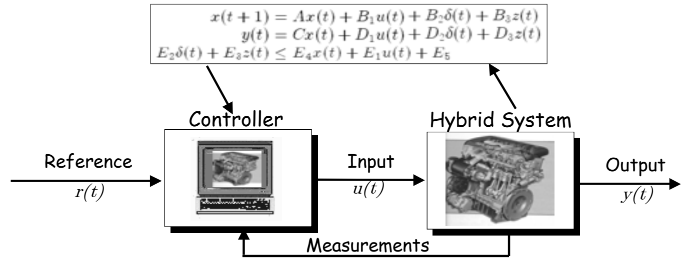
Controller Synthesis

Model Predictive Control of Hybrid Systems



- MODEL: a model of the plant is needed to predict the future behavior of the plant
- PREDICTIVE: optimization is based on the predicted future evolution of the plant
- CONTROL: control complex constrained multivariable plants

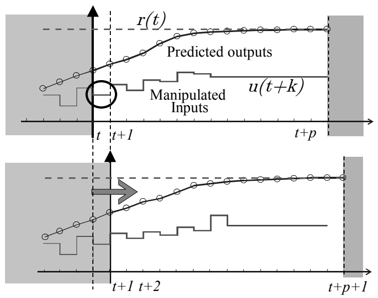
Receding Horizon Control

- At time t :

  Solve an optimal control problem over a finite future horizon p :

  – minimize $|y - r| + \rho|u|$

  – subject to constraints
  $u_{min} \leq u \leq u_{max}$
  $y_{min} \leq y \leq y_{max}$

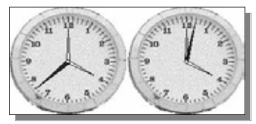


- Only apply the first optimal move $u^*(t)$
- Get new measurements, and repeat the optimization at time $t+1$

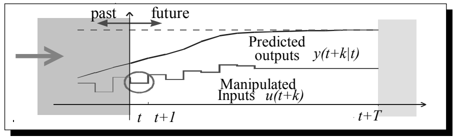
Advantage of on-line optimization: ***FEEDBACK!***

Receding Horizon - Example

Chess



MPC for Hybrid Systems



Model Predictive Control (MPC)

- At time t solve the finite-horizon open-loop, optimal control problem:

$$\min_{\xi} J(\xi, x(t), r) \triangleq \sum_{k=0}^{T-1} \|Q(y(t+k+1|t)-r)\| + \|R(u(t+k)-u_r)\|$$
$$+\sigma\left(\|\delta(t+k|t)-\delta_r\| + \|z(t+k|t)-z_r\| + \|x(t+k|t)-x_r\|\right)$$

$$\text{subj. to} \begin{cases} \text{MLD model} \\ x(t|t)=x(t) \\ x(t+T|t)=x_r \end{cases}$$

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

Closed-Loop Stability

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 $\forall Q, R \succ 0$, $\sigma > 0$, the predictive controller stabilizes the MLD system*

$$\lim_{t\to\infty} y(t) = r \qquad \lim_{t\to\infty} u(t) = u_r$$

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

(Bemporad, Morari, *Automatica,* 1999)

Stability Proof

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

$x(t+T|t)=0$ terminal constraint

IDEA: Use the value function $V(t) = J(U_t^*, t)$ as a Lyapunov function.

At time $t+1$, extend the previous sequence $U_t^* \triangleq \{u_t^*(t), u_t^*(t+1)\ldots, u_t^*(t+T-1)\}$

$$U_{\text{shift}} \triangleq \{u_t^*(t+1), \ldots, u_t^*(t+T-1), 0\}$$

By construction, U_{shift} is feasible at time $t+1$, and

$$V(t+1) = J(U_{t+1}^*, t+1) \le J(U_{\text{shift}}, t+1) =$$
$$= J(U_t^*, t) - \|y(t)\|^2 - \rho\|u(t)\|^2 =$$
$$= V(t) - \|y(t)\|^2 - \rho\|u(t)\|^2$$

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

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

$\Rightarrow y(t), u(t) \to 0$ ■

Note: Global optimum not needed !

MIQP Formulation of MPC

(Bemporad, Morari, 1999)

$$\min \sum_{k=0}^{T-1} \left\{ y'(t+k+1|t) Q y(t+k+1|t) + u'(t+k) R u(t+k) \right\}$$
$$\text{s.t. MLD dynamics}$$

- Set $\xi \triangleq [u(t), \ldots, u(t+T-1), \delta(t|t), \ldots, \delta(t+T-1|t), z(t|t), \ldots, z(t+T-1|t)]$

⇓

$$\min_{\xi} J(\xi, x(t)) = \frac{1}{2}\xi' H \xi + x'(t) F \xi + \frac{1}{2} x'(t) Y x(t)$$
$$\text{s.t.}\ G\xi \le W + S x(t)$$

Mixed Integer Quadratic Program (MIQP)

$$u \in \mathbf{R}^{n_u}, \quad \delta \in \{0,1\}^{n_\delta}, \quad z \in \mathbf{R}^{n_z} \quad \Longrightarrow \quad \xi \in \mathbf{R}^{T(n_u+n_z)} \times \{0,1\}^{T n_\delta}$$

MILP Formulation of MPC

(Bemporad, Borrelli, Morari, 2000)

$$\min \sum_{k=0}^{T-1} \|Qy(t+k+1|t)\|_\infty + \|Ru(t+k)\|_\infty$$
$$\text{s.t.MLD dynamics}$$

- Introduce slack variables: $\min |x|$ ⇒ $\min\ \epsilon$ s.t. $\epsilon \geq x$, $\epsilon \geq -x$

$$\begin{aligned} \epsilon_k^x &\geq \|Qy(t+k|t)\|_\infty \\ \epsilon_k^u &\geq \|Ru(t+k)\|_\infty \end{aligned} \Rightarrow \begin{aligned} \epsilon_k^x &\geq [Qy(t+k|t)]_i & i=1,\ldots,p,\ k=1,\ldots,T-1 \\ \epsilon_k^x &\geq -[Qy(t+k|t)]_i & i=1,\ldots,p,\ k=1,\ldots,T-1 \\ \epsilon_k^u &\geq [Ru(t+k))]_i & i=1,\ldots,m,\ k=0,\ldots,T-1 \\ \epsilon_k^u &\geq -[Ru(t+k))]_i & i=1,\ldots,m,\ k=0,\ldots,T-1 \end{aligned}$$

- Set $\xi \triangleq [\epsilon_1^x, \ldots, \epsilon_{N_y}^x, \epsilon_1^u, \ldots, \epsilon_{T-1}, U, \delta, z]$

Mixed Integer Linear Program (MILP)

$$\min_\xi J(\xi, x(t)) = \sum_{k=0}^{T-1} \epsilon_i^x + \epsilon_i^u$$
$$\text{s.t.} G\xi \leq W + Sx(t)$$

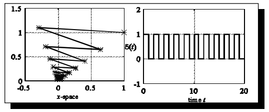
Hybrid MPC - Example

Switching System:

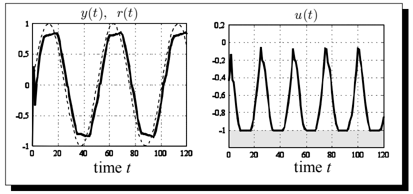
$$x(t+1) = 0.8 \begin{bmatrix} \cos\alpha(t) & -\sin\alpha(t) \\ \sin\alpha(t) & \cos\alpha(t) \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$
$$y(t) = [0\ \ 1]\, x(t)$$
$$\alpha(t) = \begin{cases} \frac{\pi}{3} & \text{if } [1\ \ 0]x(t) \geq 0 \\ -\frac{\pi}{3} & \text{if } [1\ \ 0]x(t) < 0 \end{cases}$$

Constraint: $-1 \leq u(t) \leq 1$

Open loop:



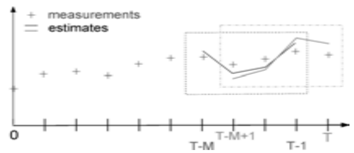
Closed loop:



State Estimation / Fault Detection

State Estimation / Fault Detection

- Problem: given past output measurements and inputs, estimate the current state/faults
- Solution: Use Moving Horizon Estimation for MLD systems (dual of MPC)



Augment the MLD model with:

- Input disturbances $\xi \in \mathbf{R}^n$
- Output disturbances $\zeta \in \mathbf{R}^p$

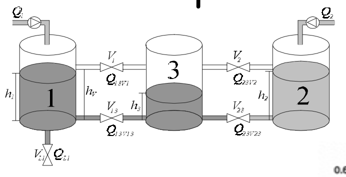
At each time t solve the optimization: $\min \sum_{k=1}^{T} \|\hat{y}(t-k|t) - y(t-k)\|^2 + \ldots$ and get estimates $\hat{x}(t)$

⇒ MHE optimization = MIQP (Bemporad, Mignone, Morari, ACC 1999)

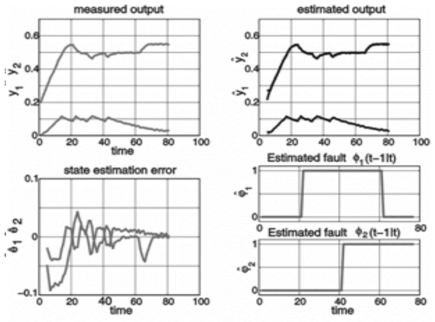
⇒ Convergence can be guaranteed (Ferrari-T., Mignone, Morari, IEEE TAC, in press)

Fault detection: augment MLD with unknown **binary** distubances $\phi \in \{0,1\}^p$

Example: Three Tank System



COSY Benchmark problem, ESF

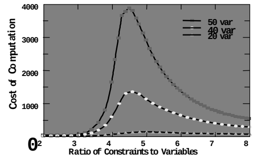


- ϕ_1 : leak in tank 1 for $20s \leq t \leq 60s$
- ϕ_2 : valve V_1 blocked for $t \geq 40s$
- Add logic constraint $[h_1 \leq h_v] \Rightarrow \phi_2 = 0$

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, Sahinidis, XPRESS-MP)

Free Matlab MILP/MIQP solver (Bemporad, Mignone, 1999)

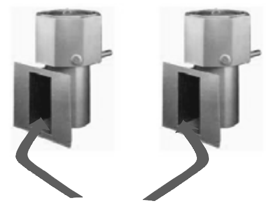
More solvers and benchmarks: **http://plato.la.asu.edu/bench.html**

- No need to reach global optimum (see proof of the theorem), although performance deteriorates

Hybrid Control Example: Heat Exchange

Hybrid Control - An Example

(Hedlund and Rantzer, CDC1999)



u_0

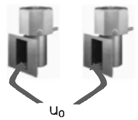
$$\begin{bmatrix}\dot{T}_1\\ \dot{T}_2\end{bmatrix} = \begin{bmatrix}-1 & 0\\ 0 & -2\end{bmatrix}\begin{bmatrix}T_1\\ T_2\end{bmatrix} + B_i u_0$$

$$B_i = \begin{cases}\begin{bmatrix}1\\0\end{bmatrix} & \text{if first furnace heated}\\ \begin{bmatrix}0\\1\end{bmatrix} & \text{if second furnace heated}\\ \begin{bmatrix}0\\0\end{bmatrix} & \text{if no heating}\end{cases}$$

- 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 u_0 is constant

Alternate Heating of Two Furnaces



- HYSDEL model:

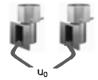
```
/* HEAT EXCHANGE model  - (C) 2001 by A. Bemporad, Zurich, March 13, 2001 */

SYSTEM furnaces {

INTERFACE {
        STATE {
                REAL t1,t2,u0; }
        INPUT {
                BOOL heat1,heat2; }
        PARAMETER {
                REAL Ts=0.08; /* sampling time, seconds */
                REAL Bd1=0.07688365361336; /* discretization of B matrix, heat 1 active */
                REAL Bd2=0.07392010551609; /* discretization of B matrix, heat 2 active */
                REAL A11=.9231; /* discretization of A matrix */
                REAL A22=.8521; /* discretization of A matrix */

                REAL u0max=10; /* upperbound on u0 */
                REAL e = 1e-6; /* precision for strict inequalities */  }
}

IMPLEMENTATION {
                AUX {REAL z1,z2; }

                DA {    z1 = {IF heat1 THEN Bd1*u0 [Bd1*u0max,0,e]};
                        z2 = {IF heat2 THEN Bd2*u0 [Bd2*u0max,0,e]}; }

                CONTINUOUS { t1 = A11*t1+z1;
                             t2 = A22*t2+z2;
                             u0 = u0; }

                MUST    {  ~heat1 | ~heat2;  /* heat1 and heat2 cannot be both active */ }
}
}
```

- MLD model:

$$x(t+1) = \begin{bmatrix} 0.9231 & 0 & 0 \\ 0 & 0.8521 & 0 \\ 0 & 0 & 1.0000 \end{bmatrix} x(t) + \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} z(t)$$

$$\begin{bmatrix} -1 & 0 \\ 1 & 0 \\ -1 & 0 \\ 1 & 0 \\ 0 & -1 \\ 0 & 1 \\ 0 & -1 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} z(t) \cdot \begin{bmatrix} -0.7688 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0.7688 & 0 \\ 0 & -0.7393 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0.7393 \\ -1.0000 & -1.0000 \end{bmatrix} u(t) + \begin{bmatrix} -0.0769 \\ 0.0769 \\ 0 \\ 0 \\ -0.0739 \\ 0.0739 \\ 0 \\ 0 \\ 0 \end{bmatrix} x_3(t) + \begin{bmatrix} 0.7688 \\ 0 \\ 0 \\ 0 \\ 0.7393 \\ 0 \\ 0 \\ 0 \\ 1.0000 \end{bmatrix}$$

Alternate Heating of Two Furnaces



- Performance index

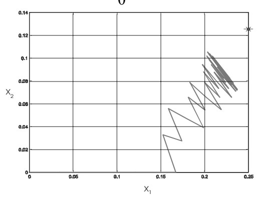
$$\min_{\{u(t),u(t+1),u(t+2)\}} \sum_{k=0}^{2} \|R(u(t+k+1)-u(t+k))\|_\infty + \|Q(x(t+k|t)-x_e)\|_\infty$$

- Constraints

$$\begin{array}{rcl} -1 & \leq x_1 \leq & 1 \\ -1 & \leq x_2 \leq & 1 \end{array}$$

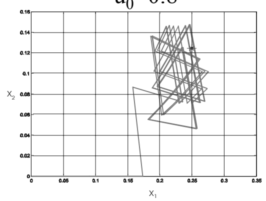
Closed-Loop Behavior

u_0=0.4



Set point cannot be reached

u_0=0.8



Set point is reached

- Computational complexity of on-line MILP

(Bemporad, Borrelli, Morari, 2000)

Linear constraints	168
Continuous variables	33
Binary variables	6
Parameters	3
Time to solve MILP (av.)	1,09 s

On-Line vs. Off-Line Optimization

$$\min_U J(U, x(t)) \triangleq \sum_{k=0}^{T-1} \|Qy(t+k+1|t)\|_\infty + \|Ru(t+k)\|_\infty$$

$$\text{subj. to} \begin{cases} \text{MLD model} \\ x(t|t) = x(t) \\ x(t+T|t) = 0 \end{cases}$$

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

Mixed-Integer Linear Program (MILP)

- 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)*

$$\min_\xi J(\xi, x(t)) \triangleq f'\xi$$
$$\text{s.t.} G\xi \leq W + Fx(t)$$

multi-parametric Mixed Integer Linear Program (mp-MILP)