深蓝学院
shenlanxueyuan.com

第五章作业分享

主讲人 dazhou

纲要

- MPC Cost Function
- MPC Equality Constraints

MPC Cost Function

- Objective 1: Keep close to reference values
- Objective 2: Avoid to actuate as much as possible
启动
- Objective 3: Enforce actuators smoothness in change

The objective function has different types of components:

- **Tracking** objectives (Reference from planner)
- **Comfort** objectives (Comfortability of vehicle motion, perceived danger)

Different weights can be applied to separate cost terms in order to favor a certain trade-off.

$$J = w_t \cdot J_{tracking} + w_c \cdot J_{comfort}$$

$\boxed{posion\ error, e_p + velocity\ error, e_v + heading\ error, e_\varphi}$

$\boxed{control\ input:\ steering, \delta, \dot{\delta} + accel, a, \dot{a}}$

MPC Cost Function

- Objective 1: Keep close to reference values

$$J = w_t \cdot J_{tracking}$$

$$\boxed{posion\ error, e_p + velocity\ error, e_v + heading\ error, e_\varphi}$$

```
/* TODO: Objective term 1: Keep close to reference values.*/
for (size_t t = 0; t < Np; t++)
{
    // See below in mpc_controller::Solve() for reference of vars, Np, and Nc
    // (1) error
    // NOTE: reported error: no suitable conversion function from "CppAD::AD<double>" to "double" exists
    CppAD::AD<double> e_cte = vars[cte_start + t] - ref_cte;
    CppAD::AD<double> e_epsi = vars[epsi_start + t] - ref_epsi;
    CppAD::AD<double> e_v_longitudinal = vars[v_longitudinal_start + t] - ref_v;    // ref_v should be the longitudinal velocity
    // (2) quadratic cost function
    fg[0] += cte_weight * pow(e_cte, 2);
    fg[0] += epsi_weight * pow(e_epsi, 2);
    fg[0] += v_weight * pow(e_v_longitudinal, 2);
}
```

MPC Cost Function

- Objective 2: Avoid to actuate as much as possible
- Objective 3: Enforce actuators smoothness in change

$w_c \cdot J_{comfort}$

$\boxed{control\ input:\ steering, \delta, \dot{\delta} + accel, a, \dot{a}}$

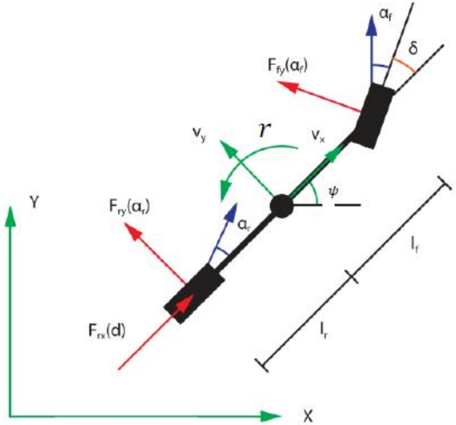
```
/* TODO: Objective term 2: Avoid to actuate as much as possible, minimize the use of actuators.*/
for (size_t t = 0; t < Nc; t++)
{
    fg[0] += steer_actuator_cost_weight_fg * pow(vars[front_wheel_angle_start + t], 2);
    fg[0] += acc_actuator_cost_weight_fg * pow(vars[longitudinal_acceleration_start + t], 2);
}
/* TODO: Objective term 3: Enforce actuators smoothness in change, minimize the value gap between sequential actuation.*/
for (size_t t = 0; t < Nc - 1; t++)
{
    // (1) change
    // NOTE: front_wheel_angle_increment_start & longitudinal_acceleration_increment_start are commented?
    CppAD::AD<double> change_steer = vars[front_wheel_angle_start + t + 1] - vars[front_wheel_angle_start + t];
    CppAD::AD<double> change_accel = vars[longitudinal_acceleration_start + t + 1] - vars[longitudinal_acceleration_start + t];
    // (2) quadratic cost function
    fg[0] += change_steer_cost_weight * pow(change_steer, 2);
    fg[0] += change_accel_cost_weight * pow(change_accel, 2);
}
```

纲要

- MPC Cost Function
- MPC Equality Constraints

Dynamic Bicycle Model

- 全局坐标系位置约束 x, y
- 航向角约束 ψ
- 车辆纵向速度约束 v_x
- 车辆侧向速度约束 v_y
- 车辆横摆角速度约束 r



$$
\begin{aligned}
\dot{x} &= v_x \cos(\psi) - v_y \sin(\psi) \\
\dot{y} &= v_x \sin(\psi) + v_y \cos(\psi) \\
\dot{\varphi} &= r \\
\dot{v_x} &= \frac{1}{m}\left(F_{r,x} - F_{f,y} \sin(\delta) + m v_y r\right) \\
\dot{v_y} &= \frac{1}{m}\left(F_{r,y} + F_{f,y} \cos(\delta) - m v_x r\right) \\
\dot{r} &= \frac{1}{I_z}\left(F_{f,y} l_f \cos(\delta) - F_{r,y} l_r\right)
\end{aligned}
$$

Dynamic Bicycle Model

MPC Equality Constraints

- 全局坐标系位置约束　　x, y

$$
\begin{gathered}
\dot{x} = v_x \cos\psi - v_y \sin\psi \\
\dot{y} = v_x \sin\psi + v_y \cos\psi \\
x_{k+1} = x_k + \dot{x}\Delta t \\
y_{k+1} = y_k + \dot{y}\Delta t
\end{gathered}
$$

```
// Approach II: Reference to Eq 1, 2 of "Dynamic Bicycle Model" in Lecture Slide 5 P33
CppAD::AD<double> vx_in_world_frame = v_longitudinal_0 * cos(psi_0) + v_lateral_0 * (-sin(psi_0));
CppAD::AD<double> vy_in_world_frame = v_longitudinal_0 * sin(psi_0) + v_lateral_0 * cos(psi_0);
fg[1 + x_start + t] = x_1 - (x_0 + vx_in_world_frame * dt);
fg[1 + y_start + t] = y_1 - (y_0 + vy_in_world_frame * dt);
```

MPC Equality Constraints

- 航向角约束 ψ

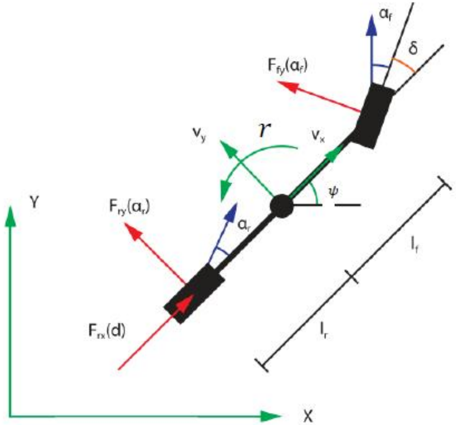
$$\dot{\psi} = \frac{v}{L}\tan\delta$$

Kinematic Bicycle Model

$$\dot{\psi} = r$$

$$\psi_{k+1} = \psi_k + \dot{\psi}\Delta t$$

```
// heading angle constraint is psi_1 = psi_0 + yaw_rate * dt;
fg[1 + psi_start + t] = psi_1 - (psi_0 + yaw_rate_0 * dt);
```



$$\dot{x} = v_x\cos(\psi) - v_y\sin(\psi)$$

$$\dot{y} = v_x\sin(\psi) + v_y\cos(\psi)$$

$$\dot{\varphi} = r$$

$$\dot{v_x} = \frac{1}{m}(F_{r,x} - F_{f,y}\sin(\delta) + mv_y r)$$

$$\dot{v_y} = \frac{1}{m}(F_{r,y} + F_{f,y}\cos(\delta) - mv_x r)$$

$$\dot{r} = \frac{1}{I_z}(F_{f,y}l_f\cos(\delta) - F_{r,y}l_r)$$

Dynamic Bicycle Model

MPC Equality Constraints

- 车辆纵向速度约束 v_x

$$\dot{v}_x = \frac{1}{m}(F_{xr} - F_{fy}\sin\delta + m v_y r)$$

$$\dot{v}_x = a_x$$
$$v_{x_{k+1}} = v_{x_k} + \dot{v}_x \Delta t$$

```
// longitudinal velocity constraint is v_long_1 = v_long_0 + a_long * dt
fg[1 + v_longitudinal_start + t] = v_longitudinal_1 - (v_longitudinal_0 + longitudinal_acceleration_0 * dt);
```

MPC Equality Constraints

- 车辆侧向速度约束 v_y

$$\dot{v_y} = \frac{1}{m}(F_{yr} - F_{fy}\cos\delta - mv_x r)$$

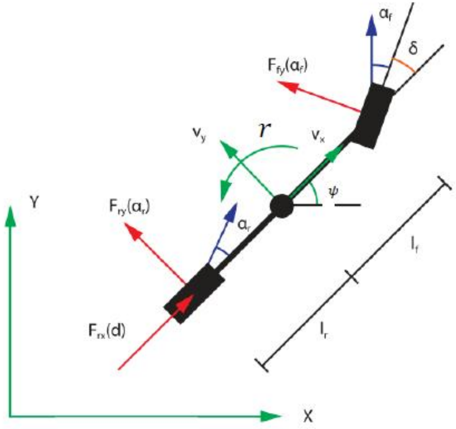
$$F_{yf} = 2c_f(\delta - \theta_{vf})$$

$$F_{yr} = 2c_r(-\theta_{vr})$$

$$\theta_{vf} = \tan^{-1}(\frac{v_y + l_f r}{v_x})$$

$$\theta_{vf} = \tan^{-1}(\frac{v_y + l_f r}{v_x})$$

$$v_{y_{k+1}} = v_{y_k} + \dot{v_y}\Delta t$$



$$\dot{x} = v_x\cos(\psi) - v_y\sin(\psi)$$
$$\dot{y} = v_x\sin(\psi) + v_y\cos(\psi)$$
$$\dot{\varphi} = r$$
$$\dot{v_x} = \frac{1}{m}(F_{r,x} - F_{f,y}\sin(\delta) + mv_y r)$$
$$\dot{v_y} = \frac{1}{m}(F_{r,y} + F_{f,y}\cos(\delta) - mv_x r)$$
$$\dot{r} = \frac{1}{I_z}(F_{f,y}l_f\cos(\delta) - F_{r,y}l_r)$$

Dynamic Bicycle Model

MPC Equality Constraints

深蓝学院
shenlanxueyuan.com

- 车辆侧向速度约束 v_y

$$\dot{v_y} = \frac{1}{m}\left(F_{yr} - F_{fy}\cos\delta - m v_x r\right),\ F_{yf} = 2c_f\left(\delta - \theta_{vf}\right),\ F_{yr} = 2c_r\left(-\theta_{vr}\right)$$

$$\theta_{vf} = \tan^{-1}\left(\frac{v_y + l_f r}{v_x}\right),\ \theta_{vf} = \tan^{-1}\left(\frac{v_y + l_f r}{v_x}\right)$$

$$v_{y_{k+1}} = v_{y_k} + \dot{v_y}\Delta t$$

```
CppAD::AD<double> delta = -front_wheel_angle_0;
/* 车辆侧向速度 */
// Reference to Eq 5 of "Dynamic Bicycle Model" in Lecture Slide 5 P33
// (1) tire forces
// Reference to Lecture Slide 4 P11 "Front and Rear Tire Forces"
CppAD::AD<double> theta_vf = atan((v_lateral_0 + lf * yaw_rate_0) / v_longitudinal_0);
CppAD::AD<double> F_yf = 2 * Cf * (delta - theta_vf);
CppAD::AD<double> theta_vr = atan((v_lateral_0 - lr * yaw_rate_0) / v_longitudinal_0);
CppAD::AD<double> F_yr = 2 * Cr * (-theta_vr);
// (2) Derive tire forces into Eq 5 to obtain a_lateral, denoted as dot_v_y in Eq 5
CppAD::AD<double> lateral_acceleration = (1 / m) * (F_yr + F_yf * cos(delta) - m * v_longitudinal_0 * yaw_rate_0);
// (3) lateral velocity constraint is v_lateral_1 = v_lateral_0 + a_lateral * dt
fg[1 + v_lateral_start + t] = v_lateral_1 - (v_lateral_0 + lateral_acceleration * dt);
```

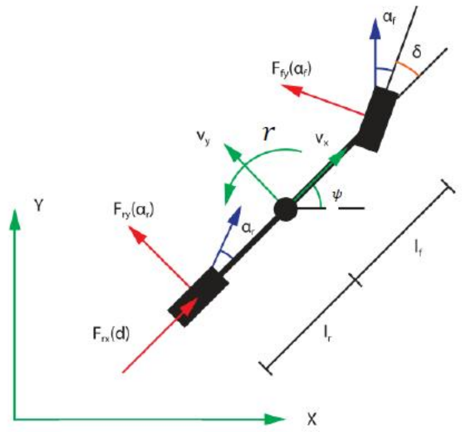
MPC Equality Constraints

- 车辆横摆角速度约束 r

$$\dot{r} = \frac{1}{I_z}(F_{yf} l_f \cos\delta - F_{fy} l_r)$$

$$r_{k+1} = r_k + \dot{r}\Delta t$$

```
/* 车辆横摆角速度 */
// Reference to Eq 6 of "Dynamic Bicycle Model" in Lecture Slide 5 P33
// (1) Derive tire forces into Eq 6 to obtain yaw_rate_rate, denoted as dot_r in Eq 6
CppAD::AD<double> yaw_rate_rate = (1 / I) * (F_yf * lf * cos(delta) - F_yr * lr);
// (2) yaw rate constraint is yaw_rate_1 = yaw_rate_0 + yaw_rate_rate * dt
fg[1 + yaw_rate_start + t] = yaw_rate_1 - (yaw_rate_0 + yaw_rate_rate * dt);
```



$$\dot{x} = v_x\cos(\psi) - v_y\sin(\psi)$$

$$\dot{y} = v_x\sin(\psi) + v_y\cos(\psi)$$

$$\dot{\varphi} = r$$

$$\dot{v_x} = \frac{1}{m}(F_{r,x} - F_{f,y}\sin(\delta) + m v_y r)$$

$$\dot{v_y} = \frac{1}{m}(F_{r,y} + F_{f,y}\cos(\delta) - m v_x r)$$

$$\dot{r} = \frac{1}{I_z}(F_{f,y} l_f \cos(\delta) - F_{r,y} l_r)$$

Dynamic Bicycle Model

在线问答

Q&A

深蓝学院
shenlanxueyuan.com

感谢各位聆听！

Thanks for Listening