Report:MPC-Based Approach to Active Steering for Autonomous Vehicle Systems

İsmail Çağdaş Yılmaz

1 Modeling

To describe the dynamics of the car 'Bicycle model' is employed and constant normal tire load, i.e., F_{z_f},F_{z_f} = constant is assumed. The following three equations represents the longitudinal, lateral and turning or yaw degrees of freedom (DOF)

$$\ddot{x} = \dot{y}\dot{\psi} + \frac{2(F_{x_f} + F_{x_r})}{m} \tag{1a}$$

$$\ddot{y} = -\dot{x}\dot{\psi} + \frac{2(F_{y_f} + F_{y_r})}{m} \tag{1b}$$

$$\ddot{\psi} = \frac{2(aF_{y_f} - bF_{y_r})}{I} \tag{1c}$$

where x and y are the coordinates of the center of mass in an inertial frame (X, Y). is the inertial heading and v is the speed of the vehicle. Speed of the vehicle is taken constant for active steering, and acceleration of the center of mass is zero. a and b represent the distance from the center of the mass of the vehicle to the front and rear axles, respectively. The vehicle's equations of motion in an absolute inertial frame are

$$\dot{Y} = \dot{x}\sin(\psi) + \dot{y}\cos(\psi) \tag{2a}$$

$$\dot{X} = \dot{x}\cos(\psi) - \dot{y}\sin(\psi) \tag{2b}$$

Longitudinal and lateral tire forces which are formulated in 2 lead to the following forces acting on the center

$$F_y = F_l\sin(\delta) + F_c\cos(\delta) \tag{3a}$$

$$F_x = F_l\cos(\delta) - F_c\sin(\delta) \tag{3b}$$

Tire forces, $F_l = f_l(\alpha, s, \mu, F_z)$, and $F_c = f_c(\alpha, s, \mu, F_z)$, for each tire are the functions of slip angle (α), slip ratio (s) road friction coefficient (μ) and normal tire load (F_z). The slip ratio defined as

$$s = \begin{cases} \frac{r\omega}{v} - 1 \text{ if } v > r\omega,\ v \neq 0 \text{ for braking} \\ 1 - \frac{r\omega}{v} \text{ if } v < r\omega,\ \omega \neq 0 \text{ for driving} \end{cases} \tag{4}$$

The slip angle represents the angle between the wheel velocity and the direction of the wheel itself:

$$\alpha = \tan^{-1}\frac{v_c}{v_l} \tag{5}$$

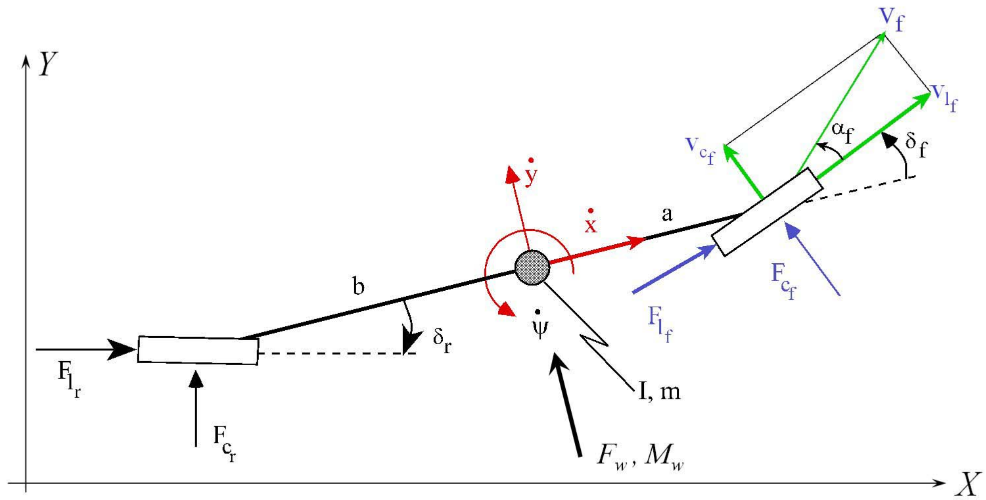

Figure 1: The simplified vehicle dynamical model. [2]

In equation 5, v_c and v_l are the lateral (or cornering) and longitudinal wheel velocities, respectively, which are expressed as

$$v_l = v_y \sin(\delta) + v_x \cos(\delta) \tag{6a}$$
$$v_c = v_y \cos(\delta) - v_x \sin(\delta) \tag{6b}$$

and

$$v_{y_f} = \dot{y} + a\dot{\psi} \qquad\qquad v_{y_r} = \dot{y} - b\dot{\psi} \tag{7a}$$
$$v_{x_f} = \dot{x} \qquad\qquad v_{x_r} = \dot{x} \tag{7b}$$

F_z is distributed between the front and rear wheels based on the geometry of the car model, described by the parameters a and b;

$$F_{z_f} = \frac{bmg}{2(a+b)} \tag{8a}$$
$$F_{z_r} = \frac{amg}{2(a+b)} \tag{8b}$$

Using the equations 1-8, the nonlinear vehicle dynamics can be described by the following compact differential equation assuming a **certain slip ratio** (s) and **friction coefficient value** (μ):

$$\dot{\boldsymbol{\xi}} = f_{s,\mu}(\boldsymbol{\xi}, u) \tag{9a}$$
$$\boldsymbol{\eta} = h(\boldsymbol{\xi}) \tag{9b}$$

where the state and input vectors are $\boldsymbol{\xi} = [y\ \dot{y}\ x\ \dot{x}\ \psi\ \dot{\psi}\ Y\ \ X]^T$ and $u = \delta_f$, respectively, and the output map is given as

$$\boldsymbol{\eta} = \begin{bmatrix} \psi \\ Y \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \boldsymbol{\xi} \tag{10}$$

Nonlinear kinematic and dynamics equations of 2WS the vehicle can also be represented with control input, δ_f, as follows, [2], [3], [4]:

$$\dot{\boldsymbol{\xi}}(t) = \left(\begin{array}{cc} \dot{y} \Rightarrow & v\sin(\psi + \beta) \\ \ddot{y} \Rightarrow & -\dot{x}\dot{\psi} + \frac{2(F_{y_f} + F_{y_r})}{m} = -\dot{x}\dot{\psi} + \frac{2}{m}(F_{c,f}\cos(\delta_f) + F_{c,r}) \\ \dot{x} \Rightarrow & v\cos(\psi + \beta) \\ \ddot{x} \Rightarrow & \dot{y}\dot{\psi} + \frac{2(F_{x_f} + F_{x_r})}{m} \\ \dot{\psi} \Rightarrow & r \\ \ddot{\psi} = \dot{r} \Rightarrow & \frac{1}{I}(\alpha F_{y_f}\cos(\delta) - bF_{y_r}) \\ \dot{Y} \Rightarrow & \dot{x}\sin(\psi) + \dot{y}\cos(\psi) \\ \dot{X} \Rightarrow & \dot{x}\cos(\psi) - \dot{y}\sin(\psi) \end{array} \right) \tag{11}$$

where $\beta = \tan^{-1}(\frac{b}{a+b}\tan(\delta_f))$ and acceleration of the car is constant, i.e. $\dot{v} = 0$. Forces in x- and y-directions, F_x and F_y, are the function of the forward steering angle, δ_f and longitudinal and lateral tire forces acting on the center.

2 Tire Model

Three possible ways to represent measured tyre data are in use [1]:

- representation by tables,
- representation by graphs,
- representation by formula.

The best way to fulfill the requirements is to find a special function, which through its particular structure is capable of describing the measured data with great accuracy and which has parameters related to the typifying quantities in a simple manner. The formula should, if possible, be able to describe:

- the side force as a function of slip angle, α_f & α_r,
- the brake force as a function of longitudinal slip,
- the self aligning torque as a function of slip angle.

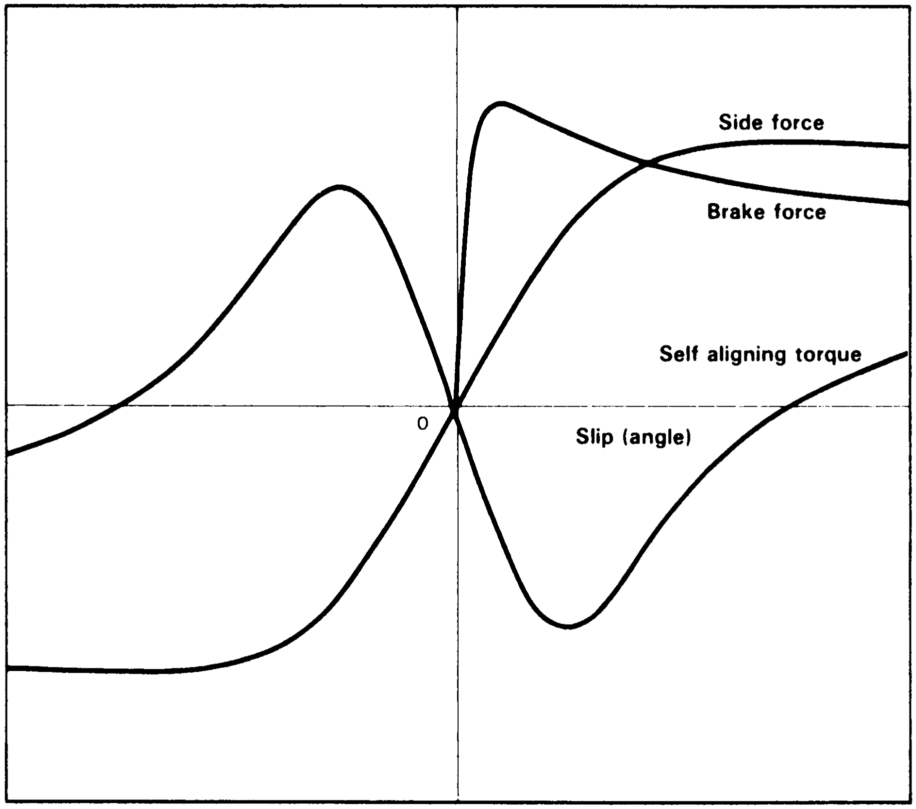

Figure 2: Steady-state tyre characteristics.

The basic form of each of the characteristics of large values the tyre suggest the use of the sine function as a first step in developing the final formula.

$$F = D\sin(B\alpha) \tag{12}$$

with F standing for either side force, self aligning torque or break force and α denoting slip angle. e. In 12 D is the peak value and the product DB equals the slip stiffness at zero slip. Eq. 12 does not give a good representation for larger values of α. A gradually increasing extension of the X axis appears to be necessary. To accomplish this, the arctan function has been used. The formula 12 now changes into:

$$F = D\sin(C\arctan(B\alpha)) \tag{13}$$

In Eq. (2) D is still the peak value, the slip stiffness at zero slip is now equal to the BCD (from now on called the stiffness). The coefficient C governs the shape of the curve in Fig 2.The value of C makes the curve look like a side force, brake force or a self aligning torque characteristic. With C determined by the shape and D determined by the peak value, only B is left to control the stiffness. Still Eq. 13 is not good enough to describe every possible measured characteristic. There may be a need for an additional coefficient which makes it possible to accomplish a local extra stretch or compression of the curve. The coefficient E has been introduced into the formula in such a way that stiffness and peak value remain

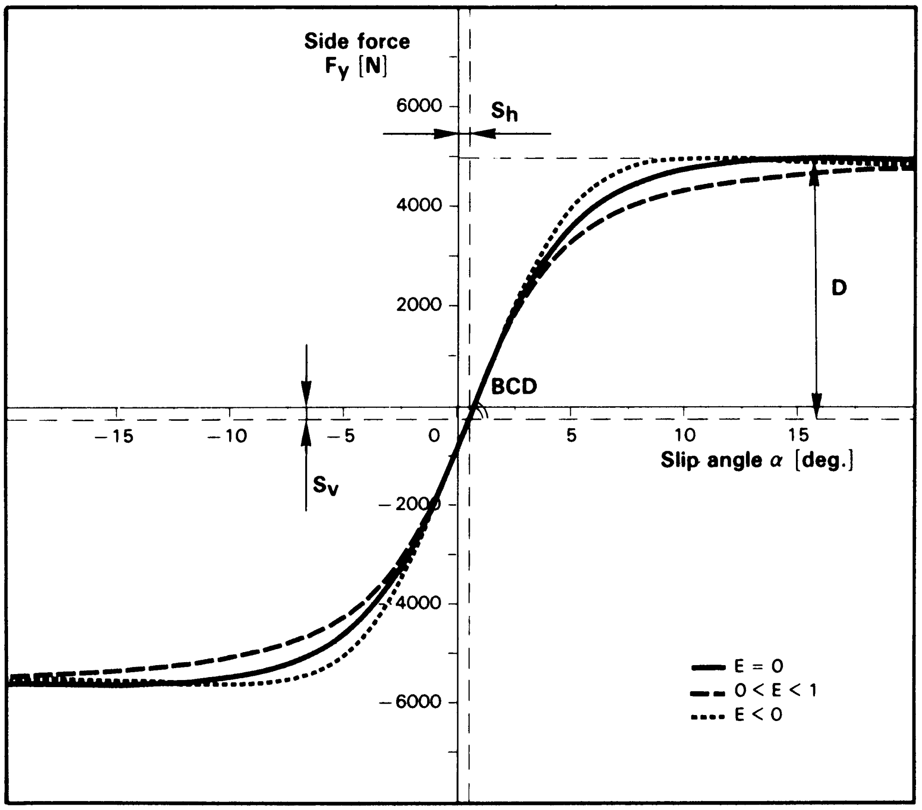

Figure 3: Coefficients appearing in tyre formula.

unaffected.

$$F = D \sin(C \arctan(B\Phi)) \tag{14a}$$

$$\Phi = (1 - E)\alpha + \frac{E}{B} \arctan(B\alpha) \tag{14b}$$

The influence of E on the side force characteristic has been shown in Fig. 3. Similar effects occur with the self aligning torque and brake force characteristics. The result is an equation with four coefficients, which is able to describe all the measured characteristics. The four coefficients are:

- $B \Rightarrow$ stiffness factor
- $C \Rightarrow$ shape factor
- $D \Rightarrow$ peak factor
- $E \Rightarrow$ curvature factor

2.1 Influence of Vertical Load

To reduce the total number of quantified coefficients and to be able to calculate forces and torques at vertical loads which are different from the values used in the measurements, it is necessary to include the vertical load explicitly in the formula. To do so, the coefficients have to be written as a function of the vertical (normal) load, F_z. The peak factor, D, as a function of F_z may be approximately represented by the relationship:

$$D = a_1 {F_z}^2 + a_2 F_z \tag{15}$$

For the stiffness BCD of the side force characteristic (cornering stiffness), the formula is written as

$$BCD = a_3 \sin(a_4 \arctan(a_5 F_z)) \tag{16}$$

and for the stiffness of both brake force (longitudinal slip stiffness) and self aligning torque (aligning stiffness) characteristics, the approximation is:

$$BCD = \frac{a_3 {F_z}^2 + a_4 F_z}{e^{a_5 F_z}} \tag{17}$$

The shape factor, C, appears to be practically independent of F_z. For above mentioned force types C takes following values:

- the side force : $C = 1.30$
- the brake force : $C = 1.65$
- the self aligning torque : $C = 2.40$

The stiffness factor B is found by stiffness by the shape and the peak factor.

$$B = \frac{BCD}{CD} \tag{18}$$

Finally, the curvature factor E as a function of F_z is given by:

$$E = a_6 {F_z}^2 + a_7 F_z + a_8 \tag{19}$$

2.2 Proposed Tyre Formulas

2.2.1 Side Force, F_y

$$F_y = D \sin(C \arctan(B\Phi)) + \cancel{\Delta S_v} \tag{20a}$$

$$\Phi = (1 - E)(\alpha + \cancel{\Delta S_h}) + (E/B) \arctan(B(\alpha + \cancel{\Delta S_h})) \tag{20b}$$

$$D = a_1 {F_z}^2 + a_2 F_z \tag{20c}$$

$$C = 1.30 \tag{20d}$$

$$B = \left(\frac{a_3 \sin(a_4 \arctan(a_5 F_z))}{CD} \right) (1 - \cancel{a_{12}|\gamma|}) \tag{20e}$$

$$E = a_6 {F_z}^2 + a_7 F_z + a_8 \tag{20f}$$

$$\Delta S_h = a_9 \gamma \tag{20g}$$

$$\Delta S_v = (a_{10} {F_z}^2 + a_{11} F_z)\gamma \tag{20h}$$

2.2.2 Self Aligning Torque, M_z

$$M_z = D\sin(C\arctan(B\Phi)) + \cancel{\Delta S_v} \tag{21a}$$
$$\Phi = (1-E)(\alpha + \cancel{\Delta S_h}) + (E/B)\arctan(B(\alpha + \cancel{\Delta S_h})) \tag{21b}$$
$$D = a_1 {F_z}^2 + a_2 F_z \tag{21c}$$
$$C = 2.40 \tag{21d}$$
$$B = \left(\frac{a_3 {F_z}^2 + a_4 F_z}{CDe^{a_5 F_z}}\right)(1 - \cancel{a_{12}|\gamma|}) \tag{21e}$$
$$E = (a_6 {F_z}^2 + a_7 F_z + a_8)/(1 - \cancel{a_{13}|\gamma|}) \tag{21f}$$
$$\Delta S_h = a_9 \gamma \tag{21g}$$
$$\Delta S_v = (a_{10} {F_z}^2 + a_{11} F_z)\gamma \tag{21h}$$

2.2.3 Brake Force, F_x

$$F_y = D\sin(C\arctan(B\Phi)) + \cancel{\Delta S_v} \tag{22a}$$
$$\Phi = (1-E)\alpha + (E/B)\arctan(B\alpha) \tag{22b}$$
$$D = a_1 {F_z}^2 + a_2 F_z \tag{22c}$$
$$C = 1.65 \tag{22d}$$
$$B = \frac{a_3 {F_z}^2 + a_4 F_z}{CDe^{a_5 F_z}} \tag{22e}$$
$$E = a_6 {F_z}^2 + a_7 F_z + a_8 \tag{22f}$$
$$\Delta S_h = a_9 \gamma \tag{22g}$$
$$\Delta S_v = (a_{10} {F_z}^2 + a_{11} F_z)\gamma \tag{22h}$$

2.3 Matlab Code for Longitudinal and Lateral Tire Forces

```
clc;
clear all;
close all;

%%Physical parameters of Car
m=1700; % in Kg
Iz=2900; %
a=1.5; %length from CoG to front axle in meters
b=1.4; %length from CoG to back axle in meters


%% Pajecka Tire Model (Magic Formula)

%Slip angle takes values between -20 and 20 degrees.
```

```
%Slip ratio takes values between -100% and 100%.
a1_x=-21.3; a2_x=1144; a3_x=49.6; a4_x=226; a5_x=0.069; a6_x=-0.006;
a7_x=0.056; a8_x=0.486;

C_x=1.65;


a1_y=-22.1; a2_y=1011; a3_y=1078; a4_y=1.82; a5_y=0.208; a6_y=0.0;
a7_y=-0.354; a8_y=0.707;

C_y=1.3;
%%

Fz_f = (b*m*9.81)/(1000*2*(a+b)); %force in K-newtons
Fz_r = (a*m*9.81)/(1000*2*(a+b)); %force in K-newtons

Fz = [Fz_f; Fz_r];

j = 0;
for k = 1:1:2

 i=0;
 j=j+1;

 for alpha = -20:0.01:20
 i=i+1;

 D_x = (a1_x * (Fz(k)^2)) + ( a2_x * Fz(k) );
 BCD_x = ((a3_x * (Fz(k)^2)) + (a4_x * (Fz(k)^2))) / (exp(a5_x * Fz(k)));
 B_x = BCD_x / (C_x * D_x);
 E_x = (a6_x * (Fz(k)^2)) + a7_x * Fz(k) + a8_x;

 D_y = (a1_y * (Fz(k)^2)) + (a2_y * Fz(k));
 BCD_y = a3_y * sind(a4_y * atand(a5_y * Fz(k)));
 B_y = BCD_y / (C_y * D_y);
 E_y = (a6_y * (Fz(k)^2)) + a7_y * Fz(k) + a8_y;


 Fl(i,j) = D_x * sind(C_x * atand(B_x * (alpha)));
 Fc(i,j) = D_y * sind(C_y * atand(B_y * (alpha)));
 end

end
```

```matlab
%%

ratio = -100:0.05:100;
figure
plot(ratio , Fl(:,1), ratio, Fl(:,2), 'linewidth', 2)
grid on
title('$Longitudinal\,\,Forces\,\,for\,\,Front\,\,and\,\,Rear\,\,Tyres$',...
    'fontsize',18, 'fontweight','b', 'interpreter', 'latex')
xlabel('$Tire\,\,slip\,\,ratio[\%]$','fontsize', 18, 'interpreter', 'latex')
ylabel('$F_{l_{f}},F_{l_{r}}[N]$','fontsize', 18, 'interpreter', 'latex')
l = legend('$F_{l_{f}}$','$F_{l_{r}}$','Location','SouthEast');
set(l, 'interpreter', 'latex', 'fontsize', 18)


alpha =-20:0.01:20;
figure
plot(alpha, Fc(:,1), alpha, Fc(:,2), 'linewidth', 2)
grid on
title('$Lateral\,\,Forces\,\,for\,\,Front\,\,and\,\,Rear\,\,Tyres$',...
    'fontsize',18, 'fontweight','b', 'interpreter', 'latex')
xlabel('$Tire\,\,slip\,\,angle[deg]$','fontsize', 18, 'interpreter', 'latex')
ylabel('$F_{c_{f}},F_{c_{r}}[N]$','fontsize', 18, 'interpreter', 'latex')
l = legend('$F_{c_{f}}$','$F_{c_{r}}$','Location','SouthEast');
set(l, 'interpreter', 'latex', 'fontsize', 18)
```

3 Problem Formulation

A finite dimensional optimal control problem can be obtained by discretizing the system dynamics as indicated in 9 with the Euler method,

$$\boldsymbol{\xi}(k+1) = f^{dt}_{s,\mu}(\boldsymbol{\xi}(k), u(k)) \tag{23a}$$

$$\boldsymbol{\eta}(k) = h(\boldsymbol{\xi}(k)) \tag{23b}$$

where the Δu formulation is used, i.e., $u(k) = u(k-1) + \Delta u(k)$ and $u(k) = \delta_f(k)$, $\Delta u(k) = \Delta\delta_f(k)$. Objective or the cost function of the system can be defined as follows:

$$J(\boldsymbol{\xi}(k), \Delta \boldsymbol{U}_t) = \sum_{i=1}^{H_p} (\hat{\boldsymbol{\eta}}_{t+i,t} - \boldsymbol{\eta}_{ref_{t+i,t}})^T \boldsymbol{Q} \, (\hat{\boldsymbol{\eta}}_{t+i,t} - \boldsymbol{\eta}_{ref_{t+i,t}})$$

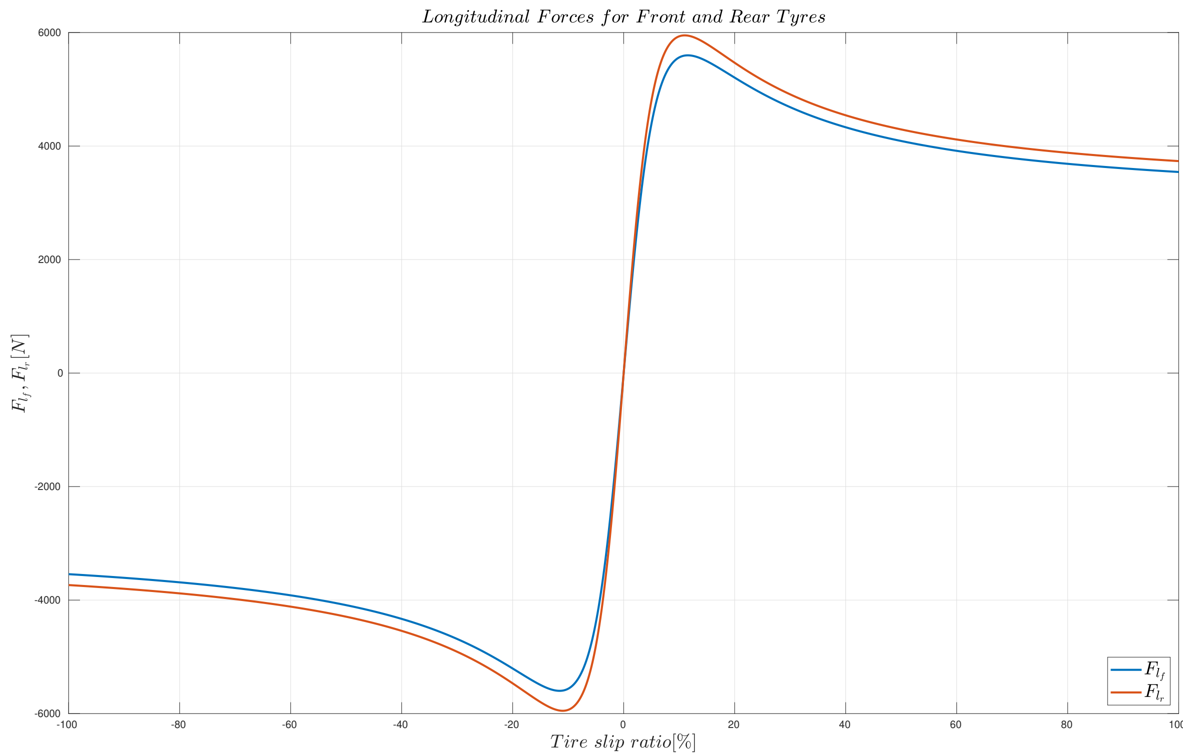

Figure 4: Longitudinal forces of front and rear tyres.

$$
\begin{aligned}
&+\sum_{i=0}^{H_c-1} \Delta u_{t+i,t}\, R\, \Delta u_{t+i,t} \\
&= \sum_{i=1}^{H_p} \left\| \hat{\boldsymbol{\eta}}_{t+i,t} - \boldsymbol{\eta}_{ref_{t+i,t}} \right\|^2_{\boldsymbol{Q}} + \sum_{i=0}^{H_c-1} \left\| \Delta u_{t+i,t} \right\|^2_R
\end{aligned}
\tag{24}
$$

In 24 the first summand reflects the desired performance on target tracking, the second summand is a measure of the steering effort. At each time step t the following finite horizon optimal control problem is solved:

$$
\begin{aligned}
\min_{\boldsymbol{\Delta U}} \quad & J(\boldsymbol{\xi}(k), \Delta \boldsymbol{U}_t) \\
\text{st.} \quad & \boldsymbol{\xi}(k+1) = f^{dt}_{s,\mu}(\boldsymbol{\xi}(k), u(k)), \;\; k = t, \ldots, t+H_p \\
& \boldsymbol{\eta}(k) = h(\boldsymbol{\xi}(k)), \;\; k = t, \ldots, t+H_p \\
& \delta_{f,min} \leq u_{k,t} \leq \Delta\delta_{f,max} \;\; k = t, \ldots, t+H_c-1 \\
& \Delta\delta_{f,min} \leq \Delta u_{k,t} \leq \Delta\delta_{f,max} \;\; k = t, \ldots, t+H_c-1 \\
& u_{k,t} = u_{k,t-1} + \Delta u_{k,t}
\end{aligned}
\tag{25}
$$

When the above optimization problem is solved at time t for the current observed states $\boldsymbol{\xi}_{t,t}$, sequence of the optimal control inputs, $\Delta \boldsymbol{U}^*_t = [\Delta u^*_{t,t}, \ldots, \Delta u^*_{t+H_c-1,t}]^T$, within the specified control horizon. The resulting state feedback control law is,

$$
\delta_f(t) = \delta_f(t-1) + \Delta u^*_{t,t} \tag{26}
$$

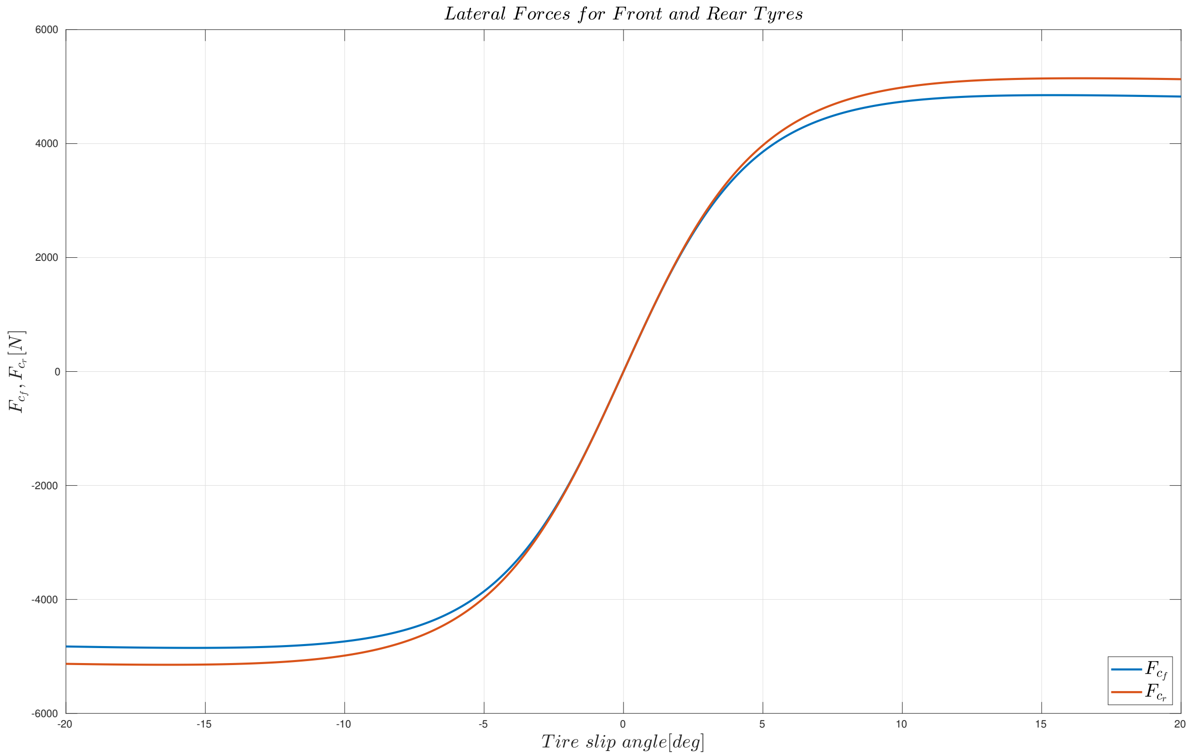

Figure 5: Lateral forces of front and rear tyres.

4 Double Lane Change on Snow Using Active Steering

4.1 Scenario Description

The reference signals (desired tracking signals) $\mu_{ref} = \begin{bmatrix} \psi_{ref} \\ Y_{ref} \end{bmatrix}$ are specified by the following set of equations:

$$Y_{ref} = \frac{d_{y_1}}{2}(1 + \tanh(z_1)) - \frac{d_{y_2}}{2}(1 + \tanh(z_2)) \tag{27a}$$

$$\psi_{ref} = \arctan\left(d_{y_1}\left(\frac{1}{\cosh(z_1)}\right)^2\left(\frac{1.2}{d_{x_1}}\right) - d_{y_2}\left(\frac{1}{\cosh(z_2)}\right)^2\left(\frac{1.2}{d_{x_2}}\right)\right) \tag{27b}$$

$$z_1 = \frac{shape}{d_{x_1}}(X - X_{s_1}) - \frac{shape}{2} \tag{27c}$$

$$z_2 = \frac{shape}{d_{x_2}}(X - X_{s_2}) - \frac{shape}{2} \tag{27d}$$

where $shape = 2.4$, $d_{x_1} = 25$, $d_{x_2} = 21.95$, $d_{y_1} = 4.05$, $d_{y_2} = 5.7$, $X_{s_1} = 27.19$ and $X_{s_2} = 56.46$. Unless differently specified, the following parameters have been used for the 2WS NLMPC controller.

4.2 Matlab Code for Checking Reference Generator

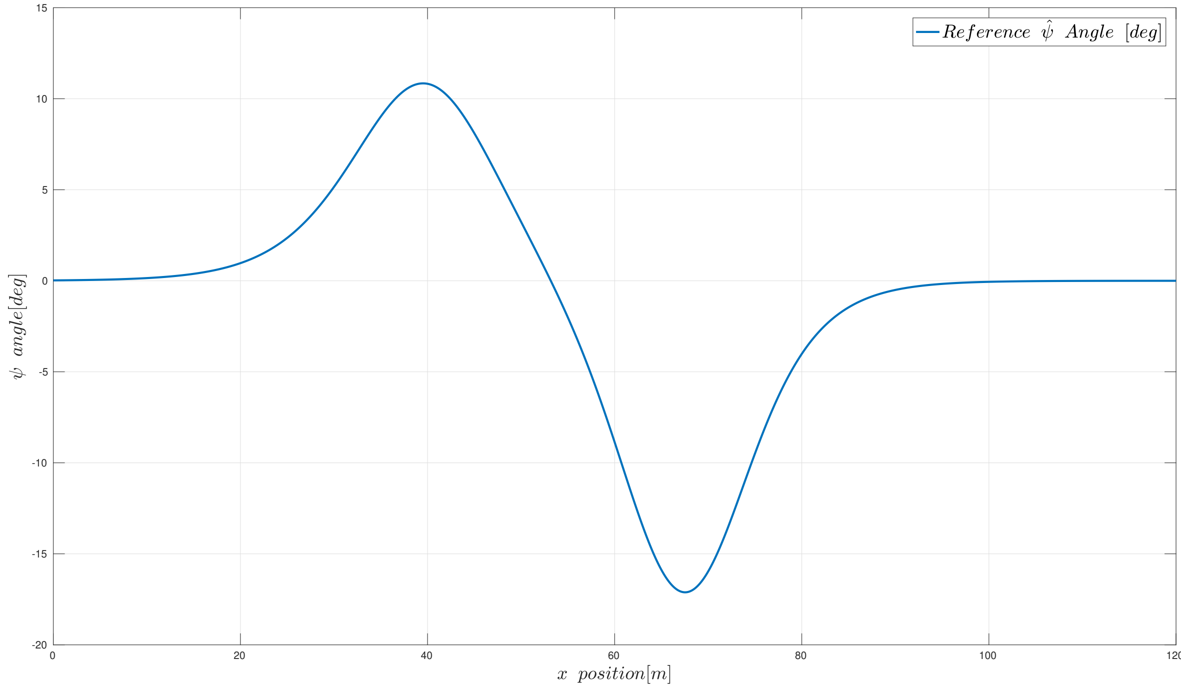

Figure 6: Reference ψ angle in degree.

```
clear;
clc;
close all;

shape = 2.4;
d_x_1 = 25;
d_x_2 = 21.95;
d_y_1 = 4.05;
d_y_2 = 5.7;
X_s_1 = 27.19;
X_s_2 = 56.46;

X = 0:0.05:120;

z_1 = (shape/d_x_1).*(X-X_s_1) - (shape/2);
z_2 = (shape/d_x_2).*(X-X_s_2) - (shape/2);

Y_ref = ((d_y_1/2)*(1+tanh(z_1)) - (d_y_2/2)*(1+tanh(z_2)));
psi_ref = [];

for i = 1:length(z_1)
    psi_ref_instant = (180/pi)*atan( d_y_1 * ( (1/cosh(z_1(i)))^2 )
        * (1.2/d_x_1) -d_y_2 * ( (1/cosh(z_2(i)))^2 ) * (1.2/d_x_2)
```

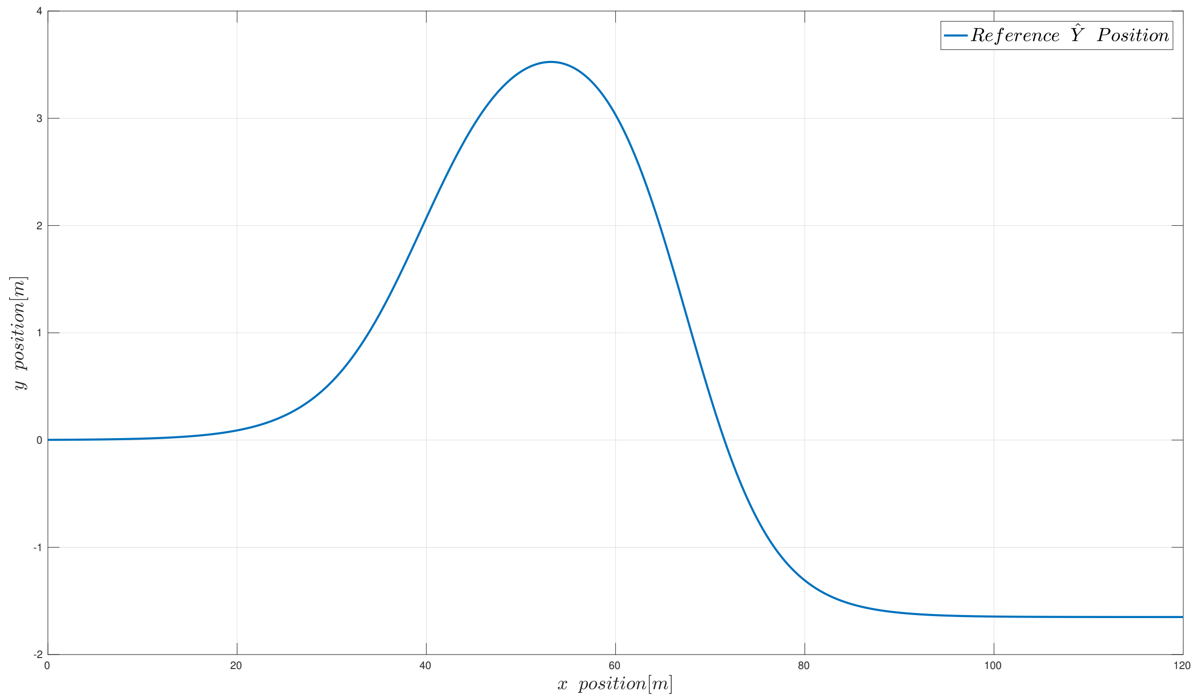

Figure 7: Reference ψ angle in degree.

```
        );
    psi_ref = [psi_ref, psi_ref_instant];
end

plot(X, Y_ref, 'LineWidth', 2);
grid on
xlabel('$x\;\;position[m]$','fontsize', 18, 'interpreter', 'latex')
ylabel('$y\;\;position[m]$','fontsize', 18, 'interpreter', 'latex')
l = legend('$Reference\;\;\hat{Y}\;\;Position$');
set(l, 'interpreter', 'latex', 'fontsize', 18)

figure
plot(X, psi_ref, 'LineWidth', 2);
grid on
xlabel('$x\;\;position[m]$','fontsize', 18, 'interpreter', 'latex')
ylabel('$\psi\;\;angle[deg]$','fontsize', 18, 'interpreter', 'latex')
l = legend('$Reference\;\;\hat{\psi}\;\;Angle\;\;[deg]$');
set(l, 'interpreter', 'latex', 'fontsize', 18)
```

- sample time: $T = 0.05$ sec;

- constraints on maximum and minimum steering angles $-30° \leq \delta_f \leq 30°$
- constraints on maximum and minimum changes in steering angles $-20°/\mathrm{s} \leq \Delta\delta_f \leq 20°/\mathrm{s}$

The controller tuning parameters at a given longitudinal vehicle speed are the prediction horizon H_p, control horizon H_c and the weighting matrices Q and R.

5 Simulation Results

5.1 Simulation Results via MATLAB & YALMIP

Dependencies For YALMIP

1. Find a suitable folder for yalmip installation and create a folder, go to the folder in MATLAB command window

   e.g. /home/cagdas/Documents/OptimizationTools/yalmip/YALMIP-master (on Linux) be sure you have permission to access,

2. Download YALMIP from website: `https://yalmip.github.io/download/`

   and copy this .zip file to the folder above

3. Copy and past the following 3 lines matlab command window
   >> unzip('YALMIP-master.zip','yalmip')
   >> addpath(genpath([pwd filesep 'yalmip']));
   >> savepath

4. Test YALMIP: run the following command in MATLAB command window, check available optimization tools among lists
   >> yalmiptest

5.2 Simulation Results via c++ (IPOPT, cppAD, matplotlibcpp)

Dependencies For C++ Implementation

1. cmake >= 3.5
2. gcc/g++ >= 5.4
3. Ipopt and CppAD: Please refer to this document for installation instructions.
4. Eigen This is already part of the source codes.

Basic Build Instructions

1. Make a build directory: **mkdir build && cd build**
2. Compile: **cmake .. && make**
3. Run it: **./active_steering_mpc**

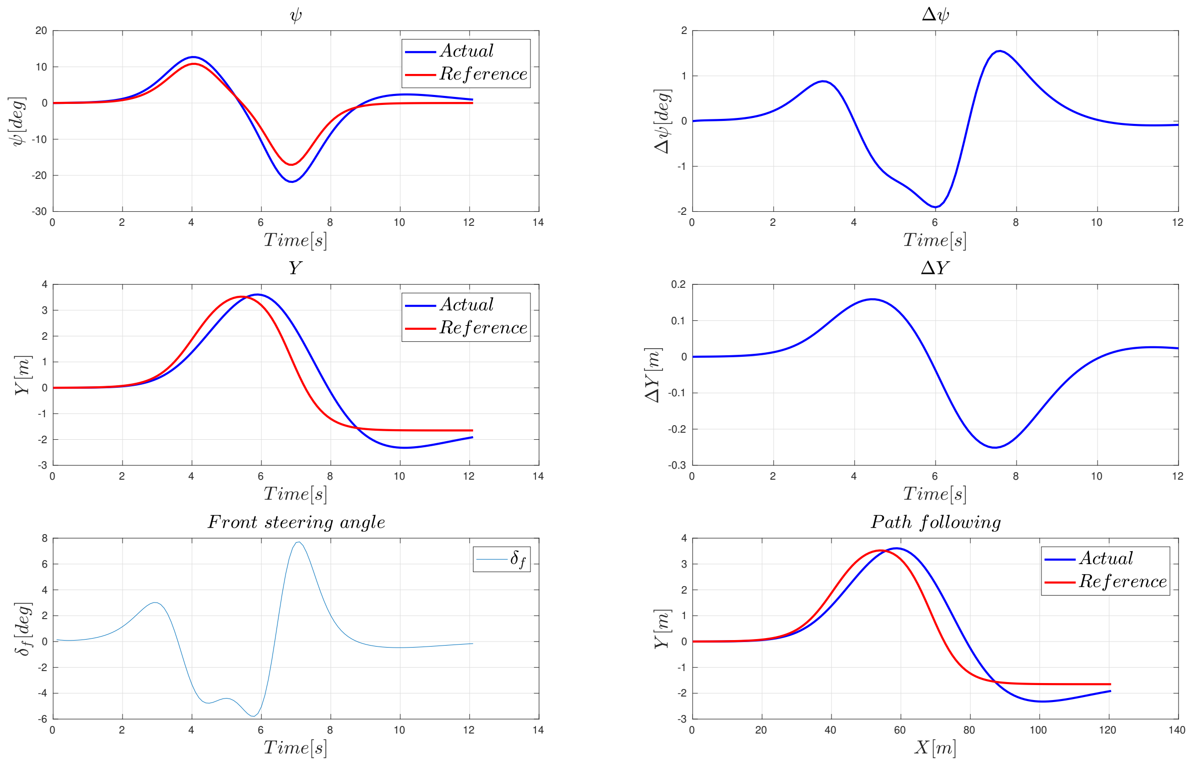

Figure 8: (MATLAB) Double lane change maneuver at 10 m/s with $H_p = 7$ and $H_c = 7$

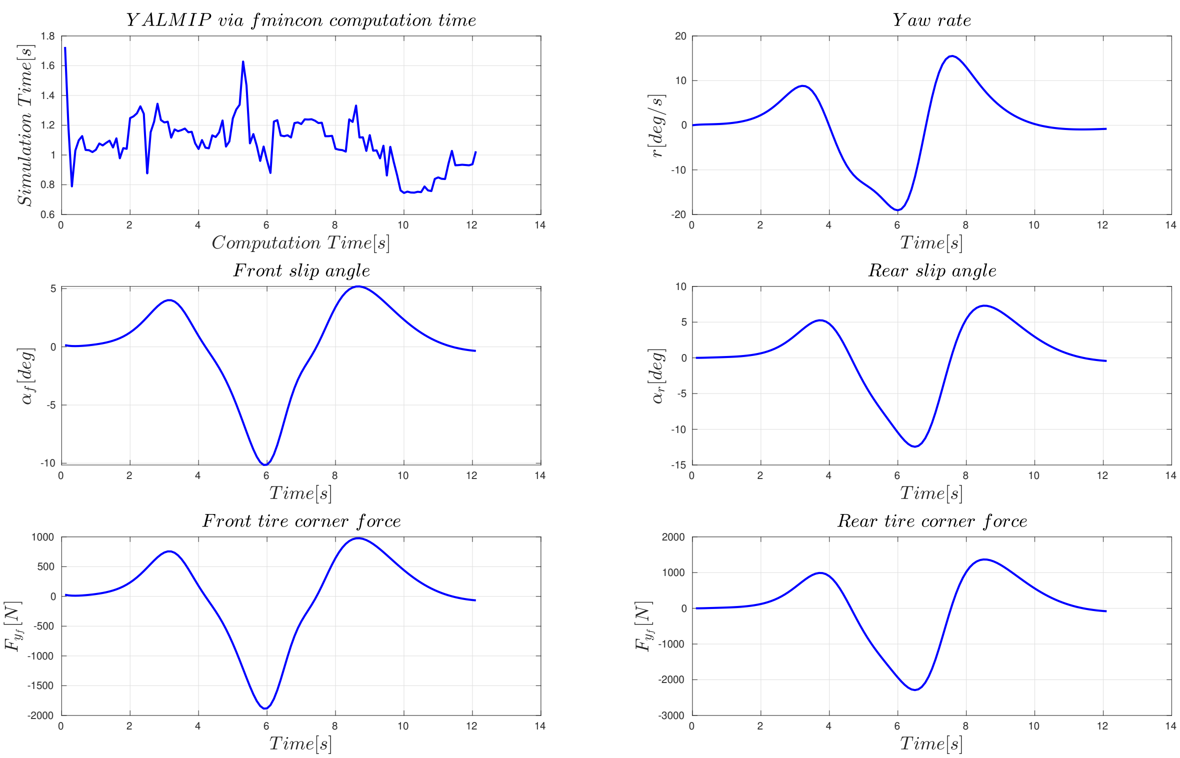

Figure 9: (MATLAB) Double lane change maneuver at 10 m/s with $H_p = 7$ and $H_c = 7$. YALMIP computation time, yaw rate, tire forces and slip angles.

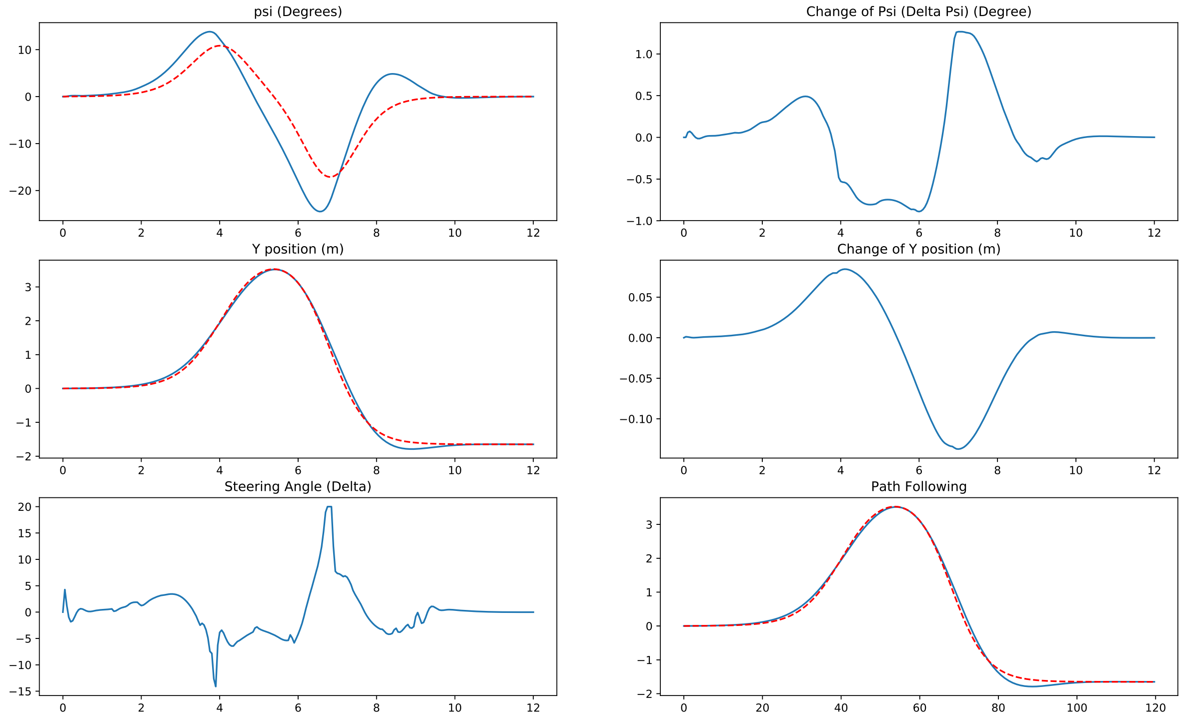

Figure 10: (c++) Double lane change maneuver at 10 m/s with $H_p = 20$ and $H_c = 20$

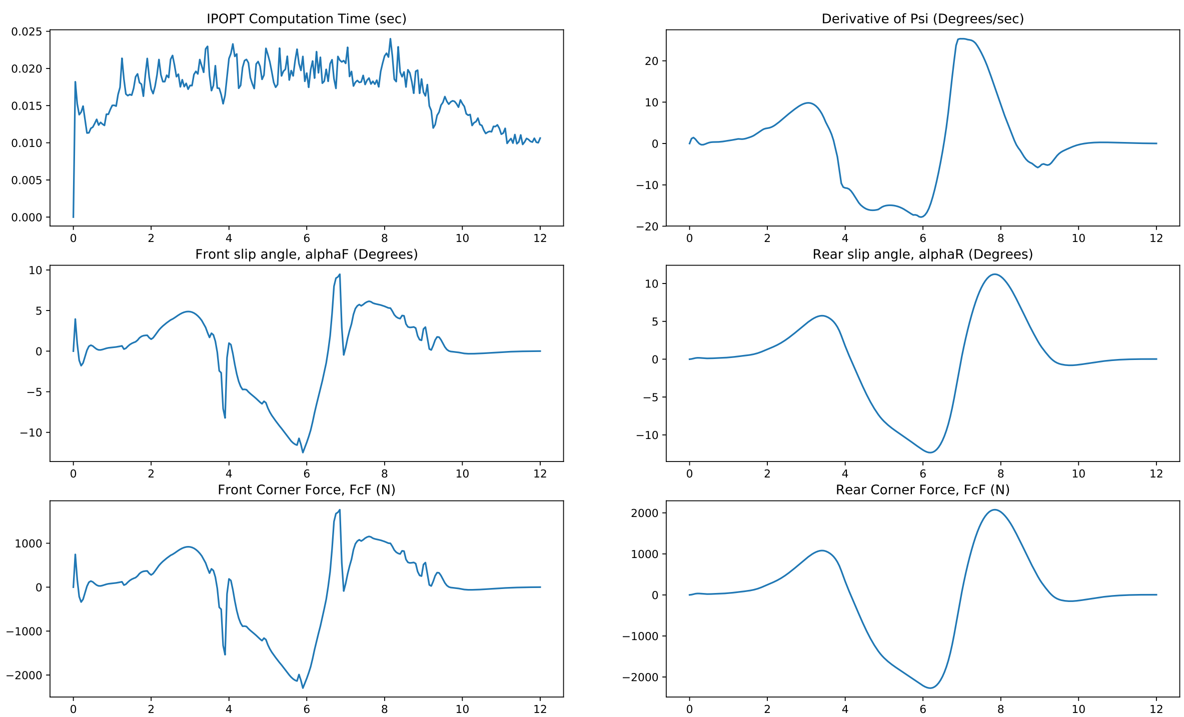

Figure 11: (c++) Double lane change maneuver at 10 m/s with $H_p = 20$ and $H_c = 20$. yaw rate, tire forces and slip angles.

References

[1] E. Bakker, L. Nyborg, and H. B. Pacejka. Tyre modelling for use in vehicle dynamics studies. In SAE Technical Paper. SAE International, 02 1987.

[2] F. Borrelli, P. Falcone, T. Keviczky, J. Asgari, and D. Hrovat. Mpc-based approach to active steering for autonomous vehicle systems. International Journal of Vehicle Autonomous Systems, 3:265–291, 2005.

[3] J. Kong, M. Pfeiffer, G. Schildbach, and F. Borrelli. Kinematic and dynamic vehicle models for autonomous driving control design. In 2015 IEEE Intelligent Vehicles Symposium (IV), pages 1094–1099, 2015.

[4] R. Rajamani. Vehicle Dynamics and Control. Mechanical Engineering Series. Springer US, 2011.