Direct Yaw Moment Control for Electric Vehicles

Levi Gershon, Mechanical Engineering
Jesse George-Akpenyi, Mechanical Engineering
Lili Sun, Mechanical Engineering
Massachusetts Institute of Technology

December 10, 2021

1 Abstract

With the advent of commercially feasible electric vehicles, in-tire vehicle motor architectures are becoming more common place. [1] In such designs, the traditional differential axle technology is replaced with a motor, allowing direct torque control of each wheel independently, and thus introducing new degrees of freedom to the system. Among other benefits, such control allows for Direct Yaw Moment Control (DYMC), where differential steering is used to induce a moment, independent of the steering angle. [2] The racecar currently under development by MIT Motorsports, for use in the Formula SAE (FSAE) Electric competition of 2022, is taken as an example case, and a controller is developed. Due to the non-linearity of the system, a sequential Linear Quadratic Regulator (sLQR) is used, along with an Extended Kalman Filter (EKF). [3] Given a set of driver commands, the optimal control is then found, and applied, yielding good control over one lap of the racetrack. This control scheme thus presents a viable method of DYMC for the vehicle, and should provide substantial performance improvements.

Table of Contents

2 Introduction

Vehicle dynamics, as might be expected from the presence of tire mechanics and aerodynamic drag, are highly non-linear. The fame given to talented racing drivers reflects this, and the trajectory planning optimization problem they routinely solve is non-trivial. It is not attempted here, beyond rudimentary motion profiling for testing the controller. Rather, what is needed is a tool to aid the driver. With each wheel's torque independently controllable, the driver has an added ability to apply a yaw moment to the vehicle, and thus to maneuver it as needed. Accordingly, for the foregoing, the desired motion profile becomes the input, and the controller's aim to track it.

The example case considered is that of a lightweight, Formula-1 style electric vehicle designed for use in the FSAE competition. The competition goal is for college students to conceive, design, and fabricate a complete electric vehicle for an autocross-style race. For the summer 2022 race, MIT's team – MIT Motorsports – is, for the first time, implementing a 4-wheel drive architecture, including independent control of the front wheels' torques. Due to the mechanical and electrical challenges of this task, a single rear motor and axle differential are maintained from previous years to minimize the scope of design changes. However, in order to best utilize the new architecture, a controller is needed to supplement the driver and provide DYMC, as described above.

The physical constraints of the system substantially limit both the actuator power, and its control authority. A detailed model of these dynamics is thus first developed, and then used to implement a controller. Due to the non-linearity, and the dependence of actuator control authority on the vehicle's state, the sLQR computes the gains at each time step, using a local linearization. Similarly, the EKF accounts for real system noise – artificially created in this model with MATLAB's pseudo-random noise generation – and these elements are combined to create a realistic model.

3 Modeling

For the purposes of this analysis, the most basic model preserving the desired system dynamics shall be used. Accordingly, longitudinal and lateral load transfer, tire traction curves, suspension compliance, pitch, and roll effects are all ignored, as is longitudinal tire slipping. A modified bicycle model is introduced, where the separation of the front wheels is only considered insofar as needed to calculate the lateral moment created, and then ignored for the remainder of the analysis. In Figure 1, the traditional bicycle model is observed, allowing for steering of the front wheel, but approximating the wheelbase of the vehicle as consisting of only two wheels. [4] [5] [6] The key forces are from the traction and cornering stiffness of each virtual tire, and the aerodynamic drag. Importantly, and substantially different from an actual bicycle, roll dynamics are ignored, and the system is presumed stable.

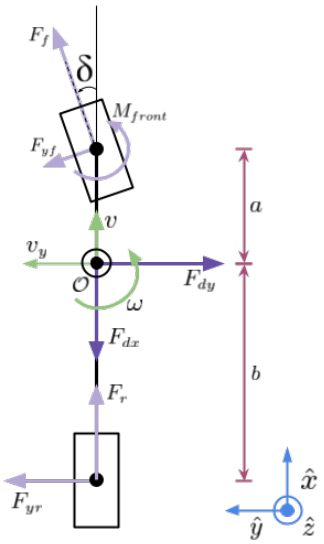

Figure 1: Bicycle model of a motor vehicle. $\hat{x}$ is the longitudinal axis of the car, and $\hat{y}$ the lateral one. v is the longitudinal velocity of the vehicle, v_y is its lateral velocity, ω its yaw rate, and δ the steering angle. It also has longitudinal drag force F_{Dx}, lateral drag force F_{Dy}, rear total traction F_r, forward total traction F_f, and respective front and rear cornering forces F_{yf} and F_{yr}. I_{zz} is the second moment of inertia of the vehicle about its center of mass and $\hat{z}$ axis.

First and foremost, Newton's second law is applied about the vehicle, calculating both forces about $\hat{x}$ and $\hat{y}$, as well as the sum of torques about $\hat{z}$. Note that the time derivative of $\frac{d\vec{p}}{dt}$ is found using the transport theorem, since $\hat{x}$ and $\hat{y}$ form a rotating reference frame.

$$I_{zz} \cdot \dot{\omega} = M_{front} + a(F_{yf}\cos\delta + F_f \sin\delta) - bF_{yr} \tag{1}$$

$$m(\dot{v} - v_y\omega) = F_f\cos\delta + F_{yf}\sin\delta + F_r - F_{Dx} \tag{2}$$

$$m(\dot{v}_y + v\omega) = F_{yf}\cos\delta + F_f\sin\delta + F_{yr} - F_{Dy} \tag{3}$$

The drag forces are calculated with the aid of computational fluid dynamic models conducted on the vehicle's aerodynamics package, with the reported CDA_{long} and CDA_{lat} being the respective drag coefficients multiplied by the areas, and ρ the density of air at sea level and 30 °C. Due to the complexity of modeling it, the moment exerted by the drag during turning is ignored.

$$F_{Dx} = \frac{1}{2}\rho CDA_{long}v^2 \tag{4}$$

$$F_{Dy} = \frac{1}{2}\rho CDA_{lat}v_y^2 \tag{5}$$

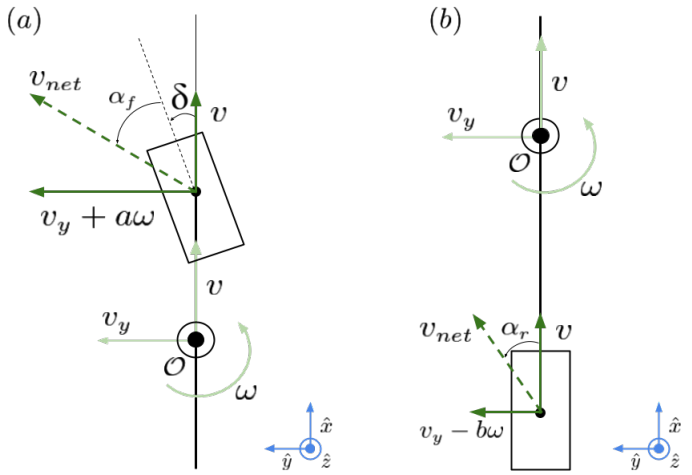

Figure 2: (a) Side-slip angle of the front tire, α_f, noting the inclusion of the car's center of mass rotation to its velocity. (b) Side-slip angle on the rear tire, α_r, again including the car's rigid body rotation.

The slip angle of a tire, as seen in figures 2, is a trigonometric function of the longitudinal, lateral, and angular velocity of the car, and is a measure the degree of slip of the tire relative to its travel.

$$\alpha_f = \arctan\left(\frac{v_y + a\omega}{v}\right) - \delta \tag{6}$$

$$\alpha_r = \arctan\left(\frac{v_y - b\omega}{v}\right) \tag{7}$$

Next, the Society of Automotive Engineers defines the cornering force for small tire slip angles as linearly proportional to this slip angle, such that the lateral forces on the tires can be found via a relativity simple relation with so-called cornering stiffness C. In reality, C shows a strong dependence on the tire normal load, but such mechanics are ignored in this analysis.

$$F_{yf} = -2C\alpha_f \tag{8}$$

$$F_{yr} = -2C\alpha_r \tag{9}$$

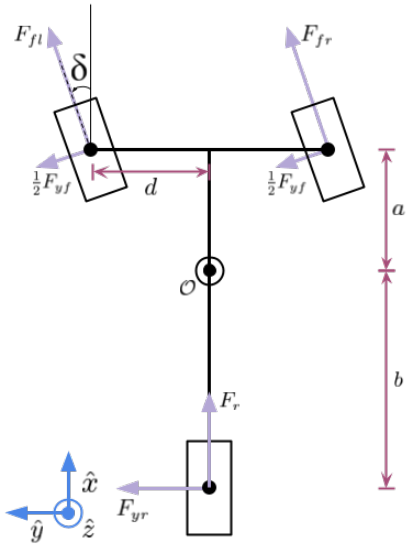

Figure 3: Tricycle model of the vehicle. Used to calculate rolling without slipping velocity conditions, and the yaw moment. F_{fl} is the traction from the front left wheel, and F_{fr} from the front right. The wheels are spaced from the center plane of the car by distance d, and the front wheel is a distance a from the center of mass, while the rear wheel is distance b from the same.

In figure 3, the separation of the front wheels is briefly considered. By inspection, the difference between the front traction forces creates a coupled force pair, hence a moment about the front of the vehicle.

$$M_y = d(F_{fr} - F_{fl}) \tag{10}$$

And assuming the tires to roll without slipping longitudinally, the angular velocity of each wheel is also found. Although the total distance travelled by the wheels is a continuous function of the history of the vehicle and steering angles, instantaneously, its time derivative depends only on the vehicle's velocity and the steering angle. The angle traversed from rest of the front-right, front-left, and rear wheels are ϕ_{fr}, ϕ_{fl}, and ϕ_r, respectively.

$$v = R_w \dot{\phi}_r \tag{11}$$

$$(v + \omega d)\cos\delta + (v_y + \omega a)\sin\delta = R_w \dot{\phi}_{fr} \tag{12}$$

$$(v - \omega d)\cos\delta + (v_y - \omega a)\sin\delta = R_w \dot{\phi}_{fl} \tag{13}$$

These geometric constraint equations are now differentiated with respect to time to find the wheel angular accelerations.

$$\dot{v} = R_w \ddot{\phi}_r \tag{14}$$

$$(\dot{v}+\dot{\omega}d)\cos\delta-(v+\omega d)\dot{\delta}\sin\delta+(\dot{v}_y+\dot{\omega}a)\sin\delta+(v_y+\omega a)\dot{\delta}\cos\delta=R_w\ddot{\phi}_{fr} \quad (15)$$

$$(\dot{v}-\dot{\omega}d)\cos\delta-(v-\omega d)\dot{\delta}\sin\delta+(\dot{v}_y-\dot{\omega}a)\sin\delta+(v_y-\omega a)\dot{\delta}\cos\delta=R_w\ddot{\phi}_{fl} \quad (16)$$

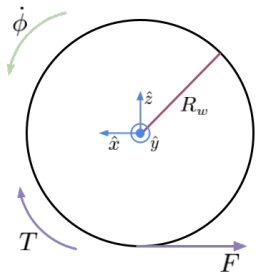

Figure 4: A free body diagram of a wheel of the car. T_{fl}, T_{fr}, and T_r are the individual torques commanded to the front-left, front-right, and rear wheels, respectively. Similarly, ϕ_{fr}, ϕ_{fl}, ϕ_r are the total angular rotations of the front-right, front-left, and rear wheels. I_w is the rotational inertia of the wheel and gearbox, and R_w is the wheel's radius.

Next, considering torque balance about the axles of each of the three wheels, such as that indicated in figure 4 once more using Newton's second law, the motions found above can be related to the forces applied. Rolling resistance is ignored, as are any viscous effects of the bearings supporting the wheel.

$$F_{fr}R_w-T_{fr}=I_w\ddot{\phi}_{fr} \quad (17)$$

$$F_{fl}R_w-T_{fl}=I_w\ddot{\phi}_{fl} \quad (18)$$

$$F_rR_w-2T_r=2I_w\ddot{\phi}_r \quad (19)$$

Setting the two relations for the wheel angular velocities equal, equations (20), (21), and (22) are obtained.

$$\dot{v}=\frac{R_w}{2I_w}(F_rR_w-2T_r) \quad (20)$$

$$(\dot{v}+\dot{\omega}d)\cos\delta-(v+\omega d)\dot{\delta}\sin\delta+(\dot{v}_y+\dot{\omega}a)\sin\delta+(v_y+\omega a)\dot{\delta}\cos\delta=\frac{R_w}{I_w}(F_{fr}R_w-T_{fr}) \quad (21)$$

$$(\dot{v}-\dot{\omega}d)\cos\delta-(v-\omega d)\dot{\delta}\sin\delta+(\dot{v}_y-\dot{\omega}a)\sin\delta+(v_y-\omega a)\dot{\delta}\cos\delta=\frac{R_w}{I_w}(F_{fl}R_w-T_{fl}) \quad (22)$$

These equations can then be re-arranged, using the linear sum and difference of (21) and (22) as seen below.

$$F_r=\frac{2I_w\dot{v}}{{R_w}^2}+\frac{2T_r}{R_w} \quad (23)$$

$$F_f = F_{fr} + F_{fl} = \frac{2I_w}{{R_w}^2}\left[\cos\delta(\dot{v} + v_y\dot{\delta}) + \sin\delta(-v\dot{\delta} + \dot{v}_y)\right] + \frac{T_{fr} + T_{fl}}{R_w} \quad (24)$$

$$F_{fr} - F_{fl} = \frac{2I_w}{{R_w}^2}\left[\cos\delta(\dot{\omega}d + \omega a\dot{\delta}) + \sin\delta(-\omega d\dot{\delta} + \dot{\omega}a)\right] + \frac{T_{fr} - T_{fl}}{R_w} \quad (25)$$

Substituting force definitions into the second law equations found in (1), (2), and (3), the following equations of motion are obtained. Combined with (24), (25), and (23), these form a system of equations linear in $\dot{v}$, $\dot{v}_y$, $\dot{\omega}$, and $\dot{\delta}$.

$$I_{zz} \cdot \dot{\omega} = d(F_{fr} - F_{fl}) + a(F_{fr} + F_{fl})\sin\delta - 2C(a\alpha_f\cos\delta - b\alpha_r) \quad (26)$$

$$m(\dot{v} - v_y\omega) = (F_{fr} + F_{fl})\cos\delta - 2C\alpha_f\sin\delta + F_r \quad (27)$$

$$m(\dot{v}_y + v\omega) = -2C\alpha_f\cos\delta + (F_{fr} + F_{fl})\sin\delta - 2C\alpha_r \quad (28)$$

Beyond this point, the algebra for the non-linear equations of motion becomes truly prohibitive. The full solution is not even worth printing, but is an analytic formula of finite length, and thus amenable to computation on modern processors. Using equations of motion (26), (27), and (28), along with the various results for F_{fl}, F_{fr}, α_f, and α_r, $\dot{v}$, $\dot{v}_y$, and $\dot{\omega}$ were found. This results in a set of non-linear state equations of the functional form below:

$$\frac{d\vec{x}}{dt} = \tilde{\mathrm{f}}(\vec{x}, \vec{u}) \quad (29)$$

Finally, the output equation is trivial, as all reported variables are measurable. The steering angle is easily measured via an encoder, and v, v_y, ω can all be found - albeit with some noise - through the vehicle's on-board Inertial Measurement Unit (IMU).

$$\vec{y} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \vec{x} + 0 \cdot \vec{u} \quad (30)$$

Where the state and input variables are:

$$u = \begin{bmatrix} \dot{\delta} \\ T_{fr} \\ T_{fl} \\ T_r \end{bmatrix}, \; x = \begin{bmatrix} \omega \\ v \\ v_y \\ \delta \end{bmatrix} \quad (31)$$

The numerical vehicle parameters used for the model can be found in table 1, and are from MIT Motorsports' design data. It should be noted that this is a highly non-linear model, and that linearizing about $\vec{x} = 0$ eliminates significant elements of the model, such as the ability of the racecar to drift.

Table 1: Model Constants

Variable	Value
a	$0.762\,\mathrm{m}$
b	$0.762\,\mathrm{m}$
d	$0.610\,\mathrm{m}$
R_w	$0.203\,\mathrm{m}$
C	$24.293\,\frac{\mathrm{N}}{\mathrm{rad}}$
I_{zz}	$180.5\,\mathrm{kg}\cdot\mathrm{m}^2$
I_w	$1.6\,\mathrm{kg}\cdot\mathrm{m}^2$
ρ [8]	$1.145\,\frac{\mathrm{kg}}{\mathrm{m}^3}$
CDA_{long}	$1.64\,\mathrm{m}^2$
CDA_{lat}	$1.72\,\mathrm{m}^2$

4 Dynamic Linearization

Accordingly, instead of using a static regulator, the model shall be updated dynamically, with the Jacobian approximation of the non-linear model calculated sequentially at each point. For each $(t, \vec{x}, \vec{u})$, we can analytically find the first order derivative along each component, hence the Jacobian, $A(\vec{x}, \vec{u})$ and $B(\vec{x}, \vec{u})$.

$$A(\vec{x}, \vec{u}) = \left.\frac{\partial \vec{f}}{\partial \vec{x}}\right|_{\vec{x},\vec{u}} \tag{32}$$

$$B(\vec{x}, \vec{u}) = \left.\frac{\partial \vec{f}}{\partial \vec{u}}\right|_{\vec{x},\vec{u}} \tag{33}$$

The Jacobians then provide the state space approximation of equation (34). Note that this is not used for the actual integration of the dynamics; for that, $\vec{f}(\vec{x}, \vec{u})$ is used, and provides the input to the feedback of the system. Rather, the linearization is only to determine system gains and to enable the EKF.

$$\frac{d\vec{x}}{dt} = A(t)\vec{x} + B(t)\vec{u} \tag{34}$$

5 Controller

Were this system linear, by the separation principle, we could design the controller independently of the observer, assuming full state feedback were available. Said feedback is available, and so we shall use a control law as in the linear case, understanding some revision may be necessary to move the closed-loop poles of the augmented observed system due to the non-linearities.

$$\vec{u} = K(\vec{r} - \vec{x}) \tag{35}$$

The gain matrix $K(\vec{x(t)}, \vec{u(t)})$ is found via optimizing the linear quadratic regulator cost function, equation (36), subject to the approximation of equation (37). The factor of $\frac{1}{2}$ on the estimated point for A and B is to account for the non-zero set-point of the system, and to center the approximation over the span of $\frac{d\vec{x}}{dt}$ it must cover.

$$V = \int_0^\infty \left(\vec{x}^T Q \vec{x} + \vec{u}^T R \vec{u}\right)\, dt \tag{36}$$

$$\frac{d\vec{x}}{dt} = A\left(\frac{\vec{x(t)}}{2}, \vec{u}\right)\vec{x} + B\left(\frac{\vec{x(t)}}{2}, \vec{u}\right)\vec{u} \tag{37}$$

Somewhat arbitrarily, the design requirement is set as a maximum error in angular velocity ω and steering angle δ of $\Delta\omega_{max} = 0.05\,\text{rad}$. Since the velocity needs only coarser control, the maximum error is set as $\Delta v_{max} 0.5\,\text{m/s}$. These are then adjusted by numerical experimentation, the results of which are seen below.

$$Q = \begin{bmatrix} \frac{10}{(\Delta\omega_{max})^2} & 0 & 0 & 0 \\ 0 & \frac{5}{(\Delta v_{max})^2} & 0 & 0 \\ 0 & 0 & \frac{5}{(\Delta v_{max})^2} & 0 \\ 0 & 0 & 0 & \frac{1}{(\Delta\omega_{max})^2} \end{bmatrix} \tag{38}$$

More strictly, the control weight costs are determined based on the physical limits of the system. The forward wheels have a tractive limit of $T_{f,max} = 126\,\text{N}\cdot\text{m}$, and the rear wheels of $T_{r,max} = 368\,\text{N}\cdot\text{m}$. Due to mechanical constraints, the steering maximum angle is $\delta_{max} = 30°$. These values were then used as a starting point for the actuation cost matrix, with the constants indicated added in order to meet the aforementioned design requirements and checked via numerical exploration.

$$R = 2\cdot 10^3 \begin{bmatrix} \frac{1}{\delta_{max}^2} & 0 & 0 & 0 \\ 0 & \frac{1}{T_{f,max}^2} & 0 & 0 \\ 0 & 0 & \frac{1}{T_{f,max}^2} & 0 \\ 0 & 0 & 0 & \frac{100}{T_{r,max}^2} \end{bmatrix} \tag{39}$$

6 Filtering Observer

Although all states are directly observable via sensors, a Kalman filter is still necessary due to to output noise and input disturbances. However, because of the non-linearity of the system,a similar approach must be taken for the observer to that done with the controller. As explained above, the system is dynamically linearized by taking the Jacobian at each time step in order to find

the controller gains; the same is done for the observer gains. This method forms part of the Extended Kalman Filter (EKF). The EKF approximately follows the same process as a regular Kalman filter, except that the state transition matrix is replaced by the dynamic Jacobian. For the Kalman Filter, we find the gain that minimizes the error covariance, defined as in equation (40).

$$\mathbf{\Sigma}(t) = E\{\tilde{\mathbf{x}}(t)\tilde{\mathbf{x}}(t)^T\}. \tag{40}$$

In the regular Kalman Filter, its time evolution is given by equation (41).

$$\dot{\mathbf{\Sigma}} = (\mathbf{A} - \mathbf{LC})\mathbf{\Sigma} + \mathbf{\Sigma}(\mathbf{A} - \mathbf{LC})^T + \mathbf{W_c}. \tag{41}$$

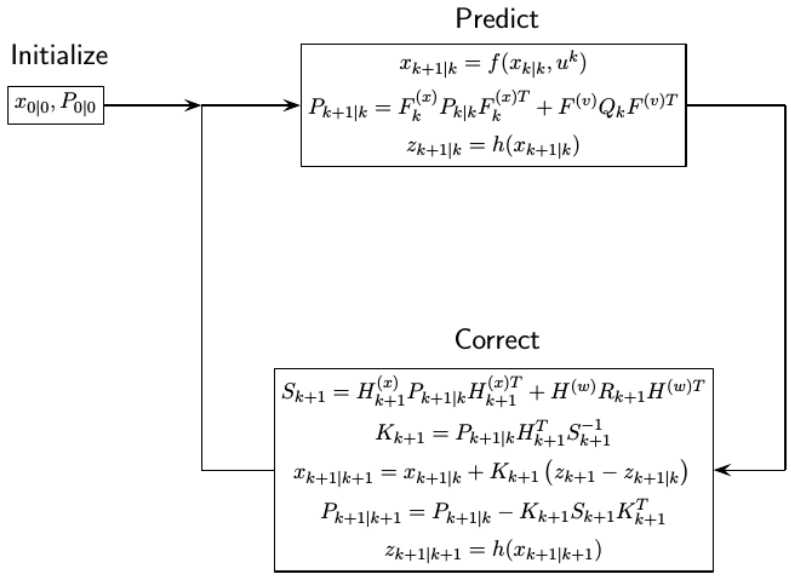

Figure 5: Extended Kalman Filter Block Diagram [9]

With the EKF, the aim is to lose as little accuracy to the linearization as possible, so for the prediction step $f(x_{k|k}, u^k)$ is used, where f is the nonlinear relation from $(x[k1], u[k1])$ to $x[k+1]$. To update the covariance *a priori* $(P_{k+1|k})$, the Jacobian matrices with respect to the state $(F_k^{(x)})$ and with respect to the noise $(F^{(v)})$ are used. There is an additional correction step after the prediction, where the latest z_k measurement is incorporated. In the predict step K is the Kalman gain, and H replaces C as the transition matrix. Once again, H is the Jacobian of $h(x)$, which maps x to the output z. On the basis of common hardware performance, a signal to noise ratio of 100 was assumed, for both the input disturbance and the output noise.

7 Motion Profiling

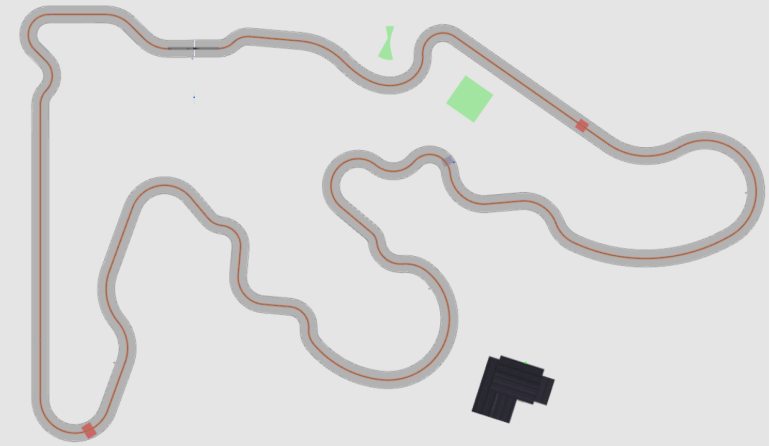

Figure 6: Hockenheim 2010 Formula Student Germany endurance track, aerial view

A bird's eye image of a representative track was found, along with its length - see figure 6. [2]. This was the course used in the Hockenheim 2010 Formula Student Germany competition, and each lap was 774 m in length. [2]. First, it was converted, using image recognition techniques, (see "TrackReader.m") to a sequence of (x, y) points for the car to follow.

These resulting points were then fed into a symmetric moving average filter twice to create a smoother trajectory with gradually changing finite differences. Next, the total distance of the trajectory was calculated assuming a linear interpolation. Finally, a time series was generated for the angular, longitudinal, and transverse velocities of the vehicle, as well as its steering angle. This was possible in part due to several simplifying assumptions about the velocity and orientation of the vehicle along the track.

It was assumed that the transverse velocity would be sufficiently low during travel that it could be reasonably neglected. Steering angle was assumed to be the difference between a desired angle, taken to be the slope of the track at the car's position, and the car's orientation, which could be calculated from the cumulative sum of the angular velocity. The angular velocity of the vehicle was approximated by applying a moving average to the slope of the generated track trajectory, spreading out the angle change over longer periods of time to represent a lag in the car's orientation with respect to the path. The longitudinal velocity represented was assumed to be near constant in magnitude at a chosen maximum speed during travel and equal to the net velocity, with fixed acceleration and deceleration rates at the beginning and end of the path.

To verify the competency of the motion profiling algorithm, the smoothed data profiles were reconstructed using calculated linear velocity and trajectory

slope data, then reconstructed using only the velocity information. While there is some positional drift over the length of the lap, it is small compared to the length of the lap, and is acceptable for the purposes of this paper.

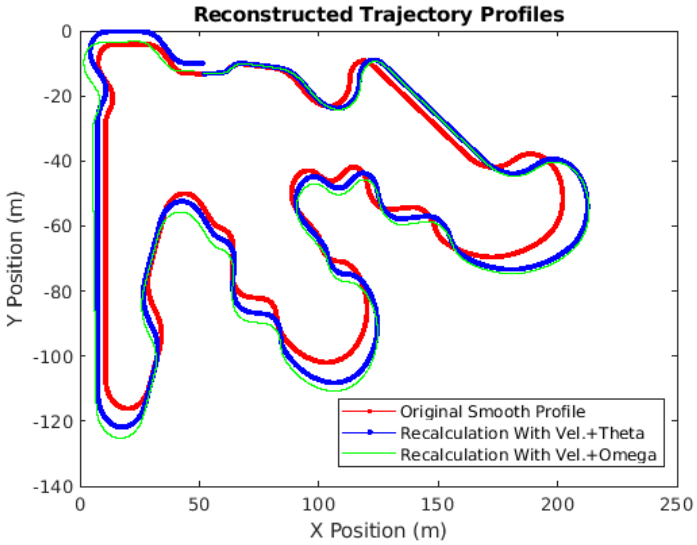

Figure 7: Reconstructed trajectory profiles of the Hockenheim track from Fig. 6. Theta is the angle of the track trajectory and Omega represents the angular velocity of the car.

8 Performance & Robustness

Tracking over the generated motion profile, the controller performs well, despite a 1% noise and input disturbance to signal ratio, as well as the adjusted set point of the controller. Figure 8 shows the degree of noise in the system prior to filtering: this system looks nigh unstable, and does not appear to be in good control. However, this is an artifact of the measurements, and figure 9 clearly shows better track of v, ω, and δ to the set point.

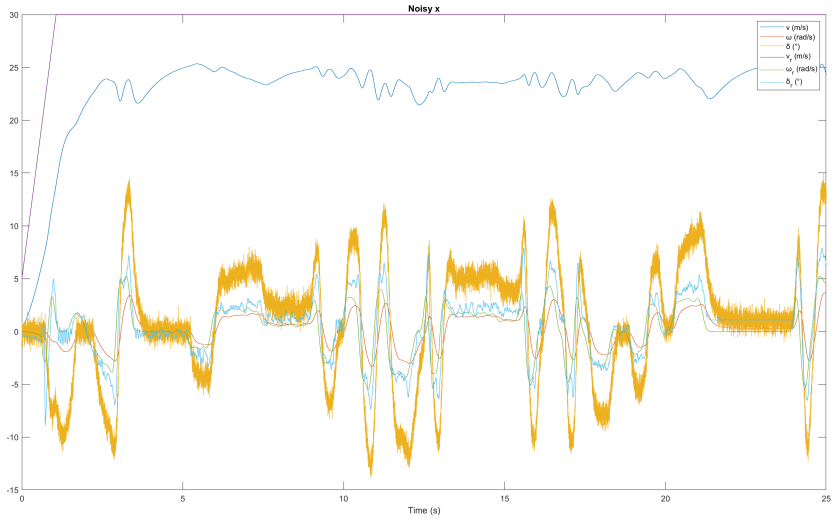

Figure 8: Unfiltered State vs Time. There is substantial noise in the ω term. Even without the filtering observer, the controller tracks the desired trajectory decently.

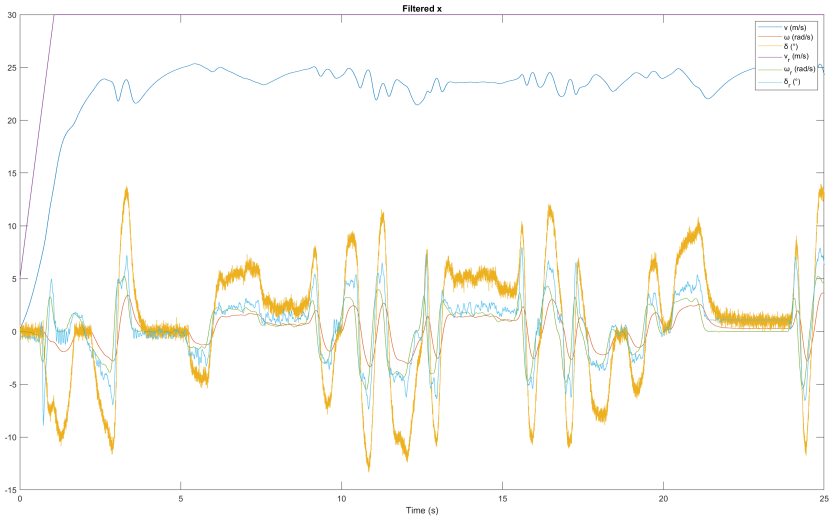

Figure 9: Filtered State vs Time. Comparing this to Fig. 8, the amplitude of noise in ω has decreased. The effect on the other state variables is less apparent, but also of lesser import for a yaw controller.

Importantly, too, the commanded inputs remain well within the required bounds of the design constraint problem.

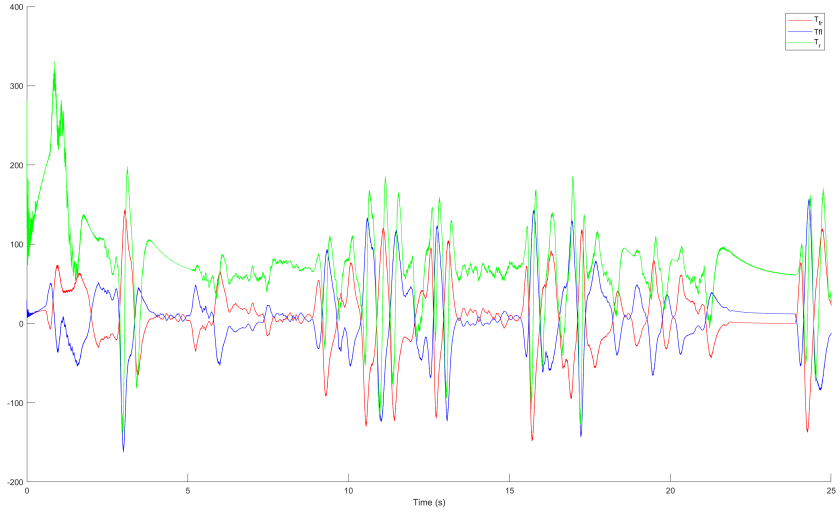

Figure 10: Actuator Input vs Time. The maximum torque on the rear "tire" is approximately 325 Nm, which is within the capability of the rear motor. Similarly, the torque inputs for both the front motors stay within their limits.

Additionally, while the errors exceed the arbitrary bounds provided in the controller section, they are physically reasonable, and could be refined with further work and better noise filtering, as seen in figure 11.

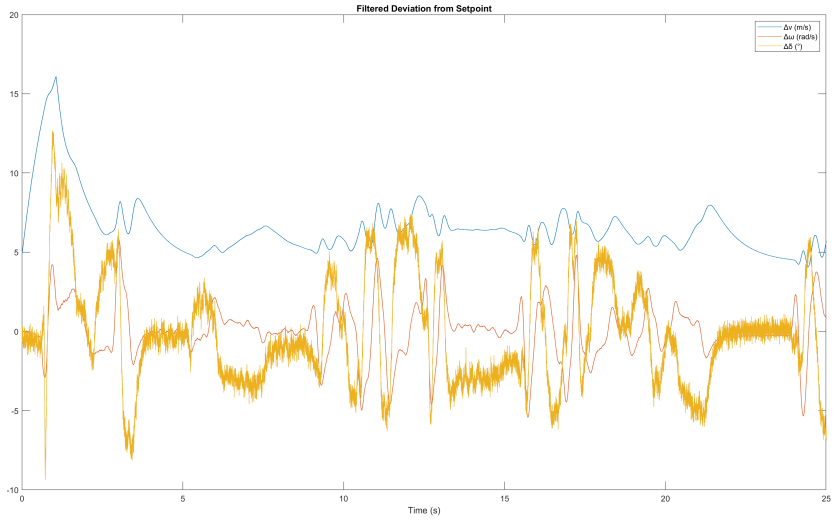

Figure 11: The deviations of the tracked state variables, as calculated from the observer, from the set point are shown.

Finally, as a note for future work, the tracking of v_y is shown in figure 12. As might be expected, turning induces some strafe, though here the controller attempted to minimize it. However, even increasing the weight in the Q matrix of the v_y error by a factor of 10^6 does not decrease the magnitude of the error by more than 50%. Clearly, then, the vehicle has poor control authority over v_y, as expected from the non-holonomic constraints and low steering angles assumed over this racetrack. That said, v_y also depends heavily on tire traction dynamics not modeled herein, and so this is observed as a weakness of the model used and an opportunity for future work.

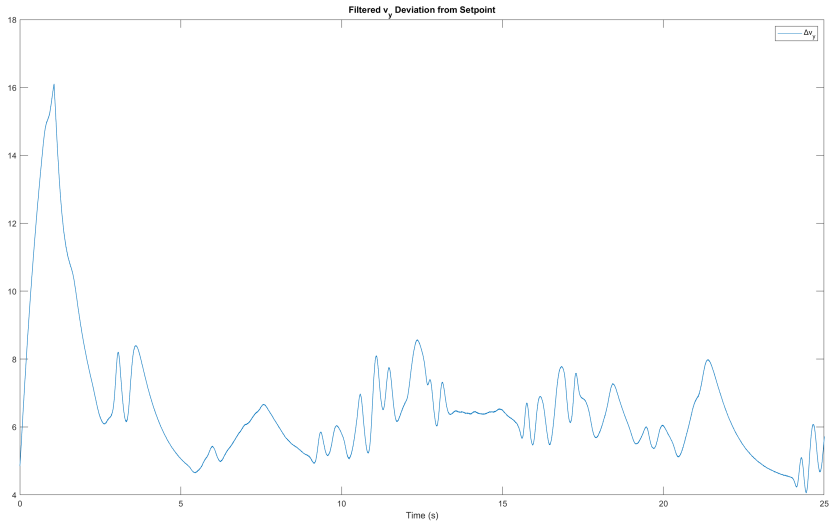

Figure 12: Plot of v_y over time during the course. The set point was zero for all times, hence this represents an error on the controller.

9 Conclusion

As seen in the results section, the sLQR developed here provides a passable form of DYMC, using the differential steering torque to accelerate turning. With the aid of the EKF, it is seen to be robust to system noise and disturbances. However, it has poor control over the system strafe, and lacks the tire slip mechanics that complicate vehicle traction distribution during acceleration. Nevertheless, it provides a basis for control, and an opportunity to implement better motion planning algorithms to enable the human driver.

Indeed, that human-machine interaction is a complicated one: the driver will need to develop intuition for the system response to wheel and throttle commands, using their biologic neural net, so to speak. In turn, the vehicle control unit must anticipate the driver and enable their desired vehicle behaviour, all the while maintaining ergonomic control dynamics. Such a problem is well beyond the scope of this paper, but would be enabled by the sLQR developed herein as a viable sub-component.

Furthermore, this paper presents a model for how electric vehicle control can be approached, even generalizing to the higher degree of freedom, four-wheel independent control architecture. Through the construction of a similar hybrid multi-wheel/bicycle model, the system dynamics could again be rendered tractable, and an sLQR created to control them given a motion planner.

References

[1] Earth Day 2021 Arrives as U.S. Electric Vehicle Sales Continue to Grow" 21-Apr-2021. https://www.bts.gov/data-spotlight/electric-vehicle-use-grows

[2] A. Medina, G. Bistue, and A. Rubio, "Comparison of typical controllers for direct yaw moment control applied on an electric race car," Vehicles, vol. 3, no. 1, pp. 127–144, 2021.

[3] T. DeWolf, "Linear-quadratic regulation for non-linear systems using finite differences," 10-Nov-2015. https://studywolf.wordpress.com/2015/11/10/linear-quadratic-regulation-for-non-linear-systems-using-finite-differences/

[4] A. Mangal and B. Fabien, "A vehicle dynamics model for torque vectoring during a turn," thesis, 2018.

[5] Bicycle Model. http://code.eng.buffalo.edu/dat/sites/model/bicycle.html

[6] Y. Ding, "Simple Understanding of Kinematic Bicycle Model," Medium, 15-Feb-2020. https://dingyan89.medium.com/simple-understanding-of-kinematic-bicycle-model-81cac6420357

[7] G. S. Vorotovic, B. B. Rakicevic , S. R. Mitic, and D. D. Stamenkovic, "Determination of Cornering Stiffness Through Integration of A Mathematical Model and Real Vehicle Exploitation Parameters ," FME Transactions, vol. 41, no. 1, pp. 66–71, Jan. 2013.

[8] Engineering ToolBox, (2003). Air - Density, Specific Weight and Thermal Expansion Coefficient vs. Temperature and Pressure. https://www.engineeringtoolbox.com/air-density-specific-weight-d˙600.html

[9] "Extended Kalman Filters," MATLAB amp; Simulink, Sep-2021. https://www.mathworks.com/help/driving/ug/extended-kalman-filters.html