Report on Combined Longitudinal and Lateral Control of a Vehicle

1. Introduction

In this project, I designed and implemented a combined control system for a four-wheeled vehicle's longitudinal (speed) and lateral (yaw rate / path-following) dynamics. My goals were:

1.1 Longitudinal Control:

- Use a PI controller to track a reference speed $v_{x,ref}$.

- Account for tire-slip saturation via the Buck-Hardt model and implement an anti-windup mechanism.

1.2 Lateral Control:

- Implement an MPC (Model Predictive Controller) to track a desired yaw rate $\dot{\psi}_{ref}$ or path.

- Update the linearized bicycle model at each time step based on the current longitudinal velocity v_x.

1.3 Combine these controllers in a full nonlinear Simulink model that includes aerodynamics, rolling resistance, slope forces, and realistic tire slip.

To accomplish these, I developed a set of MATLAB scripts and Simulink blocks, tested them individually, and finally integrated them. I used Montmelo track data as an example for path-following.

2. Vehicle Modeling

2.1 Equations of Motion

The nonlinear vehicle states I focused on are:

v_x : Longitudinal speed (m/s).
$\dot{\psi}$: Yaw rate (rad/s).
β : Vehicle sideslip angle (rad).

The motion is governed by:

$\dot{x}v_x = (F_x - F_d) / m + \beta\, v_x\, \dot{\psi}$

$\ddot{\psi} = (F_{yf}\, l_f - F_{yr}\, l_r) / I_z$

$\dot{\beta} = (F_{yf} + F_{yr}) / (m\, v_x) - \dot{\psi}$

where:

- m is the vehicle mass (kg).

- I_z is the yaw moment of inertia (kg·m²).

- l_f, l_r are distances from the center of gravity to the front/rear axles.
- F_x is the longitudinal force on the front wheels.
- F_{yf}, F_{yr} are the lateral tire forces at the front/rear wheels.
- F_d is the drag force (aero + rolling + slope).

2.2 Aerodynamic and Slope Drag

I modeled drag via:

$$F_d = (1/2)\, \rho\, C_d\, A\, v_x^2 + C_r\, m\, g \cos(\theta) + m\, g \sin(\theta),$$

where θ is the slope angle, ρ is air density, Cd is drag coefficient, and A is the frontal area.

3. Longitudinal Control (PI)

3.1 Controller Design

I employed a PI controller to keep the longitudinal velocity v_x near a reference $v_{x,ref}$. The output of the PI controller is a desired force F_x, which I then saturate to avoid exceeding friction limits. Mathematically:

$$u(t) = Kp\,(v_{x,ref} - v_x) + Ki \int (v_{x,ref} - v_x)\, dt.$$

In my code, I set:

- Kp = 1500,
- Ki = 200,
- Fmax = 9000 N.

I also computed an anti-windup gain Ka via:

$$Ka = 1 / \left((ts/3)\, Ki \right),$$

where ts is my desired settling time in seconds. In the main script:

```
Kp  = 1500;   % Proportional gain
Ki  = 200;    % Integral gain
Fmax= 9000;   % Maximum force (N)
ts  = 0.1;
Ka  = 1/(Ki*ts/3); % Anti-windup gain
```

Report on Combined Longitudinal and Lateral Control of a Vehicle

3.2 Simulink Diagram for PI Control

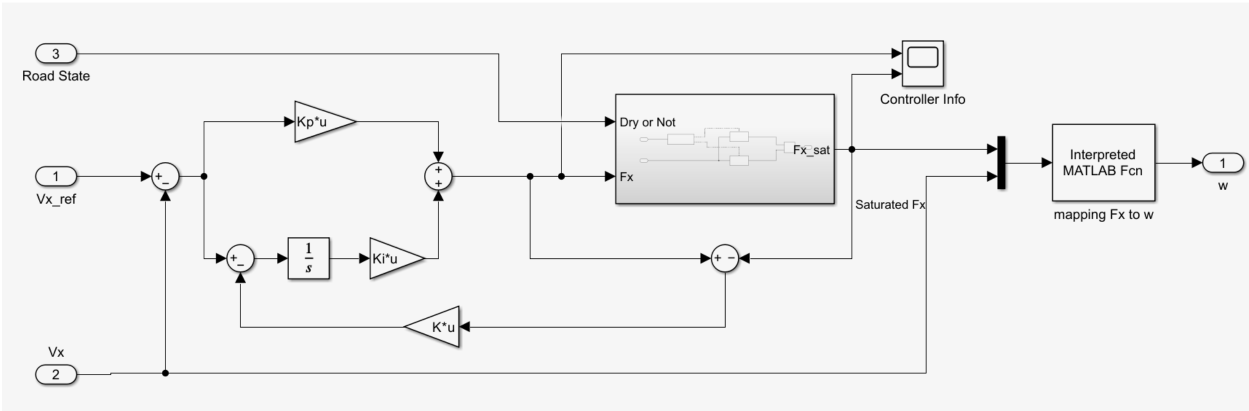


Figure (1) Longitudinal Vehicle control Simulink model

I have a summation block computing the speed error (v_x,ref − v_x). That error goes to:

1. A proportional block (Kp).

2. An integrator (1/s) with Ki.

Then I sum them to get the raw PI output, which is next saturated to ±Fmax.

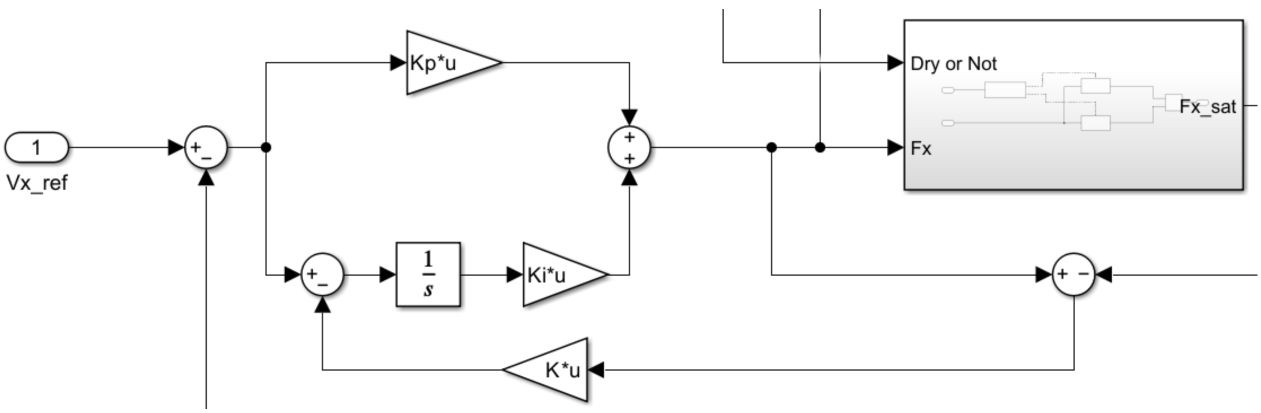


Figure (2) Anti-windup

A feedback path with gain Ka (Anti-Windup) prevents integrator windup. The final saturated force is labeled F_x.

Mapping $F_x \rightarrow \omega$:

This block calls my function Mapping_Fx_to_w.m to get $\omega = f(F_x, v_x)$. The slip ratio definition changes if I am driving ($\omega R \geq v_x$) or braking ($\omega R < v_x$).

3.3 Nonlinear vs. Linear Force Models

Report on Combined Longitudinal and Lateral Control of a Vehicle

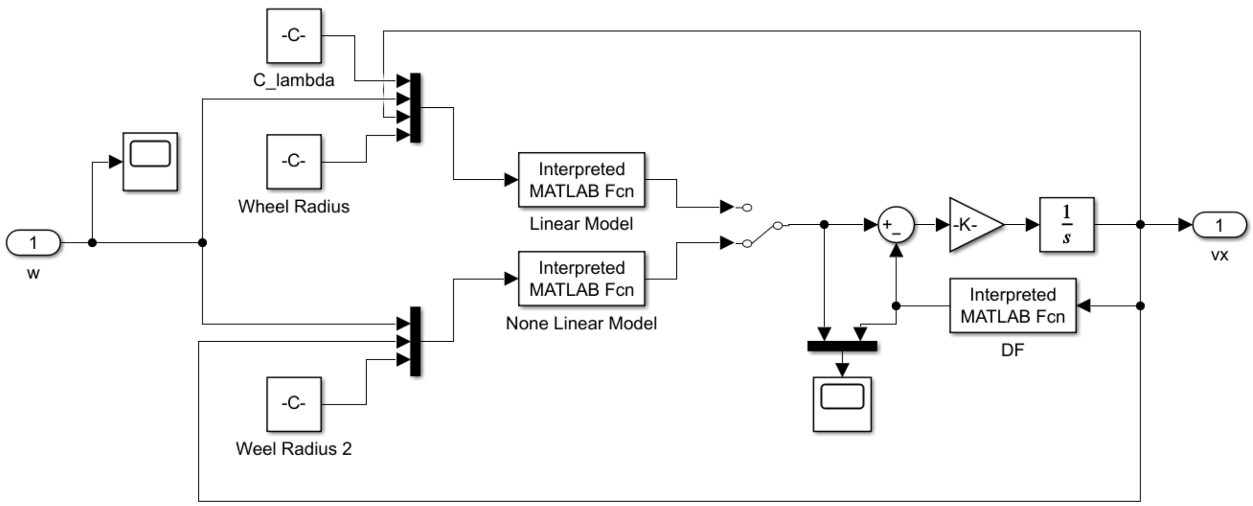


Figure (3) Linear and non-Linear model

In the diagram, I show two paths:

1. The "Linear Model" block uses $F_x = C_\lambda \lambda$.

2. The "Nonlinear Model" block calls a Buck-Hardt–type function with friction coefficient $\mu(\lambda)$.

Ultimately, these feed into the vehicle velocity integrator. In my code, the function longForce(m,R,w,Vx,Dry) checks whether the road is dry or wet to pick different friction parameters.

3.4 Simulation Results for Longitudinal Speed

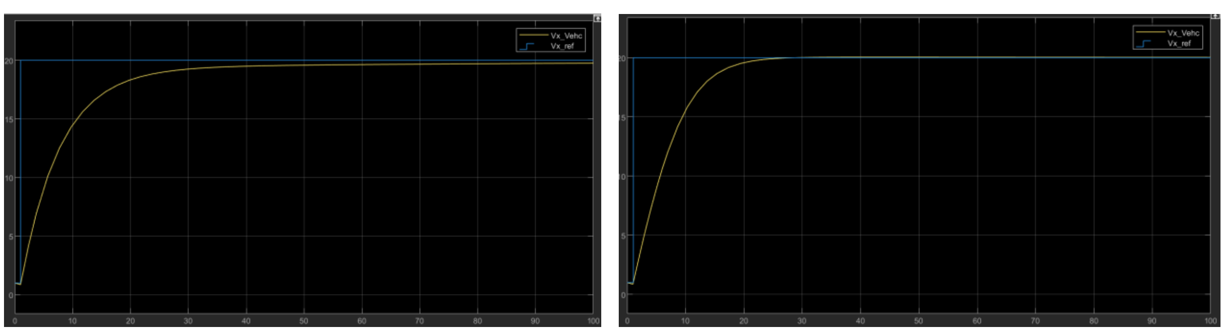


Figure (4) Linear and Nonlinear

I ran a test from v_x=0 to v_x,ref = 20 m/s (72 km/h). The yellow line is the actual vehicle speed (v_x,Vehc), and the blue line is the reference.

- I see the vehicle accelerate within ~10–15 s.
- The final speed asymptotically matches the reference with minimal overshoot.

- The linear model reaches the reference faster (due to ignoring slip), while the nonlinear slip-based model is more realistic.

4. Lateral Control (MPC)

4.1 Rationale

Since the lateral behavior depends heavily on v_x, a single PI is not optimal for all speeds. I instead used an MPC that re-linearizes the bicycle model around the current v_x.

4.2 Model Formulation

A simplified "bicycle model" for yaw rate $\dot{\psi}$ and sideslip β is:

$$\dot{x} = A\,x + B\,u, \quad x = [\ \dot{\psi}\ ;\ \beta\ ], \quad u = \delta.$$

I derive A, B using:

$$A = [\ -(l_f^2\ C_f + lr^2\ C_r)/(I_z\ v_x),\ \ -(l_f\ C_f - lr\ C_r)/I_z$$

$$-1 + (-l_f\ C_f + lr\ C_r)/(m\ v_x^2),\ \ -(C_f + C_r)/(m\ v_x)\ ],$$

$$B = [\ l_f\ C_f\ /\ I_z$$

$$C_f\ /\ (m\ v_x)].$$

I discretize this with a sampling time Ts=0.01 s. My MPC tracks a reference yaw rate $\dot{\psi}$ref and penalizes large steering angles δ. The cost function is:

$$\min \Sigma(\ (\dot{\psi}_{ref} - [1\ \ 0]\ x(k))^2\ Q + \delta(k)^2\ R\ ),\ \ k=1 \text{ to } N,$$

subject to steering constraints $\delta min \le \delta \le \delta max$. I used:

- Q = 10 (weight on yaw-rate error),
- R = 2 (weight on input effort),
- N = 5 (prediction horizon).

4.3 MPC Implementation in Code

In my MPC_Script.m:

1. I create sdpvar states for x(k), input $\delta(k)$, and reference r (which is $\dot{\psi}_{ref}$).

2. I build the cost function and constraints over a horizon N.

3. I define an optimizer object with Yalmip that can be called repeatedly in simulation.

Report on Combined Longitudinal and Lateral Control of a Vehicle

The function MPC_Sim(vx, PsiDot, Beta, PsiDotRef, controller) receives the current states and calls the Yalmip optimizer to solve for δ. Then I feed that δ into the vehicle's lateral dynamics.

5. Combined Control: Longitudinal + Lateral

5.1 High-Level Simulink Layout

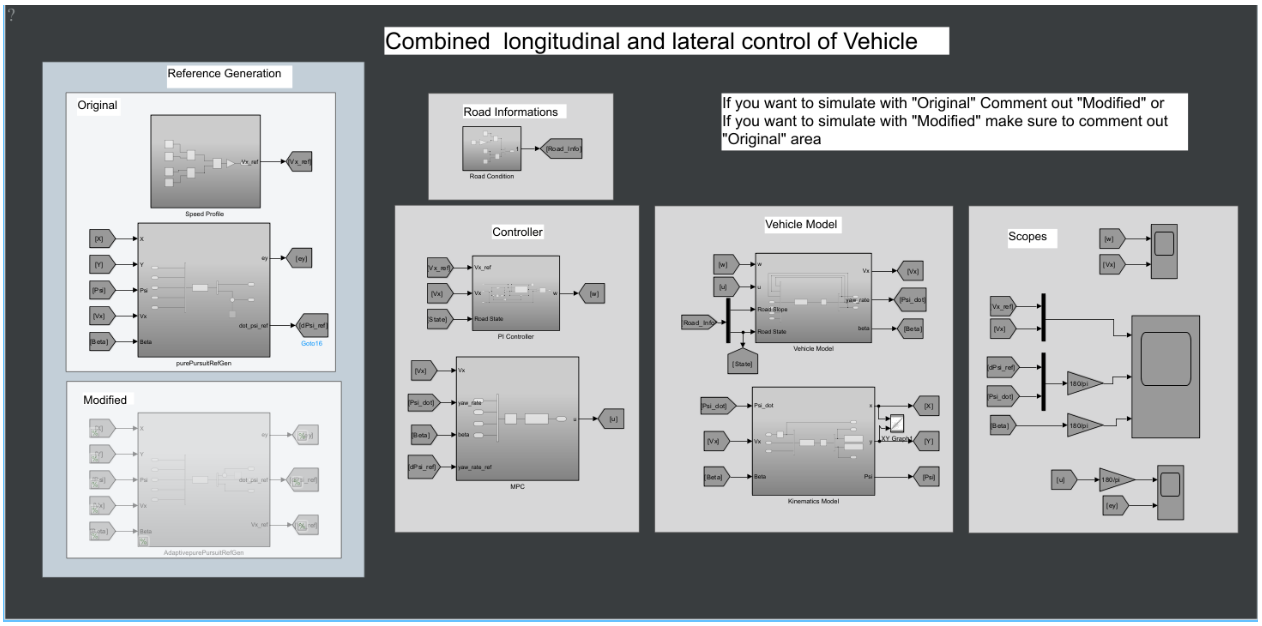


Figure (5) Combined Longitudinal and Lateral control of Vehicle

The block diagram shows:

1. Reference Generation: A block that outputs both speed $v_{x,ref}$ and yaw-rate (or path) references. I have an "Original" and a "Modified" version for pure-pursuit.

2. Road Informations: Let me specify friction (dry/wet) and slope.

3. Controller:

- The PI block as described, plus the MPC block for yaw-rate.

4. Vehicle Model:

- Integrates $\dot{x}$ v_x, $\dot{\beta}$, $\ddot{\psi}$ from the functions in LongDyn.m.
- Also uses a "Kinematic Model" for obtaining global (X,Y).

5. Scopes: Where I collect and compare the signals (v_x, $\dot{\psi}$, β, etc.).

5.2 Scenario 1 Results at 50 km/h with Dry Road

Report on Combined Longitudinal and Lateral Control of a Vehicle

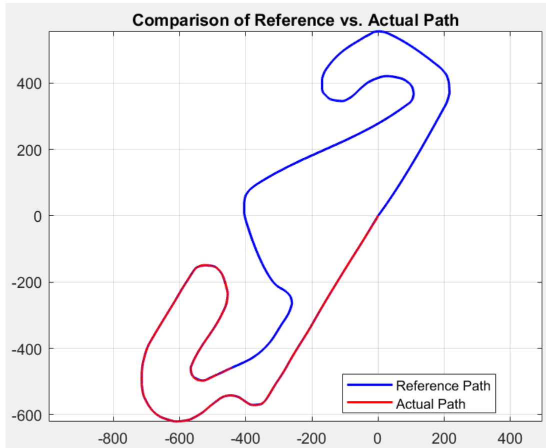
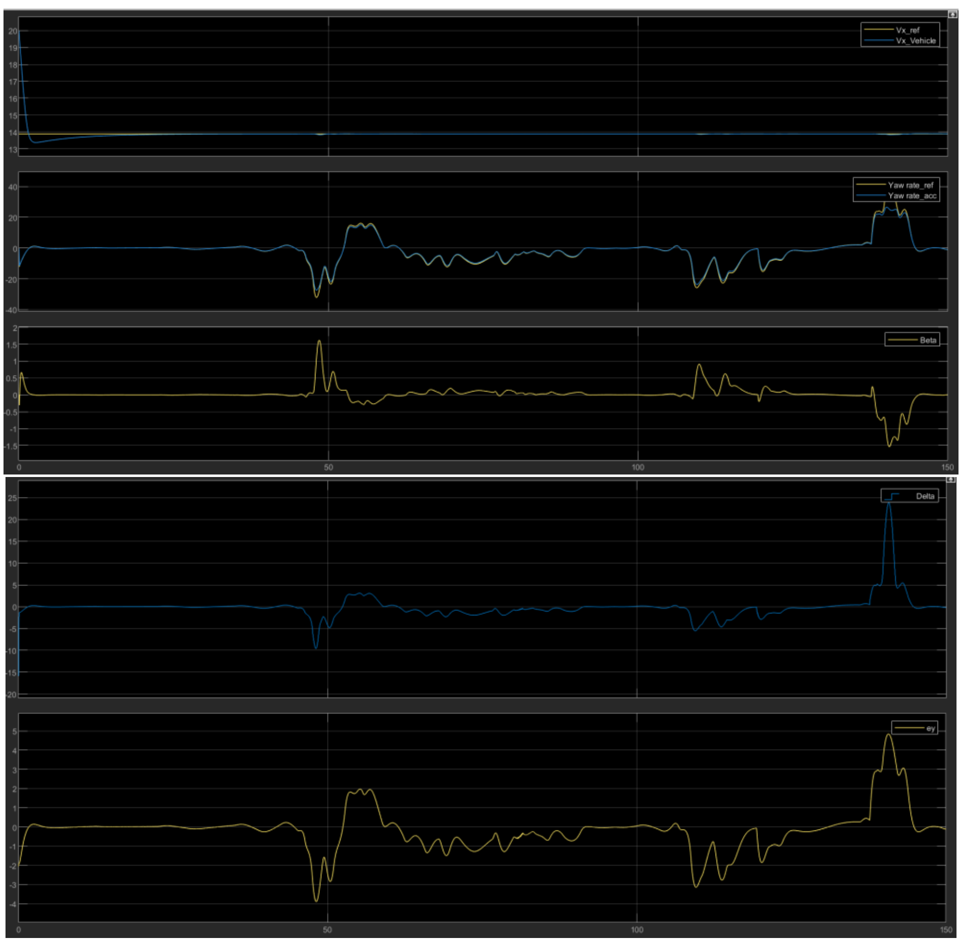


Figure (6) Montmelo track layout 50 km/h with Dry Road.

- The blue line is the reference path from Track.X, Track.Y.
- The red line is the actual path of my vehicle.
- I see some deviation in corners due to tire slip and maybe suboptimal horizon or friction approximations.



Figures (7) & (8) Time-series signals in multiple subplots

Report on Combined Longitudinal and Lateral Control of a Vehicle

1. Top: Speed reference vs. actual speed. At 50 km/h (~13.9 m/s), my PI keeps the speed close to the reference with occasional dips/spikes, likely from turning demands or slight mismatch in friction.

2. Middle: $\dot{\psi}_{ref}$ vs. $\dot{\psi}$ (the yaw rate). The MPC attempts to follow the desired yaw rate. The large negative or positive swings show sharper turns on track.

3. Bottom: The sideslip β, or steering δ, or lateral error ey. These signals confirm that I remain within a small slip angle except during high-speed cornering.

I observe that as soon as the track curves, the yaw-rate reference changes, leading the MPC to adjust δ. The vehicle speed might dip a bit if the corner requires lateral force near the friction limit.

5.3 Scenario 2: 50 km/h, Wet Road, No Slope

Road Condition: Wet 0.5

Slope: θ = 0

Speed Target: 50 km/h (≈ 13.9 m/s)

1. Path Tracking 50kmphr with wet road

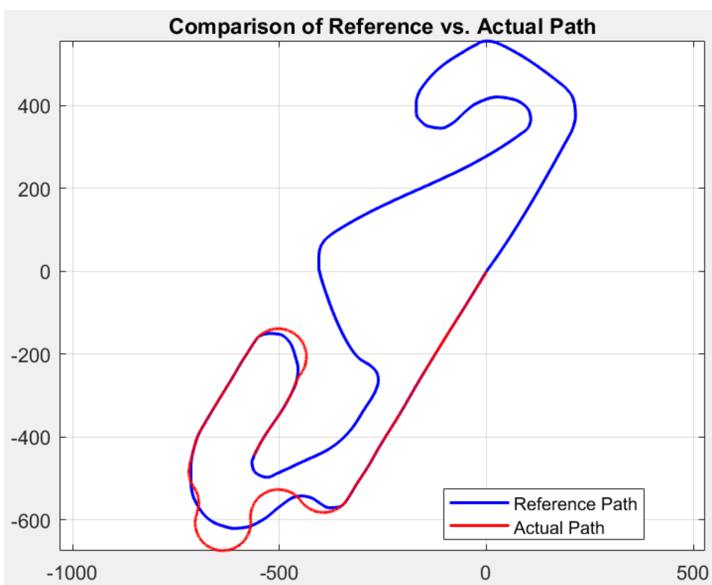


Figure (9) Montmelo track layout 50 km/h with wet Road.

The reference track is shown in blue.

1. The actual path is in red.

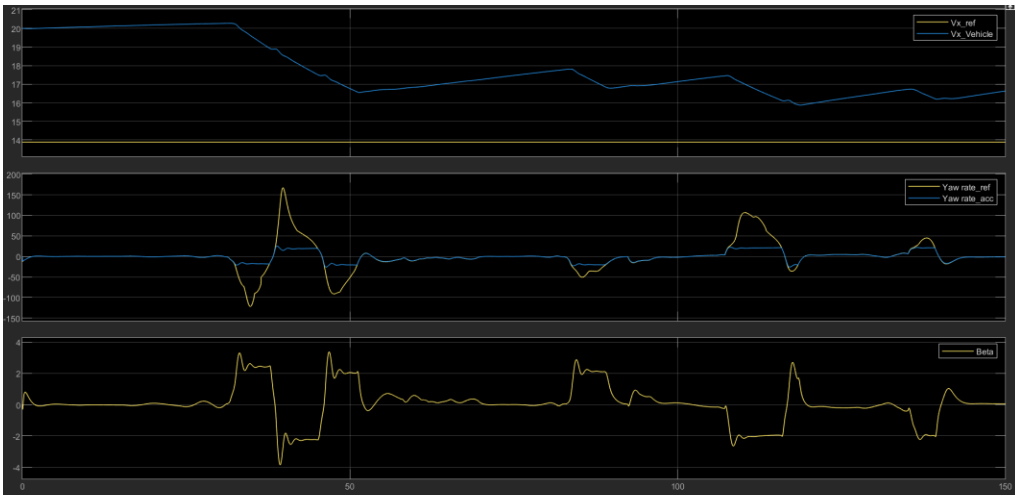
Report on Combined Longitudinal and Lateral Control of a Vehicle

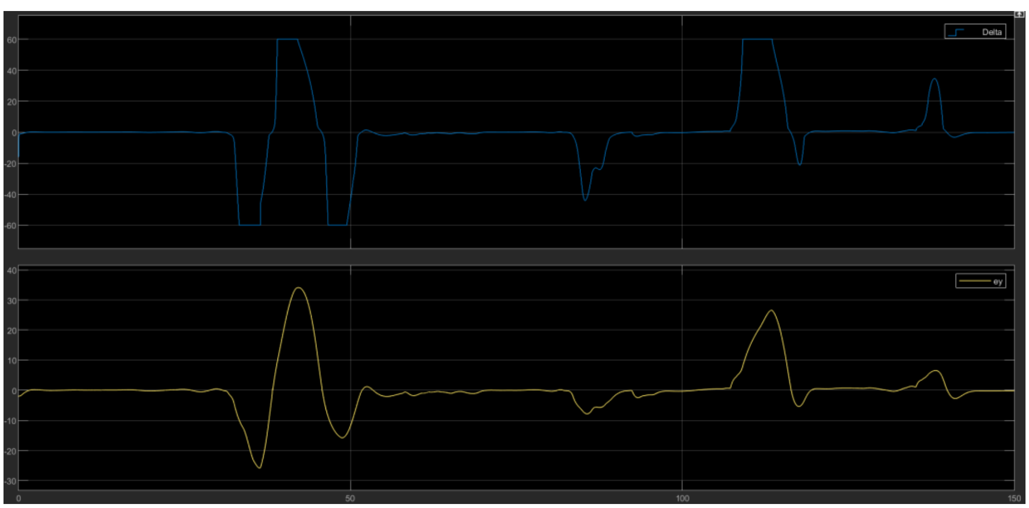
I notice that in wet conditions:

- The friction is lower, meaning the maximum tire force is reduced.
- During corners, the vehicle tends to understeer more because the front tires cannot generate as much lateral force.
- The red path deviates more from the blue path in sharp bends, especially around the first big loop near (−600, −500).

The vehicle still completes the loop, but the path is notably offset.

2. Time-Domain Signals





Figures (10) & (11) Time-series signals in multiple subplots

Looking at the subplots (speed, yaw rate, sideslip, steering):

- **Top plot:** The actual vehicle speed v_x hovers near 13–14 m/s, but can dip if the MPC requires more cornering (increasing front slip ratio).

- **Middle plot:** Yaw rate $\dot{\psi}$ compared to $\dot{\psi}_{ref}$. On a wet road, the yaw rate may lag or saturate due to limited friction, causing spikes/dips as the vehicle enters or exits corners.
- **Bottom plots:** Sideslip β and steering δ tend to be more pronounced in wet conditions, indicating struggles with cornering. The lateral error ey could be larger than in a dry scenario.

Overall, wet friction reduces cornering capability and increases stopping or turning distance, leading to greater path offset from the reference.

5.4 Scenario 3: 50 km/h, Dry Road, 0.1 rad Slope

Road Condition: Dry (μ ≈ 1)

Slope: θ = 0.1 rad (≈ 5.7°)

Speed Target: 50 km/h (≈ 13.9 m/s)

1. Path Tracking The next three images show:

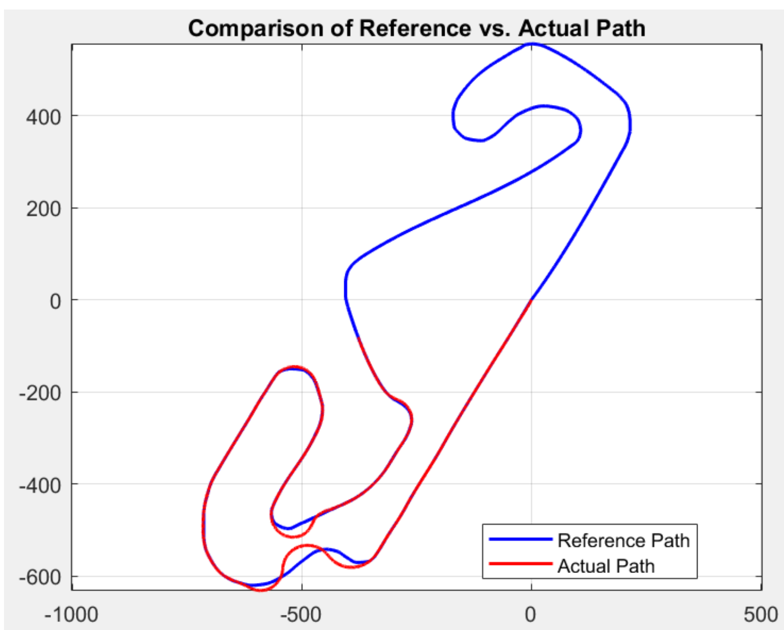


Figure (12) Montmelo track layout 50 km/h with Dry Road and 0.1 rad Slope.

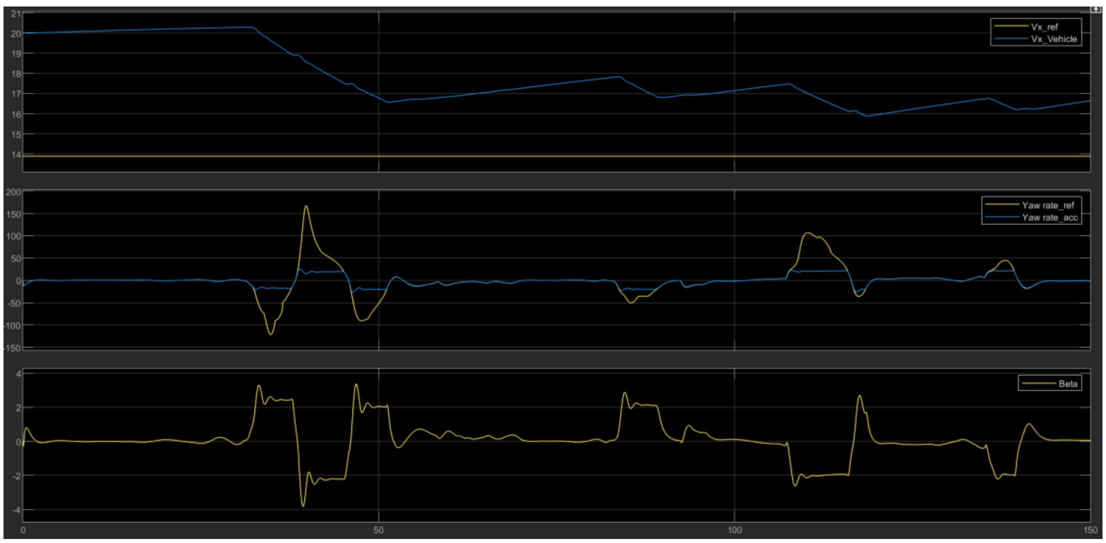
1. Blue line: Reference Montmelo track
2. Red line: Actual vehicle path at 50 km/h with a 0.1 rad slope

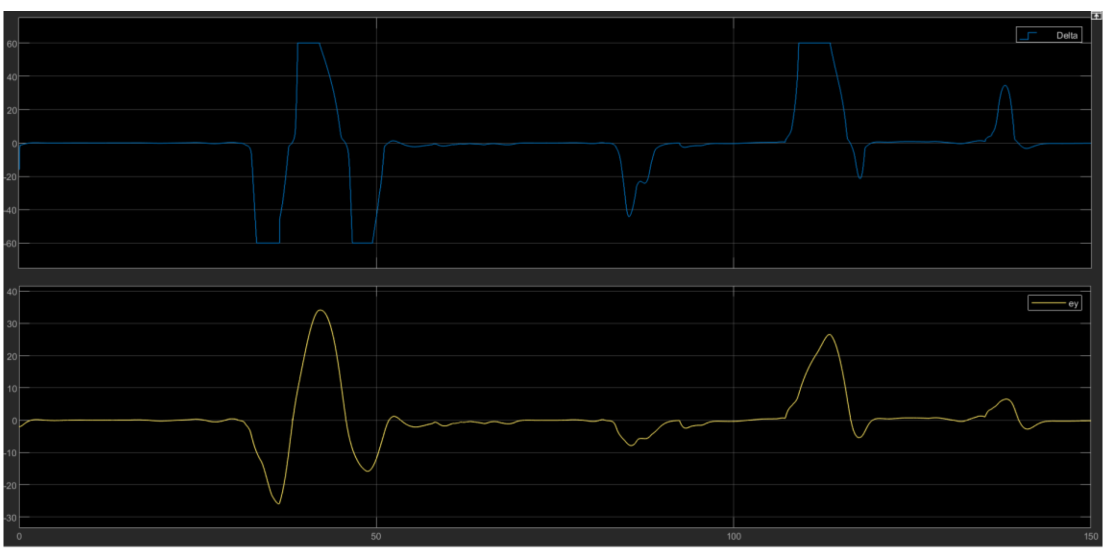
Because the road is dry (μ ≈ 1), there is better grip than under wet conditions, so the lateral offset is smaller. However, the 0.1 rad slope adds ±m g sin(θ) to the longitudinal forces. Even though the slope is mild, the PI controller must apply extra traction going uphill and provide adequate braking downhill to maintain speed.

- **Uphill sections:** The vehicle's speed might dip if the PI doesn't provide enough force quickly.
- **Downhill sections:** Speed can exceed 13.9 m/s if not properly braked.

Overall, the path follows more closely than in the wet scenario, though corners can show overshoot or undershoot due to slope-based load changes.

2. Time-Domain Signals





Figures (13) & (14) Time-series signals in multiple subplots

- **Speed:** Typically, near 13.9 m/s but can vary by ±1 m/s around corners or slope transitions.
- **Yaw Rate:** With better friction, yaw-rate tracking is more accurate than in wet conditions.
- **Beta / Steering:** Slope influences weight distribution, occasionally increasing sideslip or steering effort. On steep segments, "hunting" in speed can occur.

Thus, the dry environment aids lateral tracking, but the 0.1 rad slope challenges longitudinal speed regulation. The final path offset is less than in the wet case.

5.5 Scenario 4: 90 km/h, Dry Road, No Slope

Road Condition: Dry ($\mu \approx 1$)

Slope: $\theta = 0$

Speed Target: 90 km/h (≈ 25 m/s)

This higher speed demands greater lateral force to hold corners, risking sideslip or understeer on tight track segments.

1. Path Tracking 90kmphr with dry road with no slope

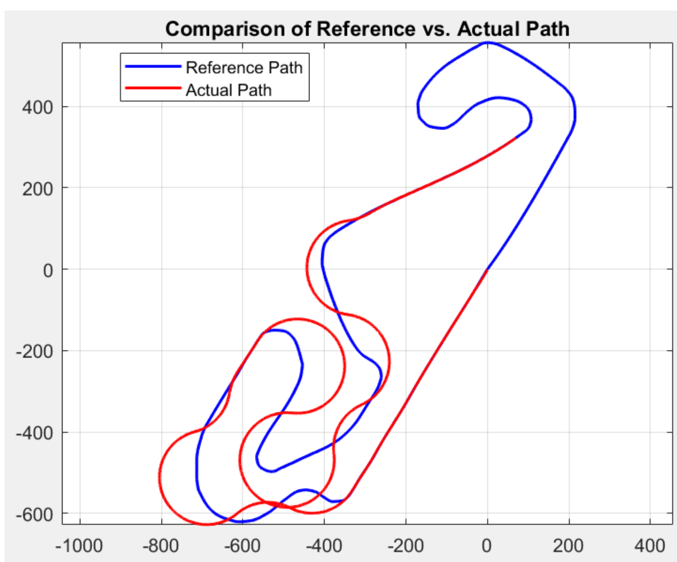
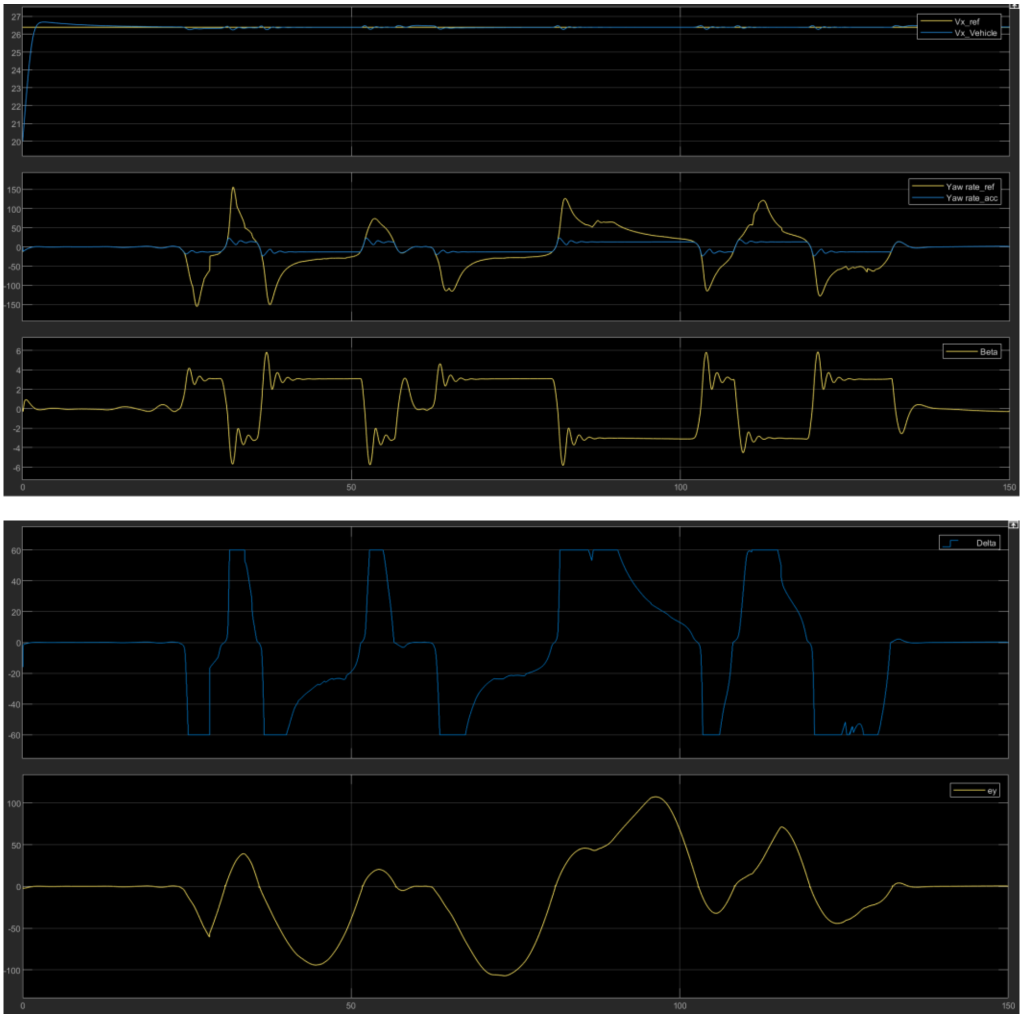


Figure (15) Montmelo track layout 90 km/h, Dry Road, No Slope

- Blue: Reference Montmelo path
- Red: Actual path at 90 km/h

Larger deviations occur at sharper turns because μg may be insufficient for the tight curvature at 25 m/s, forcing corner cuts or overshoots.

2. Time-Domain Signals



Figures (16) & (17) Time-series signals in multiple subplots

- **Speed:** The PI attempts to maintain ~25 m/s, but the speed can dip in corners if tires saturate under lateral load.
- **Yaw Rate:** High reference yaw rates are hard to achieve at 90 km/h without hitting steering or friction limits, causing yaw errors.
- **Sideslip β:** Small steering inputs yield big lateral accelerations at high speed. Once friction saturates, β can surge.
- **Steering δ:** May repeatedly hit ±60° in tight sections.

Hence, at 90 km/h, the system faces friction and linear approximation limits. Using a larger horizon or slip-angle control could improve path tracking.

6. Comparisons and Discussion

1. **Friction Impact**
    - Wet road (μ ≈ 0.5) shows larger path offsets due to reduced tire forces.
    - Dry road (μ ≈ 1) corners better, especially at moderate speeds.
2. **Slope Influence**
    - Even a mild slope (0.1 rad ≈ 5.7°) changes the required force to maintain speed.
    - Uphill needs extra thrust; downhill can cause overspeed if unregulated.
3. **Speed Level**
    - 50 km/h (dry): Reasonable cornering with moderate sideslip.
    - 90 km/h: More frequent friction saturation, leading to large yaw-rate errors and path offset.
4. **Controller Performance**
    - Longitudinal PI: Maintains set speed in most scenarios but faces overshoot/undershoot with slope or friction shifts.
    - Lateral MPC: Good at moderate speeds, hampered by friction limits or linear modeling at extremes.

Adaptive Pure Pursuit Simulations: Results and Discussion

1. **Introduction**

In my simulations, I evaluated an Adaptive Pure Pursuit strategy on the Montmelo track under two primary conditions:

1. Dry Road, No Slope
2. Dry Road, 0.1 rad Slope

I compared speed, yaw rate, sideslip, and path-tracking performance in each scenario.

✓ **Results for Dry Road, No Slope**

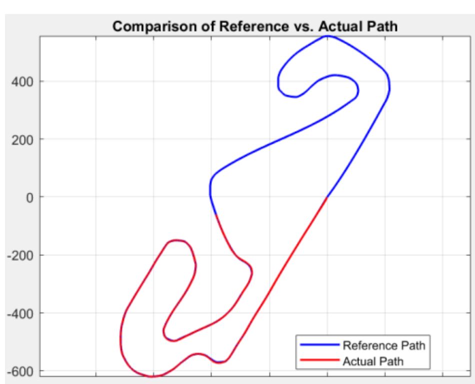
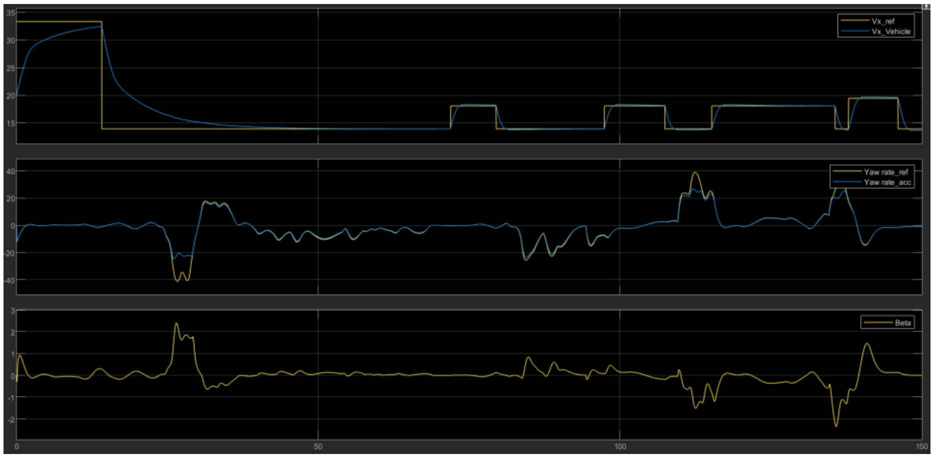
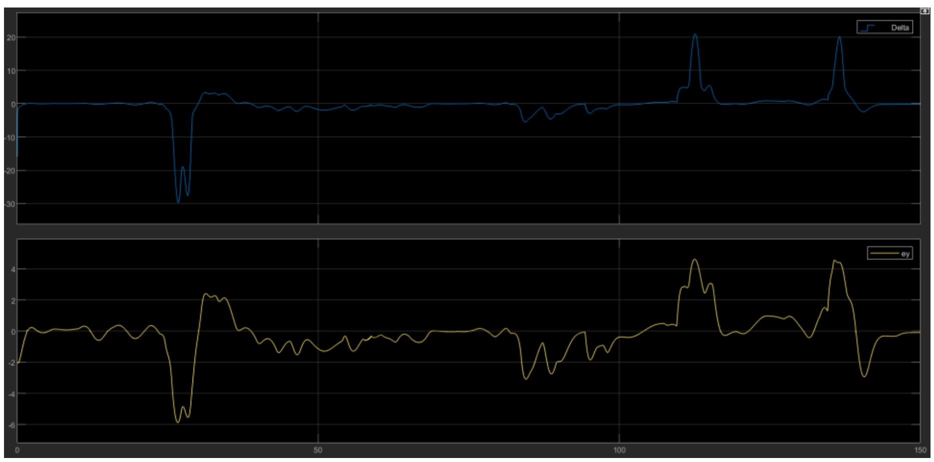


Figure (18) Montmelo track layout for Dry Road and No Slope

Report on Combined Longitudinal and Lateral Control of a Vehicle





Figures (19) & (20) Time-series signals in multiple subplots

1. Speed Tracking

   In the top subplot, I see the reference speed stepping between various values (for example, 20 m/s, 15 m/s) depending on the track segment. My actual vehicle speed attempts to follow these targets. Initially, there can be a transient overshoot or undershoot, but it settles within a few seconds. When the reference speed jumps down, the actual speed takes a short time to decelerate, showing the controller's response time.

2. Yaw Rate

   The middle subplot shows the reference yaw rate compared to the actual yaw rate. On a flat, dry road, friction is relatively high (approximately μ = 1), so the vehicle can achieve moderate to high cornering forces. As a result, the actual yaw rate stays close to the reference except in sharper turns or quick transitions.

3. Sideslip (Beta) and Steering (Delta)

The bottom plots indicate that Beta remains within a modest range, though it can spike when cornering demands or speed changes happen abruptly. Steering angle (Delta) can shift quickly if the path has back-to-back corners, but on no-slope, dry pavement, these variations remain within feasible bounds.

4. Path Comparison

   A plot contrasting the blue reference path with the red actual path reveals that alignment is fairly good on straights and gentle curves. Larger deviations appear in tighter corners, suggesting the reference speed might be optimistic or the curvature transitions are abrupt. Overall, on dry, level ground, the adaptive approach handles the track with minimal path offset apart from these sharp sections.

✓ Results for Dry Road, 0.1 rad Slope

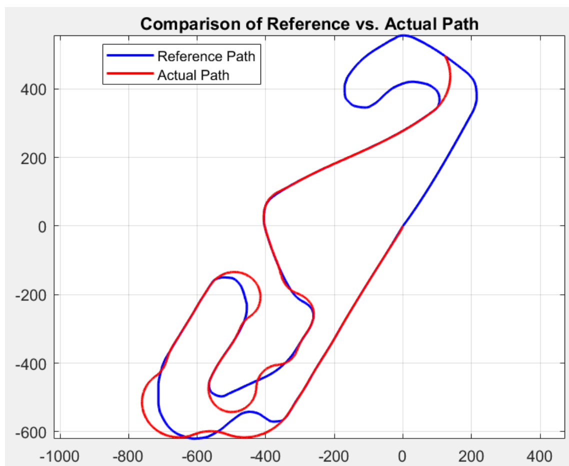
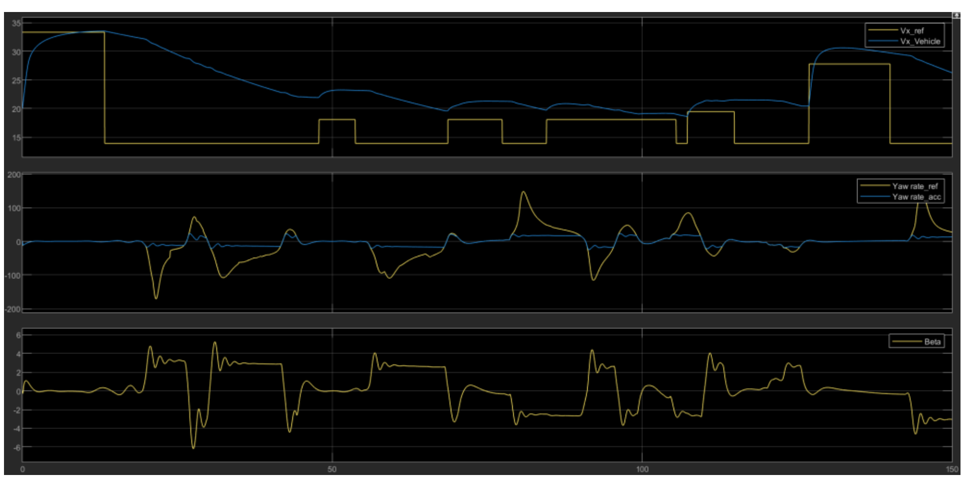
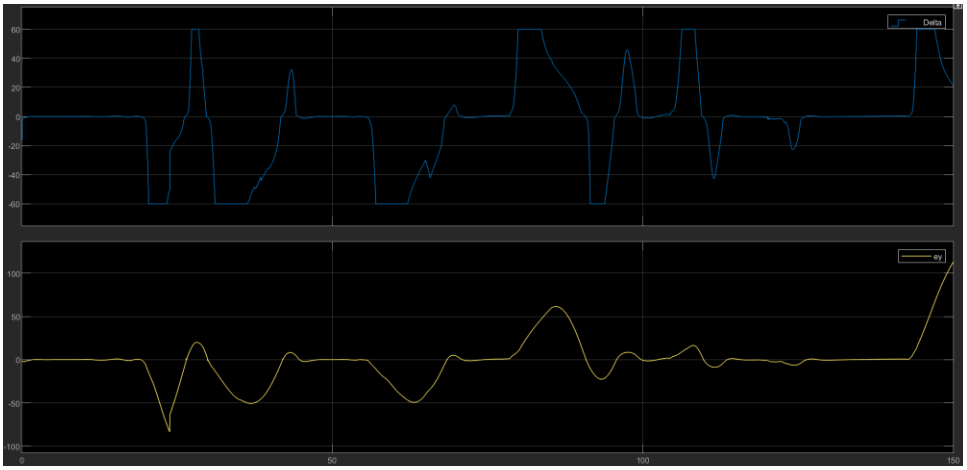


Figure (21) Montmelo track layout for Dry Road and 0.1 rad Slope



Report on Combined Longitudinal and Lateral Control of a Vehicle



Figures (22) & (23) Time-series signals in multiple subplots

When I introduce a slope of 0.1 rad (about 5.7 degrees), speed regulation and yaw performance become more variable:

1. Speed vs. Reference

   The reference speed still steps among different target values for straights and curves. However, on uphill sections, the vehicle's actual speed can lag or take longer to settle, while on downhill sections, it may briefly exceed the reference if the controller does not brake aggressively. These effects are evident whenever the speed command changes and the slope amplifies or counteracts it.

2. Yaw Rate

   Friction remains high since the road is dry, so cornering grip is still sufficient. However, weight distribution changes slightly on uphill and downhill corners, sometimes causing mild understeer (if the front axle is less loaded uphill) or sharper turn-in downhill. As a result, I see minor spikes or dips when slope and corner coincide.

3. Sideslip and Steering

   Beta (sideslip) can fluctuate more near slope transitions, especially if a corner occurs at the same time. Steering angle (Delta) may show bigger swings when slope changes overlap with steering demands, as the system compensates for gravity plus cornering forces. Still, the controller remains stable, and the path is followed reasonably well.

4. Path Alignment

   On the reference vs. actual path figure, the vehicle's path is generally correct but can show bigger deviations in uphill or downhill bends. The slope modifies normal forces, so the existing speed references might not be perfectly optimized for these gradient changes. Nonetheless, I do not see instability or severe drift.

✓ Observations and Conclusions

1. Segmented Velocity Profiles

   Discrete steps in the reference speed help slow the vehicle in corners, though they create step-like transitions that lead to short overshoots or undershoots.

2. Impact of Slope

   Even a modest slope (0.1 rad) significantly alters speed response (lag uphill, surge downhill). Yaw rate remains well-controlled, but corners can demand more steering if slope and turn coincide.

3. Friction and Stability

   Under dry conditions ($\mu \approx 1$), the vehicle maintains stability and moderate sideslip angles. On wet or lower-friction roads, path deviations would likely increase, and further controller adaptation would be required.

4. In both scenarios, the adaptive pure pursuit approach guides the vehicle around the Montmelo track with only moderate deviations. Most issues arise during transitions (speed steps, slope changes, tight corners). A smoother velocity profile or slope-aware feedforward could lessen these transient effects.

❖ Conclusion

In conclusion, I successfully designed and implemented a combined longitudinal and lateral control system for vehicle dynamics. I used a PI controller to achieve accurate tracking of the reference speed with minimal overshoot under various conditions. For lateral control, I implemented a Model Predictive Controller (MPC) that effectively managed yaw rate and path-following dynamics, adapting to changes in speed and road conditions. Through simulations on the Montmelo track, I observed that dry roads and moderate speeds allowed for accurate path tracking, while wet roads and higher speeds introduced challenges such as increased path deviations and friction limits.