Model Predictive Control of Hybrid Systems

Alberto Bemporad

Dip. Ingegneria dell’Informazione
Università di Siena
bemporad@dii.unisi.it

Automatic Control Laboratory
Swiss Federal Institute of Technology (ETH)

http://www.dii.unisi.it/~bemporad

Hybrid Systems

\[ x(t) = f(x(t), u(t)) \]
\[ y(t) = g(x(t), u(t)) \]

\[ X = \{1, 2, 3, 4, 5\} \]
\[ U = \{A, B, C\} \]

Motivation: Embedded Systems

- Consumer electronics
- Home appliances
- Office automation
- Automobiles
- Industrial plants
- ...
Hybrid Models

Key Requirements for Hybrid Models

• **Descriptive** enough to capture the behavior of the system
  - continuous dynamics (physical laws)
  - logic components (switches, automata, software code)
  - interconnection between logic and dynamics

  ![TRADE OFF](image)

• **Simple** enough for solving **analysis** and **synthesis** problems

---

Mixed Integer Programming and Hybrid Systems

0. Given a Boolean statement \( F(X_1, X_2, \ldots, X_n) \)

1. Convert to Conjunctive Normal Form (CNF):

\[
\bigwedge_{j=1}^m \left( \bigvee_{i \in P_j} X_i \bigwedge_{i \in N_j} \bar{X_i} \right)
\]

2. Transform into inequalities:

\[
1 \leq \sum_{i \in P_j} \delta_i + \sum_{i \in N_j} (1 - \delta_i) \\
\vdots \\
1 \leq \sum_{i \in P_m} \delta_i + \sum_{i \in N_m} (1 - \delta_i)
\]

\[A\delta \leq B, \quad \delta \in \{0, 1\}\]

- Every logic proposition can be translated into linear integer inequalities

---

Transformation into Linear Integer Inequalities

1. Convert to Conjunctive Normal Form (CNF):

\[
\bigwedge_{j=1}^m \left( \bigvee_{i \in P_j} X_i \bigwedge_{i \in N_j} \bar{X_i} \right)
\]

2. Transform into inequalities:

\[
1 \leq \sum_{i \in P_j} \delta_i + \sum_{i \in N_j} (1 - \delta_i) \\
\vdots \\
1 \leq \sum_{i \in P_m} \delta_i + \sum_{i \in N_m} (1 - \delta_i)
\]

\[A\delta \leq B, \quad \delta \in \{0, 1\}\]

- Every logic proposition can be translated into linear integer inequalities
**Mixed Logical Dynamical Systems**

- **Mixed Integer Programming and Hybrid Systems**
  - X₁ \(\lor\) X₂ \(\land\) X₃ \(\land\) X₄ ≥ 1
  - X₁ \(\land\) X₂ \(\land\) X₃ \(\land\) X₄ ≥ 2
  - X₁ ⊆ X₂ \(\land\) X₃ ≤ X₄ ≤ 0
  - X₁ ⊆ X₂ \(\land\) X₃ ≤ X₄ ≤ 0

- **Mixed Logical Dynamical (MLD) form**
  - \[ x(t) = Ax(t) + B_3 u(t) + B_3 \delta(t) + B_3 z(t) \]
  - \[ y(t) = Cx(t) + D_1 u(t) + D_2 \delta(t) + D_3 z(t) \]

- **A Simple Example**
  - System:
    - \[ x(t+1) = \begin{cases} 
    0.8x(t) + u(t) & \text{if } x(t) \geq 0 \\
    -0.8x(t) + u(t) & \text{if } x(t) < 0 
    \end{cases} \]
    - \(-10 \leq x(t) \leq 10, -1 \leq u(t) \leq 1\)
  - Associate \([\delta(t) = 1] \leftrightarrow [x(t) \geq 0]\)
  - Then \[ x(t+1) = 1.6z(t) - 0.8x(t) + u(t) \]
  - Rewrite as linear equation
    - \[ x(t+1) = 1.6z(t) - 0.8x(t) + u(t) \]

- **HYSDEL**
  - (HYbrid Systems DEscription Language)
    - Describe hybrid systems:
      - Automata
      - Logic
      - Lin. Dynamics
      - Interfaces
      - Constraints
    - Automatically generate MLD models in Matlab
      - MLD model is not unique in terms of the number of auxiliary variables

- **http://control.ethz.ch/~hybrid/hysdel**

- **Temporal Model, Automata, 1999**

- **Temporal, Mignone, 2000**

- **Mixed Integer Programming and Hybrid Systems**
  - \( 0 \leq x_{\delta} \leq 6 \) \( \forall x \in X \) \( x_{\delta} \leq h + M(1 - \delta) \) \( x_{\delta} \geq h - M \delta \) \( M \geq |a_{\delta} - b| \) \( \forall x \in X \) \( R^n \)

- **Temporal, Mignone, 2000**

- **Temporal Model, Automata, 1999**

- **Temporal, Mignone, 2000**
System Theory for Hybrid Systems

- Analysis
  - Realization & Transformation
  - Well-posedness
  - Stability
  - Reachability (=Verification)
  - Observability
- Synthesis
  - Control
  - State estimation
  - Identification
  - Modeling language

Other Hybrid Models

- Piecewise affine (PWA) systems
  \[ C_i = \{ x : H^i x \leq K^i \}, \quad i = 1, \ldots, s \]
  Polyhedral partition of the state-space
  Affine dynamics in each region
  \[ x(t + 1) = A^i x(t) + B^i u(t) + f^i(t) \]
  Ex: linearization of NL system at different operating point

- Linear complementarity (LC) systems
  \[ x(t + 1) = A^i x(t) + B^1 u(t) + B^2 w(t) \]
  \[ y(t) = C^i x(t) + D^1 u(t) + D^2 w(t) \]
  \[ v(t) = E^i x(t) + E^2 u(t) + E^3 w(t) + e_i \]
  \[ 0 \leq v(t), w(t) \geq 0 \]
  Ex: mechanical systems
  Generalization: Extended Lin. Compl. (ELC) systems

Equivalence Results

Theoretical properties and analysis/synthesis tools can be transferred from one class to another

MLD and PWA Systems

- MLD: \[ x(t + 1) = A x(t) + B_1 u(t) + B_2 w(t) \]
  \[ y(t) = C x(t) + D_1 u(t) + D_2 w(t) \]
  \[ v(t) = E_1 x(t) + E_2 u(t) + E_3 w(t) + e_i \]
  By well-posedness hypothesis on \( z(t), \delta(t) \) linearity of MLD constraints
  \[ z = K_1 x + K_2 u + K_3 \quad \forall (x, u) : F(x, u) = \delta \]
  PWA form
  \[ x(t + 1) = A^i x(t) + B^i u(t) + f^i(t) \]
  \[ y(t) = C^i x(t) + D^i u(t) + g^i(t) \]
  \[ F^i x(t) + G^i u(t) \leq h^i \]
  Ex: mechanical systems

- Confirms (Sontag, 1996): PWL systems and hybrid systems are equivalent
System Theory for Hybrid Systems

- Analysis
  - Realization & Transformation
  - Well-posedness
  - Stability
  - Reachability (=Verification)
  - Observability

- Synthesis
  - Control
  - State estimation
  - Identification
  - Modeling language

Well-posedness

- MLD well-posedness:
  \[ \delta(t) = F(x(t), u(t)) \]
  \[ z(t) = G(x(t), u(t)) \]

Numerical test based on mixed-integer programming available (Bemporad, Morari, Automatica, 1999)

- LC well-posedness:
  \[ x(t + 1) = A x(t) + B_1 u(t) + B_2 w(t) \]
  \[ y(t) = C x(t) + D_1 u(t) + D_2 w(t) + e \]
  \[ 0 \leq b(x(t), u(t), w(t)) \]


Well-posedness

- Are state and output trajectories defined?

Generic hybrid model:

\[ x(k + 1) = f(x(k), u(k), w(k)) \quad (1a) \]
\[ y(k) = g(x(k), u(k), w(k)) \quad (1b) \]
\[ 0 \leq h(x(k), u(k), w(k)) \quad (1c) \]

are single valued

Definition 1 Let \( \Omega \subseteq \mathbb{R}^n \times \mathbb{R}^m \) be a set of input-state pairs. A hybrid system of the form (1) is called well-posed on \( \Omega \) if (1) is uniquely solvable in \( x(k + 1) \) and \( y(k) \) for all pairs \( (x(k), u(k)) \in \Omega \).

Definition 2 Let \( X(0) \subseteq \mathbb{R}^n \) be a set of initial conditions, and \( U \subseteq \mathbb{R}^m \) a set of inputs. A hybrid system of the form (1) is called persistently well-posed on \( (X(0), U) \) if for all \( k \geq 0 \) (1) is uniquely solvable in \( x(k + 1) \) and \( y(k) \), for all pairs \( (x(k), u(k)) \) such that \( x(0) \in X(0), u(k) \in U \).

System Theory for Hybrid Systems

- Analysis
  - Well posedness
  - Realization & Transformation
  - Stability
  - Reachability (=Verification)
  - Observability
  - Control
  - State estimation
  - Identification
  - Modeling language

- MLD, LC, PWA, MMPS

- MLD, PWA, MLD

- MLD, PWA

- MLD

- MLD

- MLD
Model Predictive Control of Hybrid Systems

MPC for Hybrid Systems

- At time $t$, solve with respect to $U = \{u(t), u(t + 1), \ldots, u(t + T - 1)\}$ the finite-horizon open-loop, optimal control problem:

$$
\begin{align*}
\min_{U} J(U, x(t), r) & \equiv \sum_{k=0}^{T-1} \left[ Q \|y(t+k+1|t) - r\| + R \|u(t+k) - u\| \right] \\
& \quad + \sigma \left[ \|z(t+k|t) - z\| + \|z(t+k|t) - u\| + \|z(t+k|t) - z\| \right]
\end{align*}
$$

- Subject to

$$
\begin{cases}
MLD model \\
x(t|t) = x(t) \\
x(t+T|t) = x
\end{cases}
$$

- Apply only $u(t) = u^*$(discard the remaining optimal inputs)
- Repeat the whole optimization at time $t+1$

Receding Horizon - Example

Chess
\section*{Closed-Loop Stability}

\textbf{Theorem 1} Let \((x_r, u_r)\) be the equilibrium pair for the set point \(r\). Assume that the optimization problem is feasible at time \(t = 0\). Then \(\forall Q, R > 0, \sigma > 0\), the predictive controller stabilizes the MLD system

\[
\lim_{t \to \infty} y(t) = r \quad \lim_{t \to \infty} u(t) = u_r
\]

\(\lim_{t \to \infty} x(t) = x_r, \lim_{t \to \infty} z(t) = z_r, \lim_{t \to \infty} \delta(t) = \delta_r\), and all the constraints are fulfilled.

(Bemporad, Morari, Automatica, 1999)

\section*{Stability Proof}

Linear case: \(J(U, t) = \sum_{k=0}^{\sigma} \|y(t+k|t)\|^2 + \rho \|u(t+k)\|^2\). \(\rho > 0\)

\(x(t+T) = 0\) terminal constraint

IDEA: Use the value function \(V(t) = \min_{U_{\text{init}}} J(U_{\text{init}}, t+1)\) a Lyapunov function.

At time \(t+1\) extend the previous sequence \(U^*_1(t+1), u^*_1(t+1) \ldots, u^*_1(t+T-1)\)

By construction, \(U_{\text{init}}\) is feasible at time \(t+1\), and

\[ V(t+1) = J(U^*_1(t+1), t+1) = J(U^*_1(t)) - \|y(t)\|^2 - \rho \|u(t)\|^2 \]

\[ \Rightarrow V(t) \text{ is nonnegative and decreasing} \Rightarrow \exists \lim_{t \to \infty} V(t) \]

\[ \Rightarrow \lim_{t \to \infty} V(t) - V(t+1) = 0 \Rightarrow \|y(t)\|^2 + \rho \|u(t)\|^2 \leq V(t) - V(t+1) \to 0 \]

\[ y(t), u(t) \to 0 \]

Note: Global optimum not needed!

\section*{Hybrid MPC - Example}

\textbf{Switching System}:

\[
\begin{cases}
U(t) = 0 & \text{if } \sin(x(t)) > 0 \\
U(t) = 1 & \text{if } \sin(x(t)) < 0
\end{cases}
\]

\textbf{Constraint}: \(-1 \leq u(t) \leq 1\)

\textbf{Open loop}:

\textbf{Closed loop}:

\section*{MIQP Formulation of MPC}

\[
\begin{align*}
\min \quad & \sum_{t=0}^{T} \left\{ y'(t+k+1|t)Qy(t+k+1|t) + u'(t+k)Ru(t+k) \right\} \\
\text{s.t.} \quad & \text{MLD dynamics}
\end{align*}
\]

\[
\begin{align*}
\min \quad & \frac{1}{T} \sum_{t=0}^{T} \left[ x'(t)Hx + x'(t)Fz + 1 \right] \\
\text{s.t.} \quad & Gz \leq W + Sz(t)
\end{align*}
\]

\textbf{Mixed Integer Quadratic Program (MIQP)}

\[
\begin{align*}
\begin{cases}
u \in \mathbb{R}^n \\
\delta \in \{0, 1\}^n \\
z \in \mathbb{R}^n
\end{cases}
\end{align*}
\]

\[
\begin{cases}
\xi \in \mathbb{R}^{(n+n) \times n} \\
z \in \{0, 1\}^n
\end{cases}
\]
Hybrid Control - An Example

(Hedlund and Rantzer, CDC1999)

- Objective:
  - Control the temperature $T_1$, $T_2$ to a given set-point

- Constraints:
  - Only three operation modes:
    1. Heat only the first furnace
    2. Heat only the second furnace
    3. Do not heat any furnaces

- Amount of heating power is constant

MILP Formulation of MPC

\[
\begin{align*}
\text{min} & \quad \sum_{k=0}^{T-1} [Q_y(t+k+1)]_\infty + [Ru(t+k)]_\infty \\
\text{s.t.} & \quad \epsilon \leq x \\
& \quad \epsilon \leq x
\end{align*}
\]

- Introduce slack variables:

\[
\begin{align*}
\epsilon & \leq [Q_y(t+k+1)]_\infty \\
\epsilon & \leq [Ru(t+k)]_\infty
\end{align*}
\]

- Constraints:

\[
\begin{align*}
\epsilon & \leq [Q_y(t+k)]_\infty, \quad i = 1, \ldots, p, \quad k = 1, \ldots, T-1 \\
\epsilon & \geq -[Q_y(t+k)]_\infty, \quad i = 1, \ldots, p, \quad k = 1, \ldots, T-1 \\
\epsilon & \geq [Ru(t+k)]_\infty, \quad i = 1, \ldots, m, \quad k = 0, \ldots, T-1 \\
\epsilon & \geq -[Ru(t+k)]_\infty, \quad i = 1, \ldots, m, \quad k = 0, \ldots, T-1
\end{align*}
\]

- Set

\[
\begin{align*}
\epsilon_i & \in \{0,1\}, \quad \epsilon_{k-1}, \epsilon_{k-1}, \ldots, \epsilon_{T-1}, U, \delta, z
\end{align*}
\]

- Objective:

\[
\begin{align*}
\min J(x(t)) &= \sum_{k=0}^{T-1} \epsilon_i^2 + \epsilon_{k-1}^2 \\
\text{s.t.} & \quad G \leq W + S x(t)
\end{align*}
\]

Alternate Heating of Two Furnaces

- MLD model:

- HYSDEL model:

- Constraints

\[
\begin{align*}
-1 & \leq x_1 \leq 1 \\
-1 & \leq x_2 \leq 1
\end{align*}
\]

Alternate Heating of Two Furnaces

Performance index

\[
\begin{align*}
\min & \quad \sum_{k=0}^{T-1} \|[R(u(t+k+1) - u(t+k))]|_\infty + ||Q(x(t+k+1) - x_0)||_\infty
\end{align*}
\]

Constraints

\[
\begin{align*}
-1 & \leq x_1 \leq 1 \\
-1 & \leq x_2 \leq 1
\end{align*}
\]
Closed-Loop Behavior

\[ u_0 = 0.4 \]

Set point cannot be reached

\[ u_0 = 0.8 \]

Set point is reached

- Computational complexity of on-line MILP

<table>
<thead>
<tr>
<th>Linear constraints</th>
<th>168</th>
</tr>
</thead>
<tbody>
<tr>
<td>Continuous variables</td>
<td>33</td>
</tr>
<tr>
<td>Binary variables</td>
<td>6</td>
</tr>
<tr>
<td>States</td>
<td>2</td>
</tr>
<tr>
<td>Time to solve MILP (av.)</td>
<td>1.09 s</td>
</tr>
</tbody>
</table>

Mixed-Integer Program Solvers

- Mixed-Integer Programming is NP-hard

Phase transitions have been found in computationally hard problems. (Monasson et al., Nature, 1999)

BUT

- General purpose Branch & Bound/Branch & Cut solvers available for MILP (CPLEX) and MIQP (Fletcher-Leyffer, BARON, XPRESS-MP)
- No need to reach global optimum (see proof of the theorem), although performance deteriorates
- Good for large sampling times (e.g., 1 h) / expensive hardware …
- … but not for fast sampling (e.g. 10 ms) / cheap hardware !

On-Line vs. Off-Line Optimization

\[
\min_{u} J(U, x(t)) = \sum_{k=0}^{T-1} |Q y(t+k+1) + R u(t+k)|
\]

subj. to

- MLD model
- \[ x(t+T) = 0 \]

- On-line optimization: given \( x(t) \), solve the problem at each time step \( t \)
- Off-line optimization: get the explicit solution of the MPC controller by solving the MILP for all \( x(t) \)

Explicit Model Predictive Control

(closed-form solution to constrained LQR)
• Linear Model:

\[ x(t+1) = Ax(t) + Bu(t) \]
\[ y(t) = Cx(t) + Du(t) \]

• Performance Index:

\[ \min \{ J(U, x(t)) = \frac{1}{2} U' H U + x'(t) F U + \frac{1}{2} x'(t) Y z(t) \} \]
subject to \( GU \leq W + Kx(t) \)

\[ \text{(quadratic)} \]
\[ \text{(linear)} \]

MULTIPARAMETRIC QUADRATIC PROGRAMMING

\[ \min \frac{1}{2} U' H U + x' F U + \frac{1}{2} x' G x \]
subject to \( GU \leq W + Kx \)

- Objective: solve the QP for all \( x(t) \)
- Coordinate transformation: \( z = U + H^{-1} F x \)

\[ \min \frac{1}{2} z' H z \]
subject to \( G z \leq W + S x \)
\( S = (K + G H^{-1} F)' \)

- Karush-Kuhn-Tucker (KKT) conditions for optimality:

\[ H z + G' \lambda = 0, \quad \lambda \in \mathbb{R}^q \]
\[ \lambda_i (G z - W^q - S z x) = 0, \quad \lambda_i \geq 0, \quad i = 1, \ldots, q \]

Constraint \( i \) active: \( G' z - W^q - S z x = 0, \quad \lambda_i \geq 0 \)

Constraint \( j \) inactive: \( G' z - W^q - S z x < 0, \quad \lambda_j = 0 \)
Linearity of Solution

\[ z_0 \in X \quad \rightarrow \quad \text{solve QP to find } (z_0, \lambda_0) \]
\[ \min_{z, \lambda} \frac{1}{2} z^T H z \quad \text{subject to } G z + \lambda = 0, \lambda \geq 0 \]

- identify active constraints
- form matrices \( \bar{G}, W, \bar{S} \) by collecting active constraints
- \( \bar{G} \mathbf{z} - W \mathbf{x} = 0, \mathbf{z} \in R^q \)

KKT conditions:
\[ H z + G^T \lambda = 0, \quad \lambda \in R^q \]
\[ \lambda_i (G^T z - W^T - S^T x) = 0, \quad \lambda_i \geq 0, \quad i = 1, \ldots, q \]

From (1):
\[ z = -H^{-1} G^T \lambda \]

From (2):
\[ \lambda(x) = - (GH^{-1}G^T)^{-1}(W + S x) \]
\[ z(x) = H^{-1} G^T (GH^{-1}G^T)^{-1}(W + S x) \]

In some neighborhood of \( x_0 \), \( \lambda \) and \( z \) are explicit linear functions of \( x \)

Determining a Critical Region

- Substitute \( \lambda(x) = - (GH^{-1}G^T)^{-1}(W + S x) \) and \( z(x) = H^{-1} G^T (GH^{-1}G^T)^{-1}(W + S x) \)
- Remove redundant constraints

\[ \text{critical region } CR_0 \]

\[ CR_i \in \{ A x \leq B \} \]

Theorem: \( \{ CR_0, CR_1, \ldots, CR_N \} \) is a partition of \( X \subseteq R^n \)

Multiparametric-QP

- Regions where the first component of the solution is the same can be joined (when their union is convex).

Union of Regions

\[ z(x) \Delta u_1(x), \ldots, u_{2, m-1}(x) \]
Complexity

- Worst-case complexity analysis:

  \[ M = \sum_{i=0}^{q} 2^i \]

  combinations of active constraints

  \[ N_r \leq \sum_{k=0}^{M-1} k!q^k \]

  upper-bound to the number of regions

- Numerical Tests:

<table>
<thead>
<tr>
<th>n</th>
<th>m=2</th>
<th>m=3</th>
<th>m=4</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.44 s (7)</td>
<td>0.49 s (7)</td>
<td>0.55 s (7)</td>
</tr>
<tr>
<td>3</td>
<td>1.15 s (13)</td>
<td>2.08 s (15)</td>
<td>1.75 s (15)</td>
</tr>
<tr>
<td>4</td>
<td>2.31 s (21)</td>
<td>5.87 s (29)</td>
<td>3.68 s (29)</td>
</tr>
</tbody>
</table>

  Off-line computation time to solve the mp-QP problem and, in parentheses, number of regions in the MPC controller (m=number of control moves, n=number of states)

Convexity and Continuity

**Theorem 4** Consider the multi-parametric quadratic program (21) and let \( H > 0 \). Then the optimizer \( z(x) \) is continuous and piecewise affine, and the optimal solution \( V(x) \) is continuous, convex and piecewise quadratic.

\[
\begin{align*}
  z(x) &= \arg \min \frac{1}{2} x^T H z
  \text{subject to } G z \leq W + S x \\
  V(x) &= \min \frac{1}{2} x^T H z
  \text{subject to } G z \leq W + S x
\end{align*}
\]

**Corollary:** The MPC controller is a continuous piecewise affine function of the state

MPC and LQR

**Constrained LQR** (infinite horizon)

\[
\begin{align*}
  \min_{F(t)} \int_0^\infty \left( x(t)^T Q(t) x(t) + u(t)^T R u(t) \right) dt \\
  \text{subject to } \begin{cases} 
  Gz(t) \leq W + S x(t) \\
  \forall z(t) \end{cases}
\end{align*}
\]

**MPC** (finite horizon)

\[
\begin{align*}
  \min_{F(t)} \int_0^T \left( x(t)^T Q(t) x(t) + u(t)^T R u(t) \right) dt &+ \sum_{t=0}^{T-1} \left( x(t)^T Q(t) x(t) + u(t)^T R u(t) \right) \\
  \text{subject to } \begin{cases} 
  Gz(t) \leq W + S x(t) \\
  x(0) = x_0 \end{cases}
\end{align*}
\]

\[ R > 0, Q \geq 0, \text{ and } P, K \text{ satisfy the Riccati equation} \]

\[ K = -(R + P K P)^{-1} P K \]

\[ P = (A + B K F)^T (A + B K F) + K^T K + Q \]

Closed-Form Constrained-LQR

For every polytopic region the constrained LQR control law can be computed in closed form

\[ (Bemporad, Morari, Dua, Pistikopoulos, Automatica, in press) \]

Double Integrator Example

- System:
  \[ y(t) = \frac{1}{t} u(t) \]
  \[ x(t+1) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u(t) \]

- Constraints: \(-1 \leq u(t) \leq 1\)

- Control objective: minimize
  \[ \sum_{t=0}^{\infty} y(t) \| y(t) \|^2 + \sum_{k=0}^{N_u} u(t) \| u(t) \|^2 \]
  \[ u_{t+k} = K_{2Q} \| x(t+k) \|^2 \forall k \geq N_u \]

- Optimization problem: for \(N_u = 2\)

\[
H = \begin{bmatrix} 11.0932 & 5.4117 \\ 4.3806 & 4.4117 \end{bmatrix} \\
F = \begin{bmatrix} 4.6815 & 2.8810 \\ 10.0932 & 5.4117 \end{bmatrix} \\
G = \begin{bmatrix} -1 & 0 \\ 0 & 1 \\ 0 & -1 \end{bmatrix} \\
W = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \\
K = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}
\]

mp-QP solution

- Task:
  - Tune an MPC controller by simulation, using the MPC Simulink Toolbox.
  - Get the explicit solution of the MPC controller.
  - Validate the controller on experiments.
**MPC Tuning**

Two independent MPC controllers: \(x\)-axis, \(y\)-axis

(MPC Toolbox v2.08 - Bemporad, Morari, Ricker, 2001)

**Explicit MPC Solution**

Controller: \(x\): 22 Regions, \(y\): 23 Regions

\(x\)-MPC: sections at \(\alpha_x=0, \alpha_y=0, u_x=0, r_x=18, r_y=0\)

Region 1: LQR Controller (near Equilibrium)
Region 6: Saturation at \(-10\)
Region 16: Saturation at \(+10\)

**MPC Regulation of a Ball on a Plate**

**Design Steps:**

- Tune an MPC controller by simulation, using the *MPC Simulink Toolbox*.
- Get the explicit solution of the MPC controller.
- Validate the controller on experiments.

15th June

**Explicit MPC for Hybrid Systems**
On-Line vs. Off-Line Optimization

- **On-line optimization**: given \( x(t) \), solve the problem at each time step \( t \), by solving the Mixed-Integer Linear Program (MILP)
- **Off-line optimization**: get the explicit solution of the MPC controller by solving the MILP for all \( x(t) \)

\[
\begin{align*}
\min_{U} & \quad J(U) = \sum_{k=0}^{T-1} \left[ Q y(t + k + 1) + \| R u(t + k) \| \right] + \| K_n u(t + k) \| \\
\text{subj. to} & \quad \text{MLD model} \\
& \quad x(t+T) = \Phi x(t) + \Gamma u(t) \\
& \quad x(t) = c(t)
\end{align*}
\]

- Good for large sampling times (e.g., 1 h) / expensive hardware ...
- but not for fast sampling (e.g. 10 ms) / cheap hardware !

- Off-line optimization: get the explicit solution of the MPC controller by solving the MILP for all \( x(t) \)

\[
\begin{align*}
\min_{\xi} & \quad J(\xi) = f(\xi) \\
\text{s.t.} & \quad G \xi \leq W + F x(t)
\end{align*}
\]

- On-line optimization: given \( x(t) \), solve the problem at each time step \( t \)

Multiparametric MILP

- mp-MILP can be solved (by alternating MILPs and mp-LPs)
- **Theorem**: The multiparametric solution is piecewise affine
- **Corollary**: The MPC controller is piecewise affine in \( x \)

Multiparametric Solution

- Remarks on explicit MPC law:
  - Automatic partitioning of state-space (no gridding!)
  - Stability guarantee (value function=PWL Lyapunov function)
**Alternate Heating of Two Furnaces**

(Bemporad, Borrelli, Morari, 2000)

- mp-MILP optimization problem

\[
\min_{u(t), u(t+1), u(t+2)} \sum_{k=0}^{\infty} \| R(u(t+k+1) - u(t+k)) \|_\infty + \| Q(x(t+k+1) - x(t)) \|_\infty
\]

- Parameter set

\[
-1 \leq x_i \leq 1 \quad \text{parameterized!}
\]

- Computational complexity of mp-MILP

<table>
<thead>
<tr>
<th>Linear constraints</th>
<th>Continuous variables</th>
<th>Binary variables</th>
<th>Parameters</th>
<th>Time to solve mp-MILP</th>
<th>Number of regions</th>
</tr>
</thead>
<tbody>
<tr>
<td>168</td>
<td>105</td>
<td>168</td>
<td>33</td>
<td>6</td>
<td>105</td>
</tr>
</tbody>
</table>

**Hybrid Control Problem**

Renault Clio 1.9 DTI RXE
(Bemporad, Torrisi, 2001)

**GOAL:**

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

**mp-MILP Solution**

\[ u_0 = 0.4 \]

\[ u_0 = 0.8 \]

Set point cannot be reached

Set point is reached

NOTE: The control action of explicit and of implicit MPC are totally equal!
### Hybrid 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_e(i)}{k_e} \quad \text{\( \omega \) = engine speed} \]
  \[ F_e = \frac{R_e(i)}{k_e} M \quad \text{\( M \) = engine torque} \]
  \[ i = \text{\( i \) = gear} \]

### Gear selection

- **Geared selection (traction force):**

  \[ F_e = \frac{R_e(i)}{k_e} M \]
  
  - Depends on gear \#i
  
  - Define auxiliary continuous variables:

    \[ \text{IF} \ g_i = 1 \ \text{THEN} \ F_{e1} = \frac{R_e(i)}{k_e} M \ \text{ELSE} 0 \]

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

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

  \[ \omega = \frac{R_e(i)}{k_e} \]
  
  - Similarly, also requires 6 auxiliary continuous variables

### Hybrid Model

- **Engine torque**

  \[ C_e^- (\omega) \leq M \leq C_e^+ (\omega) \]

- **Max engine torque**

  \[ C_e^+ (\omega) \]

  - Piecewise-linearization: \( \text{(PW Toolbox, Julian, 1999)} \)

  - Requires: 4 binary aux variables

- **Min engine torque**

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

### Hysdel Model (Hybrid Systems Description Language)

- \[ 1 \]
  - \( \text{1} \text{ model, engine, gear, road, \ldots} \)

- \[ \text{2} \]
  - Define \( \text{1} \text{ auxiliary continuous variables, \ldots} \)

- \[ \text{3} \]
  - \( \text{3} \text{ sampled, \ldots} \)

- \[ \text{4} \]
  - \( \text{4} \text{ sampling, \ldots} \)

- \[ \text{5} \]
  - \( \text{5} \text{ output, \ldots} \)

- \[ \text{http://control.ethz.ch/~hybrid/hysdel} \]
  - \( \text{(Tesi, Bemporad, Mignone, 2000)} \)
Hybrid Model

- MLD model (generated by HYSDEL in Matlab)

\[
\begin{align*}
x(t+1) &= Ax(t) + B_1 u(t) + B_2 \delta(t) + B_3 z(t) \\
y(t) &= Cx(t) + D_1 u(t) + D_2 \delta(t) + D_3 z(t) \\
E_2 \delta(t) + E_3 z(t) &\leq E_4 x(t) + E_5 u(t) + E_6 
\end{align*}
\]

- 2 continuous states: \( x, v \) (vehicle position and speed)
- 2 continuous inputs: \( M, F_b \) (engine torque, brake force)
- 6 binary inputs: \( g_1, g_2, g_3, g_4, g_5 \) (gears)
- 1 continuous output: \( v \) (vehicle speed)
- 16 auxiliary continuous vars: (6 traction force, 6 engine speed, 4 PWL max engine torque)
- 4 auxiliary binary vars: (PWL max engine torque breakpoints)
- 96 mixed-integer inequalities

Hybrid Controller

- Max-speed controller

\[
\begin{align*}
\max J(u(t), x(t)) &= x(t+1|t) \\
\text{subject to} & \quad \text{MLD model} \\
x(t|t) &= x(t) 
\end{align*}
\]

MILP optimization problem
- Linear constraints
- Continuous variables: 96
- Binary variables: 18
- Parameters: 45
- Time to solve mp-MILP: 45 s
- Number of regions: 11
- (\( x(t) \) is irrelevant)

- Tracking controller

\[
\begin{align*}
\min_u J(u(t), x(t)) &= \|v(t+1|t) - v_d(t)| + \rho |\omega| \\
\text{subject to} & \quad \text{MLD model} \\
x(t|t) &= x(t) 
\end{align*}
\]

MILP optimization problem
- Linear constraints: 98
- Continuous variables: 19
- Binary variables: 10
- Parameters: 2
- Time to solve mp-MILP: 27 m
- Number of regions: 49
Hybrid Controller

• Tracking controller

\[
\min \int_{t}^{t+1} |v(t+1) - v_d(t)| + \rho |\omega|
\]

Hybrid Controller

• Smoother tracking controller

\[
\min \int_{t}^{t+1} |v(t+1) - v_d(t)| + \rho |\omega|
\]
\[\text{subj. to}\]
\[|v(t+1) - v_d(t)| < T_s \omega_{\text{max}}\]
\[x(t+1) = x(t)\]

MILP optimization problem

Linear constraints
Binary variables
Parameters
Time to solve mp-MILP (Sun Ultra 10)
Number of regions

Comments on Hybrid MPC

• Hybrid systems as a framework for new applications, where both logic and continuous dynamics are relevant
• Mixed Logical Dynamical (MLD) systems as discrete-time, computation-oriented models for hybrid systems

\[
x(t+1) = Ax(t) + Bu(t) + B_2 \delta(t) + B_3 z(t)
\]
\[y(t) = Cx(t) + Du(t) + D_2 \delta(t) + D_3 z(t)\]
\[E_2 \delta(t) + E_3 z(t) \leq E_4 x(t) + E_1 u(t) + E_5\]
Comments on Hybrid MPC

• Hybrid systems as a framework for new applications, where both logic and continuous dynamics are relevant
• Mixed Logical Dynamical (MLD) systems as discrete-time, computation-oriented models for hybrid systems
• Supervisory MPC controllers can be synthesized via on-line mixed-integer programming (MILP/MIQP)
• Piecewise Linear Optimal Controllers can be synthesized via off-line multiparametric programming for fast-sampling applications

\[ u(x) = \begin{cases} F_1x + G_1 & \text{if } H_1x \leq K_1 \\ \vdots & \vdots \\ F_Nx + G_N & \text{if } H_Nx \leq K_N \end{cases} \]

Applications: automotive control, process industry (monitoring and set-point optimization), manufacturing (scheduling), ...

References on Hybrid Systems


Download: http://control.ethz.ch/~bemporad

The End