The torque vectoring system (TVS) is a barebones torque vectoring and traction control system. To most efficiently utilize the grip of the vehicle, it is necessary to operate at the limits of grip for each tire. As such, traction control naturally arose as a consideration. To test the TVS, Simulink was employed. Below is the high-level architecture of the TVS in loop with an approximate model for PER22.

Diagram

Description automatically generated with medium confidence

In this diagram, TVS is shown to be composed of 6 systems, which outputs Torque Request (Tx).

1. Steering Model
2. Tire Model
3. Brake Model
4. PI Controller
5. Constraint Generation
6. Optimization

## Steering Model

The steering model takes the angle of the steering wheel as an input, and outputs 3 terms. The angle of each front tire, and the rack displacement of the steering system. For PER22, reverse-Ackermann steering is used. This is a non-linear steering pattern. This means that the left and right steering angles are not equal, particularly for large rack displacements. Below is the system diagram of the model.

A picture containing graphical user interface

Description automatically generated

Figure []: System Diagram for Steering Model

The picture below shows how reverse-Ackermann steering differs from other steering setups. At low speeds, Ackermann steering is advantageous because the tire angles conform to the speed difference between the inner and outer front tires. The inner tire is closer to the instantaneous center of rotation, which means it requires a larger tire angle to allow all tires to share the same instantaneous center of rotation. When this occurs, all tires can freely roll without sliding. At high speeds, reverse Ackermann steering is advantageous because of large tire deformation. When cornering at high speeds, the outside tire sustains much higher loads than the inside tire. As such, the outside tire deforms more than the inside tire. When the cornering is aggressive enough, the tire deformation is such that the tire contact patches of the tires resemble Ackermann steering. The severity of the Ackermann or reverse Ackermann steering can be seen as a continuum, where parallel steering in an example intermediatory.

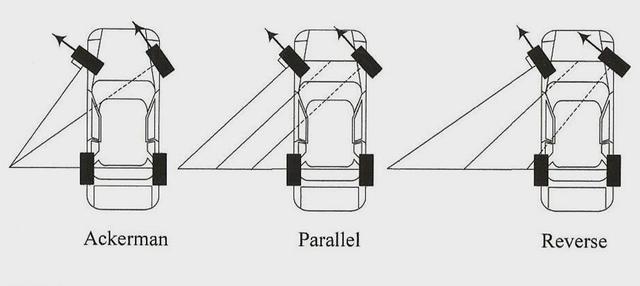


Figure []: Different Steering Setups

The physical system is shown below. The steering wheel is connected 1:1 to the input of the perpendicular shaft gearing. The gearing converts the steering wheel rotation into a linear displacement. As shown, a clockwise rotation of the steering wheel causes a rightward rack displacement, from the perspective of the driver. This will cause the tires to make a right turn.

Diagram

Description automatically generated

Figure []: Physical System Diagram

To model the system, two functions are used. The first is modeling how the steering wheel angle () relates to rack displacement, and the second is to model how rack displacement () relates to tire angles ().

Rack displacement is proportional to steering wheel angle.

Tire angles can be approximated with a cubic expression, where . The figure below is for the right tire.

Chart, line chart, scatter chart

Description automatically generated

Figure []: Approximation of Tire Steering Angles

Below are tables describing all the signals and constants that compose of this system. Note that unit conversions are needed to obtain the correct .

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Description | Range | Unit |
|  | steering wheel angle, CW is positive |  | deg |
|  | rack Displacement, right is positive |  | mm |
|  | right tire angle, right turn is positive |  | deg |
|  | left tire angle, right turn is positive |  | deg |

|  |  |  |
| --- | --- | --- |
| Constant | Value | Unit |
|  | 0.00962 |  |
|  |  |  |
|  | -0.0038 |  |
|  | 0.6536 |  |
|  | 0.1061 | deg |

## Tire Model

The tire model takes in 17 unique values and outputs 2 unique values. The tire model uses the normal force and deformation of each tire to predict the current lateral force on each tire, as well as the max longitudinal force before sliding occurs. To do this, the inputs are compressed into a single value for each tire, called non-dimensional combined slip. This value is fed into the core function, which yields the corresponding non-dimensional combined force. The 17 unique inputs are then used to decompress this value and yield the 2 unique outputs. Calspan has tested the exact tires used on PER22, and this data will be used to construct the tire model.

Diagram, text

Description automatically generated

Figure []: System Diagram for Tire Model

Before a tire slides across a surface, it deforms while maintaining a significant static friction contact with the mating surface. This deformation is known as ‘slip’. Empirical data shows that the forces that act in the plane of contact between a tire and a surface is a function of slip. However, the relationship is complicated and difficult to describe analytically. As such, data compression methods have been developed to compressed large quantities of data in one, or just a few functions. One common approach is the Pacejka ‘magic’ formula, however it can only be used to obtain lateral force. The one used here is the non-dimensional tire model, proposed by Milliken & Milliken. This method combines lateral and longitudinal force characteristics. Below is a summary of the procedure, the underlying theory and the analysis results for the LC0 tires currently used on PER22. For more information, the documents ‘Non-Dimensional Tire Model Theory’ and ‘Definition of key Parameters’ are excellent resources.

The first step is to compute normal force. Unfortunately, the method used is still in development, and is not fully understood. Once normal force is known for each tire, the 5 fundamental parameters for each tire can be computed. These parameters all modeled as only a function of normal force. They are shown boxed in in the tire model system diagram. Each parameter will be explained. In this project, the Camber Stiffness parameter is not used. It would add significant complexity, the curve fit data did not seem to be a good fit, and the effect of camber seemed to be *relatively* small.

The first two parameters are the lateral and longitudinal coefficients of friction. At Calspan, the surface used is unusually grippy. This difference in grip between test and application will For each nominal normal force, the peak lateral and longitudinal force is used to compute the coefficients.

As such, the method to compute the coefficients is simple. Simply divide the tractive forces by normal force and use the peak coefficient values. There is no need to control for slip if the dataset contains non-zero slip, because in that scenario, the force will be smaller than if no slip is present. The data used contains significant variation. As such, the top ~2.5% of force values are used to average out the results. Each coefficient is plotted with respect to the normal force and curve fit with an appropriate function. As such, the friction models are as follows, where is the normal force for each tire.

The last two parameters are the cornering (C) and longitudinal stiffness () coefficients. These are much more complicated. They are defined as follows, where is lateral slip (slip angle) and S is longitudinal slip (SAE slip ratio). These coefficients are the initial slope of the *dimensional* slip-force curve. That is, the force as a function of slip.

The Calspan data has all the inputs computed. However, these coefficients require additional filtering. For C, there must be no longitudinal slip, nor camber. For , there must be no slip angle, nor camber. Once this filtering is done, the procedure used for the friction coefficients is repeated. As such, the stiffness model are as follows. To model C, an interpolated spline was used to ensure good fit and reasonable values for large and small normal forces.

|  |  |
| --- | --- |
| Normal Force (N) | C (N/rad) |
| 0 | 0 |
| 204.13 | 13757.41 |
| 427.04 | 21278.97 |
| 668.1 | 26666.02 |
| 895.72 | 30253.47 |
| 1124.40 | 30313.18 |
| 1324.4 | 30313.18 |

The four parameters are plotted below.

Chart, line chart

Description automatically generated

Figure []: Fundamental Tire Parameters

The final step before data compression is to compute the tire slip. For Lateral slip (slip angle ), this is the angle between the direction of travel, and the direction the tire is pointing. This can be done using the vehicle center of gravity (CoG) velocity and yaw rate.

For longitudinal slip, it is defined by SAE and is the ratio of longitudinal velocity to angular velocity.

Where the velocity *at* each tire is can be found with the following 12 equations.

Now, the non-dimensional combined slip (k) may be computed. As described in Milliken & Milliken, this procedure is repeated for each tire. This number can be used to understand to net slip that is occurring, where larger k means more slip.

The input to the non-dimensional tire model has now been described. To complete the model, the output must also be computed. As described in Milliken & Milliken:

R & k were computed using the Calspan data, and an appropriate equation was used to fit the data, . The resulting plot is shown.

Chart

Description automatically generated

Figure []:

With this model, k can be used to compute R. However, R is the *combined* normalized tractive force. The distribution of force between the lateral and longitudinal components needs to be known. In the future, it may be better to use the empirical data directly to describe the distribution. However, as described by Milliken & Milliken:

To predict the lateral force on each tire, the above procedure is replicated exactly. To estimate the maximum longitudinal force before slip occurs, a chosen k value is used to obtain the corresponding R value. At this point, the proscribed procedure to decompress R is replicated exactly. The k value is chosen where the tires experience peak combined tractive force. k values above this point are counter-productive since the function is expected to be non-increasing after this point.

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Description | Reasonable Range | Unit |
|  | Lateral Coefficient of Friction |  | 1 |
|  | Longitudinal Coefficient of Friction |  | 1 |
|  | Longitudinal Tractive Force |  | N |
|  | Lateral Tractive Force |  | N |
|  | Tire Normal Force |  | N |
|  | Slip Angle |  | Rad |
| S | Slip Ratio |  | None |
| C | Cornering Stiffness |  |  |
|  | Longitudinal Stiffness |  | N |
|  | Longitudinal Velocity CoG |  |  |
|  | Lateral Velocity CoG |  |  |
|  | Yaw Rate |  |  |
|  | Tire Velocity Magnitude |  |  |
|  | Tire Angular Velocity |  |  |
|  | Non-Dimensional Combined Slip |  | 1 |
|  | Non-Dimensional Combined Force |  | 1 |
|  | Initial Ratio Splitter between |  | 1 |
|  | Ratio Splitter between |  | 1 |

|  |  |  |
| --- | --- | --- |
| Constant | Value | Unit |
|  |  |  |
|  |  |  |
|  |  |  |
|  | 3.634 | 1 |
|  |  |  |
|  |  |  |
|  | 3.797 | 1 |
|  | 39.83 |  |
|  | 813.08 |  |
|  | 0.792 | m |
|  | 0.783 | m |
|  | 0.648 | m |
|  | 0.621 | m |
|  | 0.223 | m |
|  | 1.14 | 1 |
|  |  | 1 |
|  | -1.14 | 1 |
|  | -1.027 | 1 |

## Brake Model

The brake force is used when regenerative braking is active. At low speeds the brake force is assumed to be zero, and the brake force is always in the opposite direction as the velocity.

A picture containing diagram

Description automatically generated

Figure []: System Diagram for Brake Model

It is assumed that the brake pressure and brake force are proportional. However, the brake pad is larger in the front than in the rear.

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Description | Range | Unit |
|  | Brake Torque |  | Nm |
|  | Brakeline Pressure |  | MPa |

|  |  |  |
| --- | --- | --- |
| Constant | Value | Unit |
|  | 61.3278 |  |
|  | 40.5366 |  |

## PI Controller

A proportional-integral controller with anti-windup and tracking will be used. A detailed description of this control scheme (excluding anti-windup) can be found in ‘PID Control’. Proportional control is used so that the response time is small, while integral control is used to allow the vehicle to reach steady state. Anti-windup is used because the control signal frequently reaches saturation. Without anti-windup, there vehicle would be more likely to oscillate because the integral will continue increasing without additional response from the vehicle. Tracking is used on the control signal because it is expected that the target angular acceleration will not be achieved using the corresponding torque request. As such, the tracking signal will account for that by increasing or decreasing in proportion to the error between the target and actual angular acceleration.

Graphical user interface

Description automatically generated

Figure []: System Diagram for PI Controller

The reference yaw rate is composed of two potential yaw rates. The time-optimal yaw rate, as described by ‘Time optimal yaw rate (TOY)’. This is the preferred yaw rate since it is theoretically the fastest. The second yaw rate is the friction limited yaw rate. This is the maximum allowed yaw rate for the given velocity and steering angle before the vehicle spins out of control. To compute the TOY, the following equation may be used.

Chart

Description automatically generated

Figure []: TOY

The max allowed yaw rate is currently approximated with an interpolated spline. During testing, a better function should be determined.

|  |  |
| --- | --- |
|  |  |
| 0 | 1.34 |
| 3 | 1.34 |
| 6 | 1.34 |
| 9 | 1.66 |
| 12 | 1.14 |
| 15 | 1.24 |
| 18 | 0.754 |
| 21 | 0.849 |
| 25 | 0.85 |

Once the two yaw rates are computed, the yaw rate closest to zero is chosen. Next, the error between the reference yaw rate, and the measured yaw rate is found. This error term is fed into the PI controller. To implement anti-windup, the integral portion is multiplied by a trigger term, which is equal to 0 when anti-windup is active.

‘Angular Acceleration’ is the previous iteration measured angular acceleration of the vehicle. ‘Target Angular Acceleration’ is the angular acceleration that results if the previous torque request is fulfilled. When the difference between these terms is non-zero, tracking error has occurred. In this scenario, the signal, and is non-zero. The tracking error is integrated.

The control signal from the PI portion is added with the control signal from the tracking portion. The physical significance of the control signal is that it is the desired angular acceleration of the vehicle. The PI portion is essential to proper TVS operation, but the tracking portion may not be. As such, it may be determined that the tracking portion is not needed and may be discarded.

## Constraint Generation

Torque vectoring aims to find the most optimal distribution of torque. This optimal distribution is subject to stability constraints. Many details of this section may be found in ‘Torque Vectoring Fundamentals’, chapter 7 LP Controller. Below is the system diagram.

A picture containing timeline

Description automatically generated

Figure []: System Diagram for Constraint Generation

First, the search space (torque) is defined. The slip limit is the maximum torque before tire sliding occurs. The power limit is the torque which would consume 15kW of power. The RPM limit is the maximum possible torque the motor can output at the given RPM of the motor. The Motor response limit is the largest and smallest torque the motor can achieve by the next time step. Currently, this limit is disabled. These 4 limits form the upper and lower boundaries of torque that the optimization may look at.

The rpm and power limits are (currently) only functions of motor shaft speed.

Chart, line chart

Description automatically generated

Next, user input is considered. This is done in two ways: power & yaw control. For power control, the upper limit for power is set based on the accelerator/brake pedal input. This is implemented as an inequality constraint into the optimization problem.

For yaw control, the target angular acceleration is determined from the current vehicle velocity and the steering angle. This is done by the PI control portion. This is then implemented as an equality constraint into the optimization problem.

Text, letter

Description automatically generated

Figure []: Optimization Problem

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Description | Range | Unit |
|  | Torque Request, Tx |  | Nm |
|  | Max Torque Allowed |  | Nm |
|  | Min Torque Allowed |  | Nm |
|  | Moment Arm for Fx |  | m |
|  | Moment Arm for Fy |  | m |
|  | Current Lateral Force |  | N |
|  | Target Angular Acceleration |  |  |
|  | Max Power Allowed |  | W |
|  | Min Power Allowed |  | W |
|  | Angular Velocity of Motor Shaft |  |  |
|  | Torque Slip Limit |  | Nm |
|  | Torque Power Limit |  | Nm |
|  | Torque RPM Limit |  | Nm |
|  | Torque Motor Response Limit |  | Nm |
|  | Current at Max Torque |  | A |
|  | Model Coefficient |  | RPM |
|  | Previous Tx |  | Nm |
|  | Motor Shaft Speed |  | RPM |

|  |  |  |  |
| --- | --- | --- | --- |
| Constant | Description | Value | Unit |
|  | Polar Mass Moment of Vehicle | 75 |  |
|  | Convert Motor Torque to Ground Tractive Force | [29 29 35 35] |  |
|  | Motor Efficiency | [0.85 0.85 0.85 0.85] | 1 |
|  | Power Limit | 15000 | W |
|  | Model Coefficient | 0.2667 |  |
|  | Model Coefficient | -66.353 |  |
|  | Model Coefficient | -0.0987 |  |
|  | Model Coefficient | 45.799 |  |
|  | Model Coefficient | -129.65 | hNm |
|  | Motor Response Time Constant | 0 |  |
|  | Controller Timing | 0.015 | s |

## Optimization

All the elements are constructed to be able to execute an optimization. However, the exact nature of the optimization must be carefully selected to maximize the chance of a valid solution. The only time that TVS is completely inactive is at low speeds & braking. The next step is to check for ‘zero crossing’. This is when at least one of the torques changes sign. The current optimization algorithm struggles to find the most optimal solution when this is true. A special algorithm was designed to address this issue. Finally, a secondary optimization (different from the one presented) is done when the primary optimization fails. Using this approach, a valid solution is usually found.

Diagram

Description automatically generated

Figure []: System Diagram for Optimization

The zero-crossing algorithm works as follows. Two optimizations are run sequentially, where the second optimization uses information from the first optimization. The first iteration of the optimization is given what sign to expect each torque to be. The sign is chosen based on the sign of the previous torque request. The optimization is run, and the sign of the new torque request is now used to run the second optimization. The output from this second optimization is the final torque request in the case of zero crossing.

The secondary optimization replaces the objective statement with the angular acceleration equality. As in, the objective is to minimize the error between the actual angular acceleration, and the target angular acceleration.

At this point, the logic and optimization schemes will yield a single torque request, which is directly fed to the motors.

Using the yaw equality, the actual angular acceleration is computed, and used for tracking.