MODEL PREDICTIVE CONTROL

HYBRID MPC EXAMPLES

Alberto Bemporad

http://cse.lab.imtlucca.it/~bemporad
Course Structure

- Linear model predictive control (MPC)
- Linear time-varying and nonlinear MPC
- MPC computations: quadratic programming (QP), explicit MPC
  - Hybrid MPC
  - Stochastic MPC
  - Data-driven MPC

MATLAB Toolboxes:
  - MPC Toolbox (linear/explicit/parameter-varying MPC)
  - Hybrid Toolbox (explicit MPC, hybrid systems)

Course page:
http://cse.lab.imtlucca.it/~bemporad/mpc_course.html
HYBRID MPC EXAMPLES
GOAL:

Command gear ratio, gas pedal, and brakes to track a desired speed and minimize fuel consumption

Disclaimer: This is an academic example
**HYBRID MPC FOR CRUISE CONTROL - MODEL**

- **Vehicle dynamics**

  \[
m \ddot{x} = F_e - F_b - \beta \dot{x}
  \]

  \(\dot{x} = \) vehicle speed  
  \(F_e = \) traction force  
  \(F_b = \) brake force  

  discretized with sampling time  \(T_s = 0.5 \text{ s}\)

- **Transmission kinematics**

  \[
  \omega = \frac{R_g(i)}{k_s} \dot{x}
  \]

  \(\omega = \) engine speed  
  \(C = \) engine torque  
  \(i = \) gear

  \[
  F_e \dot{x} = C \omega
  \]

  power balance:
• **Gear selection:** for each gear #i, define a binary input 
  
  \[ i = R, 1, 2, 3, 4, 5 \]

• **Gear selection (traction force):**

\[ F_e = \frac{R_g(i)}{k_s}C \]

defines auxiliary continuous variables:

\[ \text{IF } g_i = 1 \text{ THEN } F_{ei} = \frac{R_g(i)}{k_s}C \text{ ELSE 0} \]

\[ F_e = F_{eR} + F_{e1} + F_{e2} + F_{e3} + F_{e4} + F_{e5} \]

• **Gear selection (engine/vehicle speed):**

\[ \omega = \frac{R_g(i)}{k_s} \dot{x} \]

similarly, also requires 6 auxiliary continuous variables
HYBRID MPC FOR CRUISE CONTROL - MODEL

- engine torque \[-C_e^-(\omega) \leq C \leq C_e^+(\omega)\]

- max engine torque \[C_e^+(\omega)\]

Piecewise-linearization
(PWL Toolbox, Julián, 2000)

requires: 4 binary aux variables
4 continuous aux variables

- Min engine torque \[C_e^-(\omega) = \alpha_1\omega + \beta_1\]

Note: in this case the PWL constraint \[C \leq C_e^+(\omega)\] is convex, it could be handled by linear constraints without introducing any binary variable!
SYSTEM cruisecontrolmodel {

INTERFACE {
    PARAMETER {
        REAL mass = 1020; /* kg */
        REAL beta_friction = 25; /* W/m*s */
    }
    STATE {
        REAL position [0,10000];
        REAL speed [vmin,vmax];
    }
    INPUT {
        REAL torque [Cmin,Cmax];
        REAL F_brake [0,max_brake_force];
        BOOL gear1, gear2, gear3, gear4, gear5, gearR;
    }
}

IMPLEMENTATION {
    AUX {
        REAL F, Fe1, Fe2, Fe3, Fe4, Fe5, FeR;
        REAL w, w1, w2, w3, w4, w5, wR;
        BOOL dPWL1, dPWL2, dPWL3, dPWL4;
        REAL DCe1, DCe2, DCe3, DCe4;
    }
    LINEAR {
        F = Fe1+Fe2+Fe3+Fe4+Fe5+FeR;
        w = w1+w2+w3+w4+w5+wR;
    }
    AD {
        dPWL1 = wPWL1-w<=0;
        dPWL2 = wPWL2-w<=0;
        dPWL3 = wPWL3-w<=0;
        dPWL4 = wPWL4-w<=0;
    }
    DA {
        Fe1 = (IF gear1 THEN torque/speed_factor*Rgear1);
        Fe2 = (IF gear2 THEN torque/speed_factor*Rgear2);
        Fe3 = (IF gear3 THEN torque/speed_factor*Rgear3);
        Fe4 = (IF gear4 THEN torque/speed_factor*Rgear4);
        Fe5 = (IF gear5 THEN torque/speed_factor*Rgear5);
        FeR = (IF gearR THEN torque/speed_factor*RgearR);
        w1 = (IF gear1 THEN speed/speed_factor*Rgear1);
        w2 = (IF gear2 THEN speed/speed_factor*Rgear2);
        w3 = (IF gear3 THEN speed/speed_factor*Rgear3);
        w4 = (IF gear4 THEN speed/speed_factor*Rgear4);
        w5 = (IF gear5 THEN speed/speed_factor*Rgear5);
        wR = (IF gearR THEN speed/speed_factor*RgearR);
        DCe1 = (IF dPWL1 THEN (aPWL2-aPWL1)+(bPWL2-bPWL1)*w);
        DCe2 = (IF dPWL2 THEN (aPWL3-aPWL2)+(bPWL3-bPWL2)*w);
        DCe3 = (IF dPWL3 THEN (aPWL4-aPWL3)+(bPWL4-bPWL3)*w);
        DCe4 = (IF dPWL4 THEN (aPWL5-aPWL4)+(bPWL5-bPWL4)*w);
    }
    CONTINUOUS {
        position = position+Ts*speed;
        speed = speed+Ts/mass*(F-F_brake-beta_friction*speed);
        MUST {
            /* max engine speed */
            /* wemin <= w1+w2+w3+w4+w5+wR <= wemax */
            -w1 <= -wemin; w1 <= wemax;
            -w2 <= -wemin; w2 <= wemax;
            -w3 <= -wemin; w3 <= wemax;
            -w4 <= -wemin; w4 <= wemax;
            -w5 <= -wemin; w5 <= wemax;
            -wR <= -wemin; wR <= wemax;
        }
    }
}

HybridMPCforcruisecontrol-HYSDEL ©2019 A. Bemporad - "Model Predictive Control"

©2019 A. Bemporad - "Model Predictive Control"
HYBRID MPC FOR CRUISE CONTROL - MLD MODEL

- MLD model

\[
\begin{align*}
  x(t+1) &= Ax(t) + B_1 u(t) + B_2 \delta(t) + B_3 z(t) + B_5 \\
  y(t) &= C x(t) + D_1 u(t) + D_2 \delta(t) + D_3 z(t) + D_5 \\
  E_2 \delta(t) + E_3 z(t) &\leq E_4 x(t) + E_1 u(t) + E_5
\end{align*}
\]

- 2 continuous states: \( x, v \) (vehicle position and speed)
- 2 continuous inputs: \( C, F_b \) (engine torque, brake force)
- 6 binary inputs: \( g_R, g_1, g_2, g_3, g_4, g_5 \) (gears)
- 1 continuous output: \( v \) (vehicle speed)
- 18 auxiliary continuous vars: (6+1 traction force, 6+1 engine speed, 4 PWL max engine torque)
- 4 auxiliary binary vars: (PWL max engine torque breakpoints)
- 100 mixed-integer inequalities
• **Max-speed controller**

\[
\max_{u_t} J(u_t, x(t)) \triangleq v(t + 1|t)
\]

subject to

\[
\begin{align*}
\text{MLD model} \\
\quad x(t|t) &= x(t)
\end{align*}
\]

**Objective:** maximize speed

(to reproduce max acceleration plots)

- **MILP optimization problem**
  - Linear constraints: 96
  - Continuous variables: 18
  - Binary variables: 10
  - Parameters: 1
  - Time to solve mp-MILP (Sun Ultra 10): 45 s
  - **Number of regions**: 11

(parameters: Renault Clio 1.9 DTI RXE)
• **Max-speed controller**

![Graphs showing the performance of a maximal speed controller in a hybrid system. The graphs depict velocity, gear, fraction of max torque, and brakes over time.]
• Tracking controller

\[
\min_{u_t} \quad J(u_t, x(t)) \triangleq |v(t + 1|t) - v_d(t)| + \rho|\omega|
\]

s.t. \[
\begin{align*}
\text{MLD model} \\
x(t|t) &= x(t)
\end{align*}
\]

MILP optimization problem

<table>
<thead>
<tr>
<th>Linear constraints</th>
<th>98</th>
</tr>
</thead>
<tbody>
<tr>
<td>Continuous variables</td>
<td>19</td>
</tr>
<tr>
<td>Binary variables</td>
<td>10</td>
</tr>
<tr>
<td>Parameters</td>
<td>2</td>
</tr>
<tr>
<td>Time to solve mp-MILP (PC 850Mhz)</td>
<td>43 s</td>
</tr>
<tr>
<td>Number of regions</td>
<td>49</td>
</tr>
</tbody>
</table>

go to demo /demos/cruise/init_exp.m
• Tracking controller

\[
\min_{u_t} |v(t + 1|t) - v_d(t)| + \rho |\omega| \quad \rho = 0.001
\]
• Smoother tracking controller

\[
\min_{u_t} J(u_t, x(t)) \triangleq |v(t + 1|t) - v_d(t)| + \rho |\omega|
\]

s.t.

\[
\begin{align*}
|v(t + 1|t) - v(t)| & \leq a_{\max} T_s \\
x(t|t) &= x(t)
\end{align*}
\]

MILP optimization problem

<table>
<thead>
<tr>
<th>Linear constraints</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Continuous variables</td>
<td>19</td>
</tr>
<tr>
<td>Binary variables</td>
<td>10</td>
</tr>
<tr>
<td>Parameters</td>
<td>2</td>
</tr>
<tr>
<td>Time to solve mp-MILP (PC 850Mhz)</td>
<td>47 s</td>
</tr>
</tbody>
</table>

Number of regions 54
• Smoother tracking controller
HYBRID MPC FOR TRACTION CONTROL
VEHICLE TRACTION CONTROL PROBLEM

Improve driver's ability to control a vehicle under adverse external conditions (wet or icy roads)

**Model:** nonlinear, uncertain, constraints

**Controller:** suitable for real-time implementation

**Solution:** MLD hybrid framework + explicit hybrid MPC strategy
TIRE FORCE CHARACTERISTICS

- Lateral Force
- Longitudinal Force
- Slip Target Zone

Maximum Braking | Maximum Cornering | Maximum Acceleration | Tire Slip

Steer Angle

©2019 A. Bemporad - "Model Predictive Control"
Simple traction model

- Tire torque \( \tau_t \) is a function of slip \( \Delta \omega \) and road surface adhesion coefficient \( \mu \)

- Mechanical system

\[
\dot{\omega_e} = \frac{1}{J_e} \left( \tau_c - b_e \omega_e - \frac{\tau_t}{gr} \right)
\]

\[
\dot{v}_v = \frac{\tau_t}{m_v r_t}
\]

- Manifold/fueling dynamics

\[
\tau_c = b_i \tau_d(t - \tau_f)
\]

\[
v_t = \omega_t r_t = \frac{\omega_e}{gr} r_t
\]

\[
\Delta \omega = \frac{1}{r_t} (v_t - v_v) = \frac{\omega_e}{gr} - \frac{v_v}{r_t}
\]

wheel slip

©2019 A. Bemporad - "Model Predictive Control"
Nonlinear tire torque
\[ \tau_t = f(\Delta \omega, \mu) \]

Hybrid model

PWA approximation

Mixed-Logical Dynamical (MLD) Hybrid Model (discrete time)

HYSDEL

©2019 A. Bemporad - "Model Predictive Control"
The MLD Traction Model

\[ x(t + 1) = Ax(t) + B_1 u(t) + B_2 \delta(t) + B_3 z(t) + B_5 \]
\[ y(t) = Cx(t) + D_1 u(t) + D_2 \delta(t) + D_3 z(t) + D_5 \]
\[ E_2 \delta(t) + E_3 z(t) \leq E_4 x(t) + E_1 u(t) + E_5 \]

- **State**: \( x(t) \in \mathbb{R}^4 \)
- **Input**: \( u(t) \in \mathbb{R} \)
- **Aux. Binary**: \( \delta(t) \in \{0, 1\} \)
- **Aux. Continuous**: \( z(t) \in \mathbb{R}^3 \)

**Number of Mixed-Integer Inequalities** = 14

The MLD matrices are automatically generated in MATLAB format by HYSDEL
Performance and constraints

• Control objective:

\[
\min \sum_{k=0}^{N} |\Delta \omega(t + k|t) - \Delta \omega_{\text{des}}| \\
\text{s.t. } \text{MLD dynamics}
\]

• Constraints:
  • Limits on the engine torque:
    
    \[-20 \text{ Nm} \leq \tau_d \leq 176 \text{ Nm}\]
Experimental results

controller is triggered ON

(250 ms delay from commanded to actual engine torque → initial overspin)
Experiments

- 504 regions
- 20ms sampling time
- Pentium 266Mhz + Labview

Indoor ice arena

(μ ≈ 0.2)

2000 Ford Focus
2.0l 4-cyl engine
5-speed manual transmission

©2019 A. Bemporad - "Model Predictive Control"
HYBRID CONTROL OF A DISC ENGINE
Objective: develop a controller for a Direct-Injection Stratified Charge (DISC) engine that:

- automatically chooses operating mode (homogeneous/stratified)
- can cope with nonlinear dynamics
- handles constraints on A/F ratio, air-flow, spark
- achieves optimal performance (track desired torque and A/F ratio)
Two distinct regimes:

<table>
<thead>
<tr>
<th>Regime</th>
<th>Fuel Injection</th>
<th>Air-to-Fuel Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Homogeneous combustion</td>
<td>Intake stroke</td>
<td>$\lambda=14.64$</td>
</tr>
<tr>
<td>Stratified combustion</td>
<td>Compression stroke</td>
<td>$\lambda&gt;14.64$</td>
</tr>
</tbody>
</table>

- Mode is switched by changing fuel injection timing (late / early)
- Better fuel economy during stratified mode

Periodical cleaning of the aftertreatment system needed ($\lambda=14.00$, homogeneous regime)

the stratified operation can only be sustained in a restricted part of the engine operating range
DISC ENGINE

- **States**: intake manifold pressure ($p_m$)

- **Outputs**: Air-to-fuel ratio ($\lambda$), torque ($\tau$), max-brake-torque spark timing ($\delta_{mbt}$)

- **Continuous inputs**: spark advance ($\delta$), air flow ($W_{th}$), fuel flow ($W_f$)

- **Binary input**: spark combustion regime ($\rho$)

- **Disturbance**: engine speed ($\omega$) [measured]

- **Constraints on**:
  - Air-to-fuel ratio (due to engine roughness, misfiring, smoke emiss.)
  - Spark timing (to avoid excessive engine roughness)
  - Mass flow rate on intake manifold (constraints on throttle)

Dynamic equations are **nonlinear**, dynamics and constraints depend on regime $\rho$
Nonlinear model of the engine developed and validated at Ford (Kolmanovsky, Sun, ...)

Assumptions:
- no EGR (exhaust gas recirculation) rate,
- engine speed=2000 rpm.

- Intake manifold pressure:
  \[ p_m = c_m (W_{th} - W_{cyl}) = c_m (W_{th} - k_{cyl} p_m) \]

- In-cylinder Air-to-Fuel ratio:
  \[ \lambda = \frac{W_{cyl}}{W_f} = \frac{k_{cyl} p_m}{W_f} \]

- Engine torque:
  \[ \tau = \tau_{mf} + \tau_{pump} + \tau_{ind} \]
  \[ \tau_{ind} = (\theta_a + \theta_b (\delta - \delta_{mbt})^2) W_f \]
  with \( \tau_{mf}, \tau_{pump} \) functions of \( p_m \)
  where \( \theta_a, \theta_b, \delta_{mbt} \) are functions of \( \lambda, \delta \) and \( \rho \)

- Good for simulation
- Not suitable for optimization-based controller synthesis
HYBRIDIZATION OF DISC MODEL

**DYNAMICS** (intake pressure, air-to-fuel ratio, torque):

- Definition of two operating points;
- Linearization of nonlinear dynamics;
- Time discretization of the linear models.

**CONSTRAINTS** on:

- Air-to-Fuel Ratio: \( \lambda_{\text{min}}(\rho) \leq \lambda(t) \leq \lambda_{\text{max}}(\rho) \)
- Mass of air through the throttle: \( 0 \leq W_{\text{th}} \leq K \)
- Spark timing: \( 0 \leq \delta(t) \leq \delta_{\text{mbt}}(\lambda, \rho) \)

\( \rho \)-dependent dynamic equations

\( \rho \)-dependent constraints

Hybrid system with 2 modes (switching affine system)
Integral action

Integrators on torque error and air-to-fuel ratio error are added to obtain zero offsets in steady-state:

\[
\begin{align*}
\epsilon_{\tau,k+1} &= \epsilon_{\tau,k} + T_s (\tau_{\text{ref}}(t) - \tau_k) \\
\epsilon_{\lambda,k+1} &= \epsilon_{\lambda,k} + T_s (\lambda_{\text{ref}}(t) - \lambda_k)
\end{align*}
\]

\(T_s = \text{sampling time}\)

\(\tau_{\text{ref}}, \lambda_{\text{ref}} = \text{references on brake torque and air-to-fuel ratio}\)

Simulation based on nonlinear model confirms zero offsets in steady-state (despite the model mismatch)
MPC of DISC engine

\[
\min_{\xi} J(\xi, x(t)) = \sum_{k=0}^{N-1} u'_k Ru_k + y'_k Q y_k + x'_{k+1} S x_{k+1}
\]

\[
\text{subj. to } \begin{cases} 
x_0 = x(t), \\
\text{hybrid model}
\end{cases}
\]

\[
\xi = [u'_0, \gamma'_0, z'_0, \ldots, u'_{N-1}, \gamma'_{N-1}, z'_{N-1}]',
\]

where:

\[
u_k = [W_{th,k} - W_{th,ref}, W_{f,k} - W_{f,ref}, \delta_k - \delta_{ref}, \rho_k - \rho_{ref}]'
\]

\[
y_k = [\tau_k - \tau_{ref}, \lambda_k - \lambda_{ref}, \delta_{mbt} - \delta_k - \Delta \delta_{ref}]'
\]

\[
x_k = [p_{m,k} - p_{m,ref}, \epsilon_{\tau,k}, \epsilon_{\lambda,k}]'
\]

and:

\[
R = \begin{pmatrix} 
r_{W_{th}} & 0 & 0 & 0 \\ 0 & r_{W_f} & 0 & 0 \\ 0 & 0 & r_{\delta} & 0 \\ 0 & 0 & 0 & r_{\rho} \end{pmatrix}, \quad Q = \begin{pmatrix} q_{\tau} & 0 & 0 \\ 0 & q_{\lambda} & 0 \\ 0 & 0 & q_{\Delta \delta} \end{pmatrix}, \quad S = \begin{pmatrix} s_{p_m} & 0 & 0 \\ 0 & s_{\epsilon_{\tau}} & 0 \\ 0 & 0 & s_{\epsilon_{\lambda}} \end{pmatrix}
\]

Reference values are automatically generated from $\tau_{ref}$ and $\lambda_{ref}$ by numerical computations based on the nonlinear model.
DISC ENGINE - HYSDEL

SYSTEM hysdisc{
    INTERFACE{
        STATE{
            REAL pm [1, 101.325];
            REAL xtau [-1e3, 1e3];
            REAL xlam [-1e3, 1e3];
            REAL tau [0, 100];
            REAL lam [10, 60];
        }
        OUTPUT{
            REAL lambda, tau, ddelta;
        }
        INPUT{
            REAL Wth [0, 38.5218];
            REAL Wf [0, 2];
            REAL delta [0, 40];
            BOOL rho;
        }
        PARAMETER{
            REAL Ts, pm1, pm2;
            ...
        }
    } IMPLEMENTATION{
        AUX{
            REAL lam, tau1, dmbtl, lmin, lmax;
        }
        DA{
            lam = IF rho THEN 111*pm+112*Wth...
                +113*Wf+114*delta+11c
                ELSE 101*pm+102*Wth+103*Wf...
                +104*delta+10c
            tau1 = IF rho THEN tau11*pm+...
                tau12*Wth+tau13*Wf+tau14*delta+tau1c
                ELSE tau01*pm+tau02*Wth...
                +tau03*Wf+tau04*delta+tau0c)
            dmbtl = IF rho THEN dmbt11*pm+dmbt12*Wth...
                +dmbt13*Wf+dmbt14*delta+dmbt1c+7
                ELSE dmbt01*pm+dmbt02*Wth...
                +dmbt03*Wf+dmbt04*delta+dmbt0c-1;
            lmin = IF rho THEN 13 ELSE 19;
            lmax = IF rho THEN 21 ELSE 38;
        }
        CONTINUOUS{
            pm = pm1*pm+pm2*Wth;
            xtau = xtau+Ts*(taud-taul);
            xlam = xlam+Ts*(lamd-lam);
            tau = taud; lam = lamd;
        }
        OUTPUT{
            lambda = lam-lamd;
            tau = tau1-taud;
            ddelta = dmbtl-delta;
        }
        MUST{
            lmin-lam <= 0;
            lam-lmax <= 0;
            delta-dmbtl <= 0;
        }
    }
}
MPC — TORQUE CONTROL MODE

\[
\begin{align*}
\text{min} & \sum_{k=0}^{N-1} (u_k - u_r)'R(u_k - u_r) + (y_k - y_r)'Q(y_k - y_r) \\
& \quad + (x_{k+1} - x_r)'S(x_{k+1} - x_r)
\end{align*}
\]

subj. to \( \begin{cases} x_0 = x(t), \\ \text{hybrid model} \end{cases} \)

\[u(t) = [W_{th}(t), W_f(t), \delta(t), \rho(t)]\]

Weights:

\[
R = \begin{pmatrix}
0.01 & 0 & 0 & 0 \\
0 & 0.001 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1
\end{pmatrix}
\]

(prevents unneeded chattering)

\[
Q = \begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & 0.001 & 0 & 0 \\
0 & 0 & 0.01 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix}
\]

\[
S = \begin{pmatrix}
0.04 & 0 & 0 & 0 \\
0 & 1500 & 0 & 0 \\
0 & 0 & 0.01 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix}
\]

main emphasis on torque

Solve MIQP problem to compute \(u(t)\)
Simulation results (nominal engine speed)

**Engine brake torque**  \(\omega/2000\) rpm

**Air-to-fuel ratio**

**Combustion mode**

- Control horizon \(N=1\);
- Sampling time \(Ts=10\) ms;
- PC Xeon 2.8 GHz + Cplex 9.1

\[\approx 3\text{ ms per time step}\]
Simulation results (varying engine speed)

Hybrid MPC design is quite robust with respect to engine speed variations

20 s segment of the European drive cycle (NEDC)
Simulation results (varying engine speed)

Air-to-Fuel Ratio

Engine Brake Torque

Control code too complex (MIQP) → not implementable!
Explicit control law:

$$u(t) = f(\theta(t))$$

where:

$$u = [W_{th} \ W_f \ \delta \ \rho]'$$

$$\theta = [p_m \ \epsilon_T \ \epsilon_\lambda \ \tau_{ref} \ \lambda_{ref} \ \rho_{m,ref} \ W_{th,ref} \ W_{f,ref} \ \delta_{ref}]'$$

Cross-section by the $\tau_{ref}-\lambda_{ref}$ plane

- Time to compute explicit MPC:
  $$\approx 3s;$$
- Sampling time $T_s=10 \text{ ms};$
- PC Xeon 2.8 GHz + Cplex 9.1
  $$\rightarrow 8 \mu s \text{ per time step}$$

$$\approx 3\text{ms on}$$

$\mu-$controller
Motorola
MPC 555
43kb RAM
(custom made for Ford)
Explicit control law: $u(t) = f(\theta(t))$

where: $u = [W_{th} \ W_f \ \delta \ \rho]'$

$\theta = [p_m \ \epsilon_T \ \epsilon_\lambda \ \tau_{ref} \ \lambda_{ref} \ \rho_{m,ref} \ \W_{th,ref} \ \W_{f,ref} \ \delta_{ref}]'$

N=2 (control horizon)

747 partitions

Closed-loop N=2

Closed-loop N=1

adequate!
EXPLICIT HYBRID MPC OF SEMIACTIVE SUSPENSIONS
Active Suspensions

Active Suspension System
Ford Mercur XR 40i

active suspensions

passive suspensions
For Semi-Active with Variable Damping, $f(x) = C^* (x_4 - x_2)$

$C = \frac{f(x)}{(x_4 - x_2)}$, where $f(x)$ is the optimal active suspension force

$C = \text{sat}[\frac{f(x)}{(x_4 - x_2)}]$
• State-space model

\[
\dot{x} = Ax + B \bar{f} + B_ww
\]

- \( A = \begin{bmatrix}
0 & 1 & 0 & 0 \\
-\omega_{us}^2 & -2\rho\zeta\omega_s & \rho\omega_s^2 & 2\rho\zeta\omega_s \\
0 & -1 & 0 & 1 \\
0 & 2\zeta\omega_s & -\omega_s^2 & -2\zeta\omega_s \\
\end{bmatrix} \)
- \( x_1 = \) tire deflection from equilibrium
- \( x_2 = \) unsprung mass velocity
- \( x_3 = \) suspension deflection from equilibrium
- \( x_4 = \) sprung mass velocity
- \( \bar{f} = \) normalized adjustable force
- \( w = \) road velocity disturbance

\[\rho = \frac{M_s}{M_{us}}, \quad \omega_{us} = \sqrt{\frac{k_{us}}{M_{us}}}, \quad \omega_s = \sqrt{\frac{k_s}{M_s}}, \quad \zeta = \frac{\beta_s}{2\sqrt{M_sk_s}}, \quad \bar{f} = \frac{f}{M_s}\]

• Output:

\[
y = \frac{dx_4}{dt} = \begin{bmatrix} 0 & 2\zeta\omega_s & -\omega_s^2 & -2\zeta\omega_s \end{bmatrix} x - \bar{f}
\]

• Cost:

\[
J = \int (q_{x_1}x_1^2 + q_{x_3}x_3^2 + \dot{x}_4^2) dt
\]

\[
= \int (x'Qx + \dot{x}_4^2) dt
\]

• Time-discretization: \( T_s = 10 \text{ ms} \)
CONSTRAINTS ON SUSPENSION MODEL

Constraints:

1) Passivity condition:
   \[ f(x_4 - x_2) \geq 0 \]

2) Max dissipation power:
   \[ f(x_4 - x_2) \leq (2 \cdot 25.5 \cdot \omega_s)(x_4 - x_2)^2 \]

3) Saturation:
   \[ |f| \leq \sigma \]

(1), (2) are nonlinear & nonconvex physical constraints
1) Passivity condition:

\[ \bar{f}(x_4 - x_2) \geq 0 \]

- \( \bar{f} \) is ok:
  - \( \delta_v = 1 \) \( \leftrightarrow \) \( x_4 - x_2 \geq 0 \)
  - \( \delta_{\bar{f}} = 1 \) \( \leftrightarrow \) \( \bar{f} \geq 0 \)
  - \( \delta_v = 1 \) \( \rightarrow \) \( \delta_{\bar{f}} = 1 \)
  - \( \delta_v = 0 \) \( \rightarrow \) \( \delta_{\bar{f}} = 0 \)

- \( \bar{f} \) is no:

2) Max dissipation power:

\[ \bar{f}(x_4 - x_2) \leq (2 \cdot 25.5 \cdot \omega_s)(x_4 - x_2)^2 \]

- \( F \geq 0 \)

where

\[ F = \begin{cases} 
    \bar{f} - (2 \cdot 25.5 \cdot \omega_s)(x_4 - x_2) & \text{if } (x_4 - x_2) \leq 0 \\
    -\bar{f} + (2 \cdot 25.5 \cdot \omega_s)(x_4 - x_2) & \text{otherwise}
\end{cases} \]

3) Saturation:

\[ |\bar{f}| \leq \sigma \]

- \( \bar{f} \) is \( \leq \sigma \)
- \( \bar{f} \) is \( \geq -\sigma \)
get the MLD model in MATLAB

simulate the MLD model
Hybrid PWA model

- PWA model

\[
\begin{align*}
    x(k + 1) &= A_i(k)x(k) + B_i(k)u(k) + f_i(k) \\
    y(k) &= C_i(k)x(k) + D_i(k)u(k) + g_i(k) \\
    i(k) \text{ s.t. } H_i(k)x(k) + J_i(k)u(k) &\leq K_i(k)
\end{align*}
\]

- 4 continuous states
  \((x_1, x_2, x_3, x_4)\)

- 1 continuous input
  (normalized adjustable damping force \(f\))

- 2 polyhedral regions

\[
\text{>>P=pwa(S);}
\]
OPEN-LOOP SIMULATION OF PWA SUSPENSION MODEL

[Simulation diagram

©2019 A. Bemporad - "Model Predictive Control"
\[
\min \left( \sum_{k=1}^{N-1} \left( 1100x_{1,k}^2 + 100x_{3,k}^2 + \dot{x}_{4,k}^2 \right) + x_{N}'Px_N \right)
\]

- **tire deflection**
- **suspension deflection**
- **vertical acceleration**

**terminal weight** (Riccati matrix)
Closed-loop MPC results (command line)

```matlab
>> [XX, UU, DD, ZZ, TT] = sim(C, S, r, x0, Tstop);
```

```matlab
>> C = hybcon(S, Q, N, limits, refs);
```

Closed-loop MPC results

© 2019 A. Bemporad - "Model Predictive Control"
CLOSED-LOOP MPC RESULTS (SIMULINK)
\[ u(x) = \begin{cases} 
10.4748x_1 + 0.2446x_2 + 79.1519x_3 - 3.9235x_4 
= K_{LQ} 
0 
(2 \cdot 25.5 \cdot \omega_s)(x_4 - x_2) 
-1 
1 
\end{cases} \]

Explicit hybrid MPC

```
>> E = expcon(C, range, options);

>> E

Explicit controller (based on hybrid controller C)
4 parameter(s)
1 input(s)
8 partition(s)
sampling time = 0.01

The controller is for hybrid systems (tracking)
[2-norm]

This is a state-feedback controller.
Type "struct(E)" for more details.
```
Explicit hybrid MPC

generated C-code

```c
#define EXPCON_NU 1
#define EXPCON_NX 4
#define EXPCON_NY 1
#define EXPCON_TS 0.01000000
#define EXPCON_REG 8
#define EXPCON_NTH 4
#define EXPCON_NYM 4
#define EXPCON_NUC 1
#define EXPCON_HUB 0
#define EXPCON_N3XIN 1
#define EXPCON_HK 21
#define EXPCON_HP 8
static double EXPCON_F[]={
  10.4748,0,0,0,10.4748,0,0,0,-0.244554,0,
  480.564,0,
  3.92349,0,
  480.564,0
};

static double EXPCON_G[]={
  0,1e-006,-1e-006,-1,0,0,1e-006,1
};
```
### Quest of Optimal Semiactive Suspensions

#### Parameter Values Used in Simulation

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_s$</td>
<td>10 ms</td>
<td>Sampling time</td>
</tr>
<tr>
<td>$\omega_s$</td>
<td>1.5 Hz</td>
<td>Sprung mass natural frequency</td>
</tr>
<tr>
<td>$\omega_{us}$</td>
<td>10 Hz</td>
<td>Wheel-hop natural frequency</td>
</tr>
<tr>
<td>$\rho$</td>
<td>10</td>
<td>Sprung-to-unsprung mass ratio</td>
</tr>
<tr>
<td>$\zeta$</td>
<td>0</td>
<td>Damping ratio</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>1</td>
<td>Maximum force capacity</td>
</tr>
<tr>
<td>$q_1$</td>
<td>1100</td>
<td>Weight on tire deflection</td>
</tr>
<tr>
<td>$q_3$</td>
<td>100</td>
<td>Weight on suspension deflection</td>
</tr>
</tbody>
</table>

#### Table II

<table>
<thead>
<tr>
<th>$N$</th>
<th>MPC</th>
<th>Clipped LQR</th>
<th>SGM</th>
<th>LQR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20.4282</td>
<td>20.4282</td>
<td>17.4944</td>
<td>0.4446</td>
</tr>
<tr>
<td>2</td>
<td>20.4054</td>
<td>20.4282</td>
<td>17.4944</td>
<td>0.4446</td>
</tr>
<tr>
<td>3</td>
<td>20.3290</td>
<td></td>
<td>17.4944</td>
<td>0.4446</td>
</tr>
<tr>
<td>4</td>
<td>20.1100</td>
<td></td>
<td>17.4944</td>
<td>0.4446</td>
</tr>
<tr>
<td>5</td>
<td>19.7380</td>
<td></td>
<td>17.4944</td>
<td>0.4446</td>
</tr>
<tr>
<td>10</td>
<td>20.9840</td>
<td></td>
<td>17.4944</td>
<td>0.4446</td>
</tr>
<tr>
<td>12</td>
<td>19.3084</td>
<td></td>
<td>17.4944</td>
<td>0.4446</td>
</tr>
<tr>
<td>14</td>
<td>18.4842</td>
<td></td>
<td>17.4944</td>
<td>0.4446</td>
</tr>
<tr>
<td>15</td>
<td>18.5996</td>
<td></td>
<td>17.4944</td>
<td>0.4446</td>
</tr>
<tr>
<td>16</td>
<td>19.3212</td>
<td></td>
<td>17.4944</td>
<td>0.4446</td>
</tr>
<tr>
<td>20</td>
<td>18.0764</td>
<td></td>
<td>17.4944</td>
<td>0.4446</td>
</tr>
<tr>
<td>30</td>
<td>17.4494</td>
<td></td>
<td>17.4944</td>
<td>0.4446</td>
</tr>
<tr>
<td>40</td>
<td>17.1304</td>
<td></td>
<td>17.4944</td>
<td>0.4446</td>
</tr>
</tbody>
</table>
Simulation results

- Horizon $N=1$: same as Clipped-LQR!
- Better closed-loop performance for increasing $N$

Performance Index

<table>
<thead>
<tr>
<th>$N$</th>
<th>MPC</th>
<th>Clipped-LQR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.5155</td>
<td>1.5155</td>
</tr>
<tr>
<td>5</td>
<td>1.4416</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1.5238</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>1.3083</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>1.2204</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>1.1456</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>1.1462</td>
<td></td>
</tr>
</tbody>
</table>

- Simulations with road noise.
- Initial condition $x(0)=[0 0 0 0]'$
- Simulation time $T=20$ s, sampling time $T_s=10$ ms

N=1, same cost value!
VEHICLE YAW STABILITY CONTROL
Problem:
Control vehicle stability while tracking driver’s desired trajectory

- **Electronic Stability Control** (ESC)
  (Koibuchi et al., 1996)

- **Active Front Steering** (AFS)
  (Ackermann, 1997)

Main control objective:
Force the vehicle **yaw rate** to track a time-varying reference computed by the driver’s steering angle and the current vehicle velocity

Approach:
Consider the steer as a reference generator and actuate steering and differential braking (**coordinated AFS and ESC action**)
• **Bicycle model** appropriate in high speed turns (Gillespie, 1992)

- **reference frame** \((x,y,z)\) moving with the vehicle
- **front steering angle** \(\delta\) [rad]
- **tire slip angles** \(\alpha_f, \alpha_r\) [rad]
- **yaw rate** \(r\) [rad/s]

\[
\tan (\alpha_f + \delta) = \frac{v_y + ar}{v_x}, \quad \tan \alpha_r = \frac{v_y - br}{v_x}
\]
• **Tire force characteristics:** are nonlinear functions of the slip angles and of the longitudinal slip

\[
F_f(\alpha_f) = \begin{cases} 
-c_f \alpha_f & \text{if } -\hat{p}_f \leq \alpha_f \leq \hat{p}_f \\
-(d_f \alpha_f + e_f) & \text{if } \alpha_f > \hat{p}_f 
\end{cases}
\]

\[
F_r(\alpha_r) = \begin{cases} 
-c_r \alpha_r & \text{if } -\hat{p}_r \leq \alpha_r \leq \hat{p}_r \\
-(d_r \alpha_r + e_r) & \text{if } \alpha_r > \hat{p}_r 
\end{cases}
\]

• **Critical slip angles** \( \hat{p}_f, \hat{p}_r \) are threshold values where dynamics switch

• For symmetry, we can restrict to analyze clockwise turns (counter-clockwise turns can be handled by opportuneely inverting signs)
Sideslip angle-force characteristics

(a) Front tires

(b) Rear tires

Experimental tire data and piecewise linear approximation of the tire

Rear-wheel drive test vehicle equipped with active front steering and differential braking used for experimental validation
VEHICLE DYNAMICAL MODEL

slip angles

\[ \dot{\alpha}_f = \frac{\dot{v_y} + a \dot{r}}{v_x} - \dot{\delta} \]
\[ \dot{\alpha}_r = \frac{\dot{v}_y - b \dot{r}}{v_x} \]

static yaw rate

\[ r = \frac{v_x}{a + b} (\alpha_f - \alpha_r + \delta) \]

overall dynamical model

\[ \dot{\alpha}_f = \frac{F_f + F_r}{mv_x} - \frac{v_x}{a + b} (\alpha_f - \alpha_r + \delta) + \frac{a}{v_x I_z} (aF_f - bF_r + Y) \]
\[ \dot{\alpha}_r = \frac{F_f + F_r}{mv_x} - \frac{v_x}{a + b} (\alpha_f - \alpha_r + \delta) - \frac{b}{v_x I_z} (aF_f - bF_r + Y) \]
\[ r = \frac{v_x}{a + b} (\alpha_f - \alpha_r + \delta) \]

lateral velocity

\[ \dot{v}_y = \frac{F_f \cos \delta + F_r}{m} - r v_x \]

yaw rate derivative

\[ \dot{r} = \frac{a F_f \cos \delta - b F_r + Y}{I_z} \]
• **The overall dynamics model** is recast as a **PWA system** by introducing the Boolean variables

\[
\gamma_f = 0 \iff \alpha_f \leq \hat{p}_f \\
\gamma_r = 0 \iff \alpha_r \leq \hat{p}_r
\]

• By **discretizing** with sampling period \( T_s = 0.1 \text{ s} \) we obtain

\[
x(k + 1) = A_i x(k) + B_i u(k) + f_i \\
y(k) = C x(k) + D u(k)
\]

\( i \in \{1, \ldots, 4\} : \quad H_i x(k) \leq K_i \)

where \( x = [\alpha_f \, \alpha_r]' \)  

slip angles

\( u = [Y \, \delta]' \)  

yaw moment  
front steering

\( y = r \)  

yaw rate
Control goal:

stabilize the system at the equilibrium obtained with $\delta(k) = \hat{\delta}(k)$
while minimizing the use of the brake actuator $(\hat{Y}(k) = 0)$

Equilibrium condition in the linear region:

$$\begin{align*}
\dot{\hat{\alpha}}_f &= \frac{F_f + F_r}{m v_x} - \frac{v_x}{a+b} (\alpha_f - \alpha_r + \hat{\delta}) + \frac{a}{v_x I_z} (a F_f - b F_r + \hat{Y}) \\
\dot{\hat{\alpha}}_r &= \frac{F_f + F_r}{m v_x} - \frac{v_x}{a+b} (\alpha_f - \alpha_r + \hat{\delta}) - \frac{b}{v_x I_z} (a F_f - b F_r + \hat{Y})
\end{align*}$$

Time-varying set-points are defined using the overall dynamical model

$$\begin{align*}
\hat{\alpha}_f &= \frac{m \hat{v}_x^2 b c_r \hat{\delta}}{m \tilde{v}_x^2 (a c_f - b c_r) - c_f c_r (a + b)^2} \\
\hat{\alpha}_r &= \hat{\alpha}_f \frac{a c_f}{b c_r} \\
\hat{r} &= \frac{\tilde{v}_x}{a+b} (\hat{\alpha}_f - \hat{\alpha}_r + \hat{\delta})
\end{align*}$$

Current longitudinal velocity $v_x(k)$ is used to update set-points.
**Yaw rate tracking:**

zero tracking error in steady state is provided by integral action

\[ I_r(k + 1) = I_r(k) + r(k) - r_s(k) \]
\[ r_s(k + 1) = r_s(k) \]

**The global hybrid dynamical model** of the vehicle is given by

\[ z(k + 1) = \tilde{A}_i z(k) + \tilde{B}_i u(k) + \tilde{f}_i \]
\[ y(k) = \tilde{C} z(k) + \tilde{D} u(k) \]
\[ i \in \{1, \ldots, 4\} : \tilde{H}_i z(k) \leq \tilde{K}_i \]

\[ \tilde{A}_i = \begin{bmatrix} A_i & 0 & 0 & 0 \\ \frac{v_x}{a+b} & \frac{v_x}{a+b} & 1 & -1 \end{bmatrix}, \quad \tilde{B}_i = \begin{bmatrix} B_i \\ 0 \\ \frac{v_x}{a+b} \\ 0 \end{bmatrix}, \quad \tilde{f}_i = \begin{bmatrix} f_i \\ 0 \end{bmatrix}, \]
\[ \tilde{C} = \begin{bmatrix} \frac{v_x}{a+b} & -\frac{v_x}{a+b} & 0 & 0 \end{bmatrix}, \quad \tilde{D} = \begin{bmatrix} 0 & \frac{v_x}{a+b} \end{bmatrix}, \]
\[ \tilde{H}_i = \begin{bmatrix} H_i & 0 & 0 \end{bmatrix}, \quad \tilde{K}_i = K_i. \]
• The **optimal control problem** solved at every time step $k$ is

$$
\min_{u_k} \sum_{j=0}^{N-1} \left\{ (z_{k+j|k} - \hat{z})'Q_z(z_{k+j|k} - \hat{z}) + (y_{k+j|k} - \hat{y})'Q_y(y_{k+j|k} - \hat{y}) + (u_{k+j|k} - \hat{u})'Q_u(u_{k+j|k} - \hat{u}) \right\}
$$

s.t. $z_{k|k} = z(k)$
hybrid dynamics
state and input constraints

- **slip angles tracking & int. action**
- **yaw rate tracking**
- **penalty on actuators action**

**MIQP**

• State and input **constraints:**

$$
\begin{align*}
[z]_1(k) & \geq -\hat{p}_f & \text{front slip angle} \\
[z]_2(k) & \geq -\hat{p}_r & \text{rear slip angle} \\
[u]_1(k) & \leq 1000 \ \text{[Nm]} & \text{yaw moment} \\
[u]_2(k) & \leq 0.35 \ \text{[rad]} & \text{front steering}
\end{align*}
$$
• Simulations run on a **nonlinear vehicle model** including:

  - Longitudinal and lateral vehicle dynamics
  - Yaw rate dynamics
  - Steering actuation dynamics

• Driver’s steering command **constant** over the simulation interval

• **Controller setup** after calibration:
  
  - Prediction horizon \( N = 3 \)
  
  - Weight matrices \( Q_z = \begin{bmatrix} 0.1 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 \\ 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad Q_u = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad Q_y = 1 \)
• **Stability analysis** under nominal conditions (with $\hat{\delta} = 0$):

open-loop vs closed-loop

• Controller has to cope with **linearization errors**
• **Stability analysis** under nominal conditions (with $\hat{\delta} = 0$):

  - **open-loop**
  - **closed-loop**

  **stable** initial conditions
  **unstable** initial conditions

  **closed-loop provides a larger** stability region
• Robustness analysis w.r.t. model mismatches (with $\hat{\delta} = -0.05$):

  - nominal longitudinal velocity $\hat{v}_x = 20$ m/s
  - real longitudinal velocity $v_x = 15$ m/s and $v_x = 25$ m/s

• Stability and fast tracking response are provided
• **Turns on slippery road surface** (with $\hat{\delta} = -0.05$):

  ‣ several values tested: $\hat{s} = 0$, $s = 0.20$, $s = 0.30$, $s = 0.35$

• **Good degree of robustness** with respect to slip mismatches
• Computational issues:

  ‣ MPC-based approach is **viable for experimental tests**
    (average CPU time 17ms, worst-case CPU time 63 ms in MATLAB),
    but requires a **MIQP solver** in the ECU

• Explicit solution of hybrid MPC control problem:

  ‣ Exploits **multiparametric programming** techniques to provide a description of
    the control law as an **explicit** function of the state

  ‣ All the computation is executed **off-line**, only simple set-membership tests and
    function evaluations are performed on-line to compute the control action

  ‣ However, the explicit solution requires **memory**
    (around **5000 polytopes** to be stored, in this case!)

  ‣ An **approximation** of the solution is needed
    for real implementation
SWITCHED MPC SOLUTION

- Simpler solution: assume PWA mode remains constant in prediction
- Design a linear MPC controller for each mode, make it explicit
- 4 linear explicit MPC’s are enough (linear/saturation × front/rear)

Simulation results:
Experimental results (slalom test):
• Experimental results (stability recovery test):
EXPERIMENTAL RESULTS

- Experimental results (double lane change):

Vehicle center of mass (circle) and heading (line)

control is active

controller off
HYBRID MPC OF INDUCTION OF ESCHERICHIA COLI
• **Goal:** control the lactose regulation system of a colony of *E. coli*

![Diagram showing the lactose network](image)

**TMG** = *thio-methyl galactosidase* concentration

• Model, measurements, and actuation are at the **entire colony** level
**Bistable** lactose regulation system of E. coli

\[
\begin{align*}
\lambda_1(T_e) \\
\lambda_2(T_e)
\end{align*}
\]

Two-state Markov chain

- The probabilities \(x_{lo}, x_{hi}\) to be in low/high state satisfy the dynamics

\[
\frac{d}{dt} \begin{bmatrix} x_{lo} \\ x_{hi} \end{bmatrix} = \begin{bmatrix} -\lambda_1(T_e) & \lambda_2(T_e) \\ \lambda_1(T_e) & -\lambda_2(T_e) \end{bmatrix} \begin{bmatrix} x_{lo} \\ x_{hi} \end{bmatrix}
\]

- Transition rates \(\lambda_1, \lambda_2\) modeled as piecewise constant functions of \(T_e\)

\[
\begin{array}{c|c|c}
T_e [10^{-4} \text{mM}] & \lambda_1(T_e) [\text{min}^{-1}] & \lambda_2(T_e) [\text{min}^{-1}] \\
\hline
1.4, 1.5 & 8.68 \cdot 10^{-4} & 5.91 \cdot 10^{-5} \\
1.5, 1.6 & 9.27 \cdot 10^{-4} & 3.61 \cdot 10^{-3} \\
1.6, 1.7 & 1.13 \cdot 10^{-3} & 2.36 \cdot 10^{-3} \\
1.7, 1.8 & 1.39 \cdot 10^{-3} & 1.54 \cdot 10^{-3} \\
1.8, 1.9 & 1.67 \cdot 10^{-3} & 9.53 \cdot 10^{-4} \\
1.9, 2.0 & 1.93 \cdot 10^{-3} & 5.54 \cdot 10^{-4} \\
\end{array}
\]
• Hybrid MPC problem
  – switched linear system
  – constraints on input $T_e$ and $dT_e/dt$
  – penalties on tracking error $y - y_r$ and input rate $dT_e/dt$

• Closed-loop results
  – MPC controller developed with Hybrid Toolbox in MATLAB
  – Mixed-Integer Linear Program solver GLPK
  – solution time: 32 ms (worst case=280 ms) on 1.2 GHz laptop
  – sampling time = 10 min

©2019 A. Bemporad - "Model Predictive Control"