed. However, accurate approximations of system dynamics are still needed for traditional SMC design.

The work shown here is truly model-free, requiring only knowledge of the system order, state measurements, and the previous control inputs for derivation of the control law. Thus, the time, money, and effort needed for the system modeling and tuning usually required by traditional SMC and other control methods is eliminated. The MFSMC derived in section 2.1 can be expanded to MIMO systems which will expand the utility of the MFSMC to a wide range of systems. This includes the control of an aircraft. The re-derivation and application of the control law for single order systems described in section 2.3.2 is in preparation for future work to control an aircraft in the lateral and directional axes. The MFSMC is theoretically the most efficient in terms of fuel-efficiency as the controller output is only dependent on state measurements. Thus, the development and application of the MFSMC to aircraft autopilot designs and other similar applications will increase fuel-efficiency and reduce emissions.

References

[1] S. Islam, "A Model-Free Control System Based on the Sliding Mode Control with Automatic Tuning Using as On-Line Parameter Estimation Approach," 2020.

[2] R. M. Reis, "A New Model-Free Sliding Mode Control Method with Estimation A New Model-Free Sliding Mode Control Method with Estimation of Control Input Error of Control Input Error," 2016. [Online]. Available: https://scholarworks.rit.edu/theses

[3] S. S. Stephens, M. Navarro, K. Ban, J. A. Fischer, M.-A. Pelletier, T. Sick, M. Trautschold, M. C. Cotting, and A. L. Crassidis, "A Limited Evaluation of a Model-Free Control Law System," *Proceedings of the AIAA SCITECH 2023 Forum*, AIAA 2023-0483, National Harbor, Maryland, Jan. 2023. doi: https://doi.org/10.2514/6.2023-0483

[4] J.-J. E. Slotine and W. Li, *Applied Nonlinear Control*. Englewood Cliffs: Prentice-Hall, Inc., 1991.