Master in Control Engineering

Process Automation
2020-2021

Dipartimento di Ingegneria informatica
automatica e gestionale Antonio Ruberti

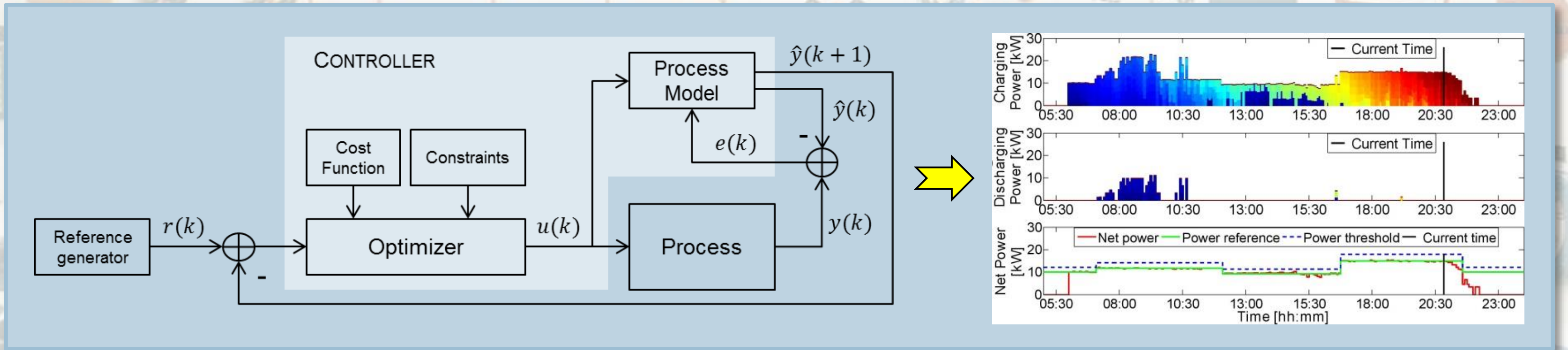
Sapienza
Università di Roma

Master in Control Engineering

Process Automation

12. Commercial MPC schemes

Slides based on:
E.F. Camacho, C. Bordons Alba, “Model Predictive Control”, *Advanced Textbooks in Control and Signal Processing*, Springer,-Verlag, XXII, 2nd ed., 2007, 405 p., ISBN 978-0-85729-398-5.



Outline

- Commercial MPC schemes
  - Dynamic Matrix Control (DMC)
  - Model Algorithmic Control (MAC)
  - Predictive Functional Control (PFC)
  - Summary

Dynamic Matrix Control (DMC)

- End of 70s
  - Cutler, Ramakar (Shell Oil company)
- Prediction Models
  - Step-response process model
    - Applicable to stable processes without intergrators
  - Constant disturbance model along the horizon

$$\hat{n}(t+k|t) = \hat{n}(t|t), k = 1, \dots, p, \quad (1)$$

with

$$\hat{n}(t|t) = y_m(t) - y(t) \quad (2)$$

where $y(t)$ is the output of the process model at time t:

$$y(t) = \sum_{i=1}^{\infty} g_i \Delta u(t-i) \quad (3)$$

Dynamic Matrix Control (DMC)

- Predicted output (p=prediction horizon)

$$\hat{y}(t+k|t) = \sum_{i=1}^{\infty} g_i \Delta u(t+k-i|t) + \hat{n}(t+k|t), k = 1, \dots, p \quad (4)$$

(4)+(1)

$$\hat{y}(t+k|t) = \sum_{i=1}^{k} g_i \Delta u(t+k-i|t) + \sum_{i=k+1}^{\infty} g_i \Delta u(t+k-i) + \hat{n}(t|t), k = 1, \dots, p \quad (5)$$

(5)+(2)

$$\hat{y}(t+k|t) = \sum_{i=1}^{k} g_i \Delta u(t+k-i|t) + \sum_{i=k+1}^{\infty} g_i \Delta u(t+k-i) + y_m(t) - y(t), k = 1, \dots, p \quad (6)$$

(6)+(3)

$$\hat{y}(t+k|t) = \sum_{i=1}^{k} g_i \Delta u(t+k-i|t) + \sum_{i=k+1}^{\infty} g_i \Delta u(t+k-i) + y_m(t) - \sum_{i=1}^{\infty} g_i \Delta u(t-i), k = 1, \dots, p \quad (7)$$

$$\hat{y}(t+k|t) = \sum_{i=1}^{k} g_i \Delta u(t+k-i|t) + y_m(t) + \sum_{i=1}^{\infty} (g_{k+i} - g_i) \Delta u(t-i), k = 1, \dots, p \quad (8)$$

- Free and forced response

$$\hat{y}(t+k|t) = \hat{y}_c(t+k|t) + f(t+k), k = 1, \dots, p \quad (10)$$

- Free response

$$f(t+k) = y_m(t) + \sum_{i=1}^{\infty} (g_{k+i} - g_i) \Delta u(t-i), k = 1, \dots, p \quad (11)$$

- Forced response

$$\hat{y}_c(t+k|t) = \sum_{i=1}^{k} g_i \Delta u(t+k-i|t), k = 1, \dots, p \quad (12)$$

Dynamic Matrix Control (DMC)

- Predicted output
  - The process is asymptotically stable and has no integrators $\Rightarrow g_{k+i} \cong g_i, i > N$

$$\hat{y}(t+k|t) \cong \sum_{i=1}^{k} g_i \Delta u(t+k-i|t) + f(t+k), k = 1, \dots, p \quad (12)$$

with

$$f(t+k) = y_m(t) + \sum_{i=1}^{N} (g_{k+i} - g_i) \Delta u(t-i) \quad (13)$$

– If the control horizon is smaller than the prediction horizon, for $k = m, m+1, \dots, p$:

$$\hat{y}(t+k|t) \cong \sum_{i=1}^{k} g_i \Delta u(t+k-i|t) + y_m(t) + \sum_{i=1}^{N} (g_{k+i} - g_i) \Delta u(t-i) =$$

$$= \sum_{i=k-m+1}^{k} g_i \Delta u(t+k-i|t) + \cancel{\sum_{i=1}^{k-m} g_i \Delta u(t+k-i|t)} + y_m(t) + \sum_{i=1}^{N} (g_{k+i} - g_i) \Delta u(t-i)$$

$$\boxed{\Delta u(t+k|t) = 0, \forall k = m, \dots, p-1}$$

$$\hat{y}(t+k|t) \cong \sum_{i=k-m+1}^{k} g_i \Delta u(t+k-i|t) + f(t+k)$$

Dynamic Matrix Control (DMC)

- Matrix formulation
  - Let p and m be the prediction and control horizons, respectively, with $m \leq p$
  - (12):

$$\hat{y}(t+1|t) = g_1 \Delta u(t|t) + f(t+1)$$

$$\hat{y}(t+2|t) = g_2 \Delta u(t|t) + g_1 \Delta u(t+1|t) + f(t+2)$$

$$\hat{y}(t+3|t) = g_3 \Delta u(t|t) + g_2 \Delta u(t+1|t) + g_1 \Delta u(t+2|t) + f(t+3)$$

$$\dots$$

$$\begin{aligned}\hat{y}(t+p|t) &= \textstyle\sum_{i=p}^{p-m+1} g_i \Delta u(t+p-i|t) + f(t+p) = \\ &= \underbrace{g_p \Delta u(t|t) + g_{p-1} \Delta u(t+1|t) + \cdots + g_{p-m+1} \Delta u(t+m-1|t)}_{m \text{ terms (future control actions)}} + f(t+p)\end{aligned}$$

- Define
  - The dynamic matrix

$$\boldsymbol{G} := \begin{pmatrix} g_1 & 0 & 0 & \dots & 0 \\ g_2 & g_1 & 0 & \dots & 0 \\ g_3 & g_2 & g_1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ g_m & g_{m-1} & g_{m-2} & \cdots & g_1 \\ g_{m+1} & g_m & g_{m-1} & \cdots & g_2 \\ \dots & \dots & \dots & \dots & \dots \\ g_p & g_{p-1} & g_{p-2} & \cdots & g_{p-m+1} \end{pmatrix} \in \mathbb{R}^{p \times m}$$

Dynamic Matrix Control (DMC)

- Matrix formulation
  - Define
    - the vector of output predictions along the prediction horizon

$$\hat{\boldsymbol{y}} \coloneqq \left(\hat{y}(t+k|t)\right)_{k=1,\dots,p} \in \mathbb{R}^{p\times 1}$$

    - the vector of control increments along the control horizon

$$\boldsymbol{u} \coloneqq \left(\Delta u(t+k|t)\right)_{k=0,\dots,m-1} \in \mathbb{R}^{m\times 1}$$

    - the vector of free responses (known constants)

$$\boldsymbol{f} \coloneqq \left(f(t+k)\right)_{k=1,\dots,p} \coloneqq \left(f_k(t)\right)_{k=1,\dots,p} \in \mathbb{R}^{p\times 1}$$

    - The state vector

$$\boldsymbol{x}(t) \coloneqq \begin{pmatrix} y_m(t) & \Delta u(t-1) & \Delta u(t-2) & \dots & \Delta u(t-N+1) \end{pmatrix}^T \in \mathbb{R}^{(N+1)\times 1}$$

Dynamic Matrix Control (DMC)

- Matrix formulation
  - Define
    - The matrix

$$F := \begin{pmatrix} 1 & g_2 - g_1 & g_3 - g_2 & \dots & g_{1+N} - g_N \\ 1 & g_3 - g_1 & g_4 - g_2 & \dots & g_{2+N} - g_N \\ 1 & g_4 - g_1 & g_5 - g_2 & \dots & g_{3+N} - g_N \\ \dots & \dots & \dots & \dots & \dots \\ 1 & g_{p+1} - g_1 & g_{p+2} - g_2 & \dots & g_{p+N} - g_N \end{pmatrix} \in \mathbb{R}^{p\times(N+1)}$$

» Note that $\boldsymbol{F}\boldsymbol{x}(t) = \left(f_k(t)\right)_{k=1,\dots,p} \in \mathbb{R}^{p\times 1}$

- Predicted output

$$\hat{\boldsymbol{y}} = \boldsymbol{G}\boldsymbol{u} + \boldsymbol{F}\boldsymbol{x}$$

Dynamic Matrix Control (DMC)

- Predicted output in case of measured disturbance $d(t)$
  - Step disturbance response model (stable, no integrators)
    - $D(z^{-1}) = d_1 z^{-1} + d_2 z^{-2} + \cdots + d_{N_d} z^{-N_d}$
  - Same computation as with the input case
- System dynamics

$$\hat{\boldsymbol{y}} = \boldsymbol{G}\boldsymbol{u} + \boldsymbol{f} + \boldsymbol{D}\boldsymbol{d} + \boldsymbol{f}_{\boldsymbol{d}}$$

  - Disturbance dynamic matrix

$$\boldsymbol{D} \coloneqq \begin{pmatrix} d_1 & 0 & 0 & \dots & 0 \\ d_2 & d_1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ d_p & d_{p-1} & d_{p-2} & \dots & d_{p-m+1} \end{pmatrix} \in \mathbb{R}^{p \times m}$$

  - vector of disturbance increments along the control horizon

$$\boldsymbol{d} \coloneqq \big(\Delta d(t+k|t)\big)_{k=1,\dots,m} \in \mathbb{R}^{m \times 1}$$

  - vector of free disturbance responses (known constants)

$$\boldsymbol{f}_d \coloneqq \big(f_d(t+k)\big)_{k=1,\dots,p} \in \mathbb{R}^{p \times 1}$$

- A prediction on the evolution of d is needed to compute the term $\boldsymbol{D}\boldsymbol{d}$ to obtain

$$\hat{\boldsymbol{y}} = \boldsymbol{G}\boldsymbol{u} + \boldsymbol{f}'$$

with free response $\boldsymbol{f}' = \boldsymbol{f} + \boldsymbol{D}\boldsymbol{d} + \boldsymbol{f}_{\boldsymbol{d}}$

Dynamic Matrix Control (DMC)

- Control algorithm
  - Least-square minimization
    - $e(t) \coloneqq y(t) - w(t)$: error
    - $\hat{e}(t+k|t) = \hat{y}(t+k|t) - \widehat{w}(t+k|t)$: predicted error
    - Cost function

$$J(\boldsymbol{u}) = \sum_{1}^{p} |\hat{e}(t+j|t)|^2 = \hat{\boldsymbol{e}}^T \hat{\boldsymbol{e}} = (\hat{\boldsymbol{y}} - \widehat{\boldsymbol{w}})^T (\hat{\boldsymbol{y}} - \widehat{\boldsymbol{w}})$$

with

$\hat{\boldsymbol{e}} \in \mathbb{R}^{p \times 1}$: vector of predicted errors along the prediction horizon p

    - (Unconstrained) optimization problem

$$\min_{\Delta u(t+i-1), i=1,2,\dots,m} J(\boldsymbol{u})$$

  - Usually the control effort is included in the cost function
    - Cost function

$$J(\boldsymbol{u}) = \sum_{1}^{p} |\hat{e}(t+j|t)|^2 + \lambda \sum_{1}^{p} \big(\Delta u(t+j-1|t)\big)^2 = \hat{\boldsymbol{e}}^T \hat{\boldsymbol{e}} + \lambda \boldsymbol{u}^T \boldsymbol{u}$$

with

$\boldsymbol{u} \in \mathbb{R}^{p \times 1}$: vector of future control increments along the control horizon m

$\lambda \in \mathbb{R}$: wheight of control effort minimization vs. error minimization

Dynamic Matrix Control (DMC)

- Control algorithm (no constraints)
  - (Unconstrained) Quadratic Programming problem

$$\min_{\Delta u(t+i-1), i=1,2,\dots,m} \hat{\boldsymbol{e}}^T\hat{\boldsymbol{e}} + \lambda \boldsymbol{u}^T\boldsymbol{u}$$

  - Closed-for solution for unconstrained problem and $\lambda = 0$
    - $J(\boldsymbol{u}) = \hat{\boldsymbol{e}}^T\hat{\boldsymbol{e}} = (\widehat{\boldsymbol{w}} - \widehat{\boldsymbol{y}})^T(\widehat{\boldsymbol{w}} - \widehat{\boldsymbol{y}}) = (\widehat{\boldsymbol{w}} - \boldsymbol{G}\boldsymbol{u} - \boldsymbol{f})^T(\widehat{\boldsymbol{w}} - \boldsymbol{G}\boldsymbol{u} - \boldsymbol{f})$
    - Gradient

$$\left.\frac{dJ(\boldsymbol{u})}{d\boldsymbol{u}}\right|_{\boldsymbol{u}=\boldsymbol{u}^*} = 2\boldsymbol{G}^T\boldsymbol{G}\boldsymbol{u}^* - 2\boldsymbol{G}^T(\widehat{\boldsymbol{w}} - \boldsymbol{f}) = 0$$

    - Optimal control sequence

$$\boldsymbol{u}^* = (\boldsymbol{G}^T\boldsymbol{G})^{-1}\boldsymbol{G}^T(\widehat{\boldsymbol{w}} - \boldsymbol{f})$$

      » Remark 1
      $(\boldsymbol{G}^T\boldsymbol{G})^{-1}\boldsymbol{G}^T$ is the pseudo-inverse of G; if G is square (i.e., $m = p$) and non-singular, the pseudo-inverse coincides with the inverse:

$$\boldsymbol{u}^* = \boldsymbol{G}^{-1}(\widehat{\boldsymbol{w}} - \boldsymbol{f})$$

      » Remark 2

      If the process has a delay d, the prediction horizon must be chosen such that

$$p \geq m + d$$

      otherwise the matrix $\boldsymbol{G}^T\boldsymbol{G}$ is singular

Dynamic Matrix Control (DMC)

- Control algorithm (no constraints)
  - (Unconstrained) Quadratic Programming problem
$$\min_{u(t+i-1),i=1,2,\ldots,m} \hat{\boldsymbol{e}}^T\hat{\boldsymbol{e}} + \lambda \boldsymbol{u}^T\boldsymbol{u}$$
  - Closed-for solution for unconstrained problem and $\lambda > 0$
    - $J(\boldsymbol{u}) = \hat{\boldsymbol{e}}^T\hat{\boldsymbol{e}} + \lambda \boldsymbol{u}^T\boldsymbol{u} = (\widehat{\boldsymbol{w}} - \widehat{\boldsymbol{y}})^T(\widehat{\boldsymbol{w}} - \widehat{\boldsymbol{y}}) + \lambda \boldsymbol{u}^T\boldsymbol{u}$
      - Gradient
$$\left.\frac{dJ(\boldsymbol{u})}{d\boldsymbol{u}}\right|_{\boldsymbol{u}=\boldsymbol{u}^*} = 2(\boldsymbol{G}^T\boldsymbol{G} + \lambda\boldsymbol{I})\boldsymbol{u}^* - 2\boldsymbol{G}^T(\widehat{\boldsymbol{w}} - \boldsymbol{f}) = 0$$
      - Optimal control sequence
$$\boldsymbol{u}^* = (\boldsymbol{G}^T\boldsymbol{G} + \lambda\boldsymbol{I})^{-1}\boldsymbol{G}^T(\widehat{\boldsymbol{w}} - \boldsymbol{f})$$

Dynamic Matrix Control (DMC)

- Control algorithm
  - Constrained problem
    - Generic j-th constraint on the output variables and/or on the control variables

$$\sum_{i=1}^{N} \left(c_y^{i,j} \hat{y}(t+i|t) + c_u^{i,j} u(t+i-1|t) + c_j \right) \leq 0$$

where

- $c_y^{i,j}$: coefficient of constraint j for the predicted output at time $t+i$
- $c_u^{i,j}$: coefficient of constraint j for the control variable at time $t+i-1$
- c_j: constant term of constraint j

  - Constrained quadratic programming

$$\begin{aligned} &\min_{u(t+i-1),i=1,2,\ldots,m} \hat{\boldsymbol{e}}^T \hat{\boldsymbol{e}} + \lambda \boldsymbol{u}^T \boldsymbol{u} \\ &s.t. \\ &\boldsymbol{R}\boldsymbol{u} \leq \boldsymbol{C} \end{aligned}$$

where

- $\boldsymbol{R} \in \mathbb{R}^{N_c \times 2N}$: coefficient matrix of the N_c constraints
- $\boldsymbol{C} \in \mathbb{R}^{N_c \times 1}$: matrix of the constant terms of the N_c constraints

Model Algorithmic Control (MAC)

- 80s
  - Richelet et al.
  - Analogous to the DMC but with impulse response model
  - Basic algorithm: $m = p$
- Prediction Models
  - Impulse-response process model
    - Applicable to stable processes without intergrators
    - The first N samples of the impulse response are considered
      - $m < N$
  - Constant disturbance model along the horizon

$$\hat{n}(t+k|t) = \hat{n}(t|t) = y_m(t) - \hat{y}(t|t), k = 1, \dots, m, \tag{14}$$

with

$$\hat{y}(t|t) = \sum_{i=1}^{N} h_i u(t-i) \tag{15}$$

Model Algorithmic Control (MAC)

- Predicted output

$$\hat{y}(t+k|t) = \sum_{i=1}^{N} h_i u(t+k-i|t) + \hat{n}(t+k|t), k = 1, \dots, m$$

  - Free and forced response

$$\hat{y}(t+k|t) = f_c(t+k) + f_f(t+k) + \hat{n}(t|t), k = 1, \dots, m$$

    - $f_c(t+k) = \sum_{i=1}^{k} h_i u(t+k-i|t)$
    - $f_f(t+k) = \sum_{i=k+1}^{N} h_i u(t+k-i)$

- $k = 1$
  - $f_c(t+1) = \underbrace{h_1 u(t|t)}_{\text{1 term (future control action)}}$
  - $f_f(t+1) = \underbrace{h_N u\big(t-(N-1)\big) + \cdots + h_3 u(t-2) + h_2 u(t-1)}_{N-1 \text{ terms (past control actions)}}$

Model Algorithmic Control (MAC)

- Predicted output

$$\hat{y}(t+k|t) = \sum_{i=1}^{k} h_i u(t+k-i|t) + \sum_{i=k+1}^{N} h_i u(t+k-i) + \hat{n}(t+k|t), k = 1, \ldots, m$$

  - Free and forced response
    - $k = 2$
      - $f_c(t+2) = \underbrace{h_2 u(t|t) + h_1 u(t+1|t)}_{\text{2 terms (future control actions)}}$
      - $f_f(t+2) = \underbrace{h_N u\big(t-(N-2)\big) + \cdots + h_4 u(t-2) + h_3 u(t-1)}_{N-2 \text{ terms (past control actions)}}$
    - …
    - $k = m$
      - $f_c(t+m) = \underbrace{h_m u(t|t) + \cdots + h_2 u(t+m-2|t) + h_1 u(t+m-1|t)}_{m \text{ terms (all future control actions)}}$
      - $f_f(t+m) = \underbrace{h_N u\big(t-(N-m)\big) + \cdots + h_{m+2} u(t-2) + h_{m+1} u(t-1)}_{N-m \text{ terms (past control actions)}}$

Model Algorithmic Control (MAC)

- Matrix formulation
  - Define
    - the vector of output predictions along the prediction horizon

$$\hat{\boldsymbol{y}} := \left(\hat{y}(t+k|t)\right)_{k=1,\dots,m} \in \mathbb{R}^{m\times 1}$$

    - the vector of $(N-1)$ past control actions

$$\boldsymbol{u}_- := \left(u(t+k-1)\right)_{k=-N+2,-N+3,\dots,0} = \begin{pmatrix} u(t+1-N) \\ u(t+2-N) \\ \dots \\ u(t-1) \end{pmatrix} \in \mathbb{R}^{(N-1)\times 1}$$

    - the vector of candidate future control actions along the control horizon

$$\boldsymbol{u}_+ := \left(u(t+k-1)\right)_{k=1,\dots,m} = \begin{pmatrix} u(t|t) \\ u(t+1|t) \\ \dots \\ u(t+m-1|t) \end{pmatrix} \in \mathbb{R}^{m\times 1}$$

    - the vector of disturbances

$$\boldsymbol{n} := \left(n(t+k)\right)_{k=1,\dots,m} \in \mathbb{R}^{m\times 1}$$

Model Algorithmic Control (MAC)

- Matrix formulation
  - Define
    - The matrices

$$\boldsymbol{H}_1 := \begin{pmatrix} h_1 & 0 & \dots & 0 \\ h_2 & h_1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ h_m & h_{m-1} & \dots & h_{+1} \end{pmatrix} \in \mathbb{R}^{m \times m}$$

$$\boldsymbol{H}_2 := \begin{pmatrix} h_N & h_{N-1} & \dots & h_i & \dots & h_2 \\ 0 & h_N & \dots & h_{i+1} & \dots & h_3 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & h_N & \dots & h_{m+1} \end{pmatrix} \in \mathbb{R}^{m \times (N-1)}$$

- Predicted output

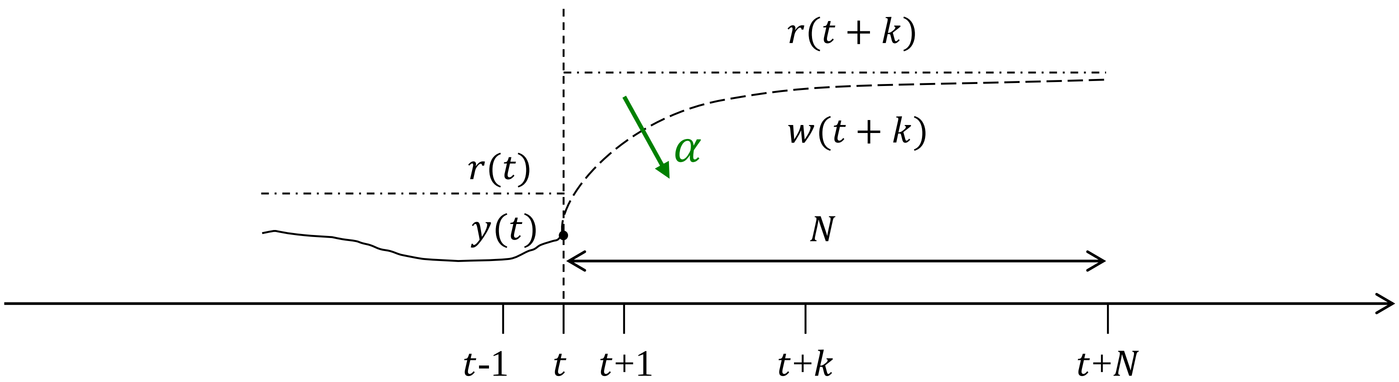
$$\hat{\boldsymbol{y}} = \boldsymbol{H}_1 \boldsymbol{u}_+ + \boldsymbol{H}_2 \boldsymbol{u}_- + \boldsymbol{n}$$

Model Algorithmic Control (MAC)

- Control algorithm
  - Reference trajectory computed from the reference signal

$$w(t+k) = \begin{cases} w(t) = y(t) \\ \alpha w(t+k-1) + (1-\alpha) r(t+k), k = 1, \dots, m \end{cases}$$

    - $r(t+k) = r(t), \forall k = 1,2,\dots,N$ if If the reference signal is not known a priori
    - The parameter $\alpha \in (0,1)$ determines the desired speed of the approach to the setpoint, i.e., the performance/robustness trade-off
      - » $\alpha \uparrow \Rightarrow$ smooth approach $\Rightarrow$ more robust system
      - » $\alpha \downarrow \Rightarrow$ aggressive approach $\Rightarrow$ faster system



Model Algorithmic Control (MAC)

- Control algorithm
  - Least-square minimization
    - Predicted error
      - $\hat{e}(t+k|t) = w(t+k|t) - \hat{y}(t+k|t)$
        $= w(t+k|t) - \boldsymbol{H}_2\boldsymbol{u}_- - \boldsymbol{H}_1\boldsymbol{u}_+ - \boldsymbol{n}$
        $= w(t+k|t) - f(t+k|t) - \boldsymbol{H}_1\boldsymbol{u}_+$
        $= w(t+k|t) - f(t+k|t) - \boldsymbol{H}_1\boldsymbol{u}_+$

      with

      $\boldsymbol{f} = \left(f(t+k|t)\right)_{k=1,\dots,m} := \boldsymbol{H}_2\boldsymbol{u}_- + \boldsymbol{n}$

      - Recalling that

        $\hat{n}(t+k|t) = \hat{n}(t|t) = y_m(t) - \sum_{i=1}^{N} h_i u(t-i)\,, k = 1, \dots, m$

      we can say that $\mathbf{f}$ collects all the known values

      » Depends on past inputs and on current and past outputs

Model Algorithmic Control (MAC)

- Control algorithm
  - Least-square minimization
    - Cost function
$$J(\boldsymbol{u}) = \hat{\boldsymbol{e}}^T \hat{\boldsymbol{e}} + \lambda \boldsymbol{u}_+^T \boldsymbol{u}_+$$
      - Optimal control sequence
$$\boldsymbol{u}_+^* = (\boldsymbol{H}_1^T \boldsymbol{H}_1 + \lambda \boldsymbol{I})^{-1} {\boldsymbol{H}_1}^T (\boldsymbol{w} - \boldsymbol{f})$$
    - The computation of the optimal control actions implies a (simple) inversion of the square matrix $(\boldsymbol{H}_1^T \boldsymbol{H}_1 + \lambda \boldsymbol{I}) \in \mathbb{R}^{\boldsymbol{m} \times \boldsymbol{m}}$
    - Since $p = m$, if $\lambda = 0$ and there is no dead-time, the computation of the optimal control actions just implies the inversion $\boldsymbol{H}_1^{-\boldsymbol{1}} = (\boldsymbol{H}_1^T \boldsymbol{H}_1)^{-\boldsymbol{1}} {\boldsymbol{H}_{\boldsymbol{1}}}^T$
    - If $\lambda = 0$ and there is a dead-time, $\boldsymbol{H}_1^T \boldsymbol{H}_1$ is singular
      - E.g., dead-time $\theta = 2$ ($\Rightarrow h_1 = h_2 = 0$), $m = 4$:

$$\boldsymbol{H}_1 \coloneqq \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ h_3 & 0 & 0 & 0 \\ h_4 & h_3 & 0 & 0 \end{pmatrix} \in \mathbb{R}^{4 \times 4}$$

Predictive Functional Control (PFC)

- 80s
  - Richelet et al.
  - Proposed for fast processes
    - The control signal is *structured* as a linear combination of *base functions*
    - The cost function is evaluated only in a limited number of *coincidence points* along the prediction horizon
- Prediction Models
  - State-space process model
    - Applicable to unstable processes with some awareness
      - The idea is to allow only control signals which stabilize the process

Predictive Functional Control (PFC)

- Prediction model
  - LTI state-space model
    - $\begin{cases} x(t) = Mx(t-1) + Nu(t-1) \\ y(t) = Qx(t) \end{cases}$
  - Disturbance model
    - E.g., constant disturbance model along the horizon
      - $\hat{n}(t+k|t) = \hat{n}(t|t) = y_m(t) - \hat{y}(t|t), k = 1, \dots, p,$

      with
      - $\hat{y}(t|t) = Qx(t)$

  - Remark
    - Delays may be implicit in the model
      - Example

$$\begin{cases} x(t) = \begin{bmatrix} 0.5 & 0 \\ -0.1 & 0.2 \end{bmatrix} x(t-1) + \begin{bmatrix} 0.1 \\ 0 \end{bmatrix} u(t-1) \\ y(t) = \begin{bmatrix} 0 & 4 \end{bmatrix} x(t) \end{cases}$$

Predictive Functional Control (PFC)

- Structured control law

$$u(t+k|t) \coloneqq \sum_{i=1,\dots,n_B} \mu_i(t) B_i(k)$$

  - $\mu_i(t)$: coefficient of base function i computed at time t
  - n_B: number of base functions
  - $B_i(k)$: value of base function i at time k

  – The choice of the base function set defines the input profile
    - E.g., it can be used to enforce a predetermined behaviour in terms of smoothness

  – Example
    - Polynomial base functions
      - $\mathcal{B} = \{B_i = k^{i-1}, i = 1, \dots, n_B\}$
        » If $n_B = 3 \Rightarrow \mathcal{B} = \{1, k, k^2\}$
          » it can express the majority of reference signals

Predictive Functional Control (PFC)

- Predicted output

  1) System response to the base functions $B_i(t)$ with initial state $x_{B_i}(0) = 0, i = 1, \dots, n_B$

     – From the state-space model:

$$\begin{cases} x_{B_i}(1) = M x_{B_i}(0) + N B_i(0) = N B_i(0) \\ y_{B_i}(1) = Q x_{B_i}(1) = Q N B_i(0) \end{cases}, i = 1, \dots, n_B$$

$$\begin{cases} x_{B_i}(2) = M x_{B_i}(1) + N B_i(1) = M N B_i(0) + N B_i(1) \\ y_{B_i}(2) = Q x_{B_i}(2) = Q M N B_i(0) + Q N B_i(1) \end{cases}, i = 1, \dots, n_B$$

…

$$\begin{cases} x_{B_i}(k) = M^{k-1} N B_i(0) + M^{k-2} N B_i(1) + \cdots + N B_i(k-1) \\ y_{B_i}(k) = Q M^{k-1} N B_i(0) + Q M^{k-2} N B_i(1) + \cdots + Q N B_i(k-1) \end{cases}, i = 1, \dots, n_B$$

Predictive Functional Control (PFC)

- Predicted output

2) System response to $u(t+k|t) = \sum_{i=1,\dots,n_B} \mu_i(t) B_i(k)$

$$\begin{cases} x(t+1) = Mx(t) + Nu(t) = Mx(t) + \sum_{i=1,\dots,n_B} NB_i(0)\,\mu_i(t) \\ y(t+1) = Qx(t+1) = QMx(t) + \sum_{i=1,\dots,n_B} QNB_i(0)\,\mu_i(t) \\ \qquad = QMx(t) + \sum_{i=1,\dots,n_B} y_{B_i}(1)\,\mu_i(t) \end{cases}$$

$$\begin{cases} x(t+2) = Mx(t+1) + Nu(t+1) \\ \qquad = M^2x(t) + \sum_{i=1,\dots,n_B} MNB_i(0)\,\mu_i(t) + \sum_{i=1,\dots,n_B} NB_i(1)\,\mu_i(t) \\ y(t+2) = Qx(t+2) \\ \qquad = QM^2x(t) + \sum_{i=1,\dots,n_B}\big(QMNB_i(0) + QNB_i(1)\big)\,\mu_i(t) \\ \qquad = QM^2x(t) + \sum_{i=1,\dots,n_B} y_{B_i}(2)\,\mu_i(t) \end{cases}$$

...

$$\begin{cases} x(t+k) = M^kx(t) + \sum_{i=1,\dots,n_B}\big(M^{k-1}NB_i(0) + M^{k-2}NB_i(1) + \cdots + NB_i(k-1)\big)\,\mu_i(t) \\ y(t+k) = QM^kx(t) + \sum_{i=1,\dots,n_B} y_{B_i}(k)\,\mu_i(t) \end{cases}$$

– $$\boxed{\hat{y}(t+k|t) = QM^kx(t) + \sum_{i=1,\dots,n_B} y_{B_i}(k)\,\mu_i(t) + \hat{n}(t+k|t)}$$

Predictive Functional Control (PFC)

- Control law
  - Least-square minimization
    - $\hat{e}(t+k|t) = \hat{y}(t+k|t) - \widehat{w}(t+k|t)$: predicted error
  - Coinicidence points
    - The response is evaluated only over a limited number of time instants
      - $\mathcal{C} = \{h_1, h_2, \dots, h_{N_H}\}$: set of coincident points
  - Cost function

$$J(\boldsymbol{u}) = \sum_{j=1}^{n_H} \left|\hat{e}\left(t+h_j|t\right)\right|^2 + \lambda \sum_{j=1}^{n_H} \left(\Delta u\left(t+h_j-1|t\right)\right)^2 = \hat{\boldsymbol{e}}^T\hat{\boldsymbol{e}} + \lambda \boldsymbol{u}^T\boldsymbol{u}$$

with

$\hat{\boldsymbol{e}} \in \mathbb{R}^{n_H \times 1}$: vector of predicted errors along the prediction horizon p

$\boldsymbol{u} \in \mathbb{R}^{n_H \times 1}$: vector of future control increments along the control horizon m

$\lambda \in \mathbb{R}$: wheight of control effort minimization vs. error minimizatio

Predictive Functional Control (PFC)

- Control law
  - Unconstrained optimization problem ($\lambda = 0$)

$$\min_{u(t+h_i-1),i=1,2,\dots,n_H} J(\boldsymbol{u})$$

$$J(\boldsymbol{u}) = \sum_{j=1}^{n_H} \left|\hat{y}(t+h_j|t) - \hat{w}(t+h_j|t)\right|^2$$
$$= \sum_{j=1}^{n_H} \left|QM^{h_j}x(t) + \sum_{i=1,\dots,n_B} y_{B_i}(h_j)\,\mu_i(t) + \hat{n}(t+k|t) - \hat{w}(t+h_j|t)\right|^2$$

- Define
  - $\boldsymbol{y}_B(h_j) := \begin{pmatrix} y_{B_1}(h_j) & \dots & y_{B_{n_B}}(h_j) \end{pmatrix} \in \mathbb{R}^{1\times n_B}$
  - $\boldsymbol{\mu}(t) := \begin{pmatrix} \mu_1(t) \\ \dots \\ \mu_{n_B}(t) \end{pmatrix} \in \mathbb{R}^{n_B\times 1}$
  - $d(t+h_j) := \hat{w}(t+h_j|t) - QM^{h_j}x(t) - \hat{n}(t+k|t)$

- $J(\boldsymbol{\mu}) = \sum_{j=1}^{n_H} \left|\boldsymbol{y}_B(h_j)\boldsymbol{\mu}(t) - d(t+h_j)\right|^2$

Predictive Functional Control (PFC)

- Control law
  - Cost function with $\lambda = 0$

$$J(\boldsymbol{\mu}) = \sum_{j=1}^{n_H} \left| \boldsymbol{y}_B(h_j)\boldsymbol{\mu}(t) - d(t+h_j) \right|^2$$

- Define
  - $\boldsymbol{Y}_{\boldsymbol{B}} := \begin{pmatrix} \boldsymbol{y}_B(h_1) \\ \dots \\ \boldsymbol{y}_B(h_{n_H}) \end{pmatrix} \in \mathbb{R}^{n_H \times n_B}$
  - $\boldsymbol{d}(t) := \begin{pmatrix} d(t+h_1) \\ \dots \\ d(t+h_{n_H}) \end{pmatrix} \in \mathbb{R}^{n_H \times 1}$

- $J(\boldsymbol{\mu}) = (\boldsymbol{Y}_{\boldsymbol{B}}\boldsymbol{\mu} - \boldsymbol{d})^T(\boldsymbol{Y}_{\boldsymbol{B}}\boldsymbol{\mu} - \boldsymbol{d}) = \boldsymbol{\mu}^T\boldsymbol{Y}_{\boldsymbol{B}}^T\boldsymbol{Y}_{\boldsymbol{B}}\boldsymbol{\mu} + \boldsymbol{d}^T\boldsymbol{d} - 2\boldsymbol{\mu}^T\boldsymbol{Y}_{\boldsymbol{B}}^T\boldsymbol{d}$

Predictive Functional Control (PFC)

- Control law
  - Cost function with $\lambda = 0$

$$J(\boldsymbol{\mu}) = (\boldsymbol{Y_B\mu} - \boldsymbol{d})^T(\boldsymbol{Y_B\mu} - \boldsymbol{d}) = \boldsymbol{\mu}^T\boldsymbol{Y}_{\boldsymbol{B}}^T\boldsymbol{Y_B\mu} + \boldsymbol{d}^T\boldsymbol{d} - 2\boldsymbol{\mu}^T\boldsymbol{Y}_{\boldsymbol{B}}^T\boldsymbol{d}$$

    - Gradient:

$$\frac{dJ(\boldsymbol{\mu})}{d\boldsymbol{\mu}} = 2\boldsymbol{Y}_{\boldsymbol{B}}^T\boldsymbol{Y_B\mu} - 2\boldsymbol{Y}_{\boldsymbol{B}}^T\boldsymbol{d}$$

    - Optimal coefficients

$$\boldsymbol{Y_B\mu}^* = \boldsymbol{d} \Rightarrow \boldsymbol{\mu}^* = \left(\boldsymbol{Y}_{\boldsymbol{B}}^{\boldsymbol{T}}\boldsymbol{Y}_{\boldsymbol{B}}\right)^{-\boldsymbol{1}}\boldsymbol{Y}_{\boldsymbol{B}}^{\boldsymbol{T}}\,\boldsymbol{d}$$

  - Cost function with $\lambda \neq 0$

$$J(\boldsymbol{\mu}) = (\boldsymbol{Y_B\mu} - \boldsymbol{d})^T(\boldsymbol{Y_B\mu} - \boldsymbol{d}) + \lambda\boldsymbol{u}^T\boldsymbol{u}$$

    - Optimal coefficients

$$\boldsymbol{\mu}^* = \left(\boldsymbol{Y}_{\boldsymbol{B}}^{\boldsymbol{T}}\boldsymbol{Y}_{\boldsymbol{B}} + \lambda\boldsymbol{I}\right)^{-\boldsymbol{1}}\boldsymbol{Y}_{\boldsymbol{B}}^{\boldsymbol{T}}\,\boldsymbol{d}$$

  - Computation of the first control action

$$u^*(t|t) \coloneqq \sum_{i=1,\dots,n_B} \mu_i^*(t)B_i(0)$$

MPC scheme comparison

- DMC
  - Step response model
    - No a priori knowledge of the process required
  - On-line computation feasible depending on the number of samples of the step response
    - Quadratic cost function
  - Not applicable to unstable processes and to processes with integrators
- MAC
  - Equivalent to DMC with impulse response process model
  - Simplified by choosing the control horizon equal to the prediction horizon
- PFC
  - State-space process model
    - It requires some a priori knowledge of the process
  - The on-line computational burden is negligible
    - Quadratic cost function
    - PFC replaces objective optimization by forcing a subset of the predictions samples to match the set point trajectory at given time instants, named coincident points
    - The control actions are expressed as linear combinations of base fiunctions and zre parameterized in an equivalent way to the set point
  - It can be used with unstable and nonlinear processes
  - Accuracy depends on the number and on the choice of the coincident points

Summary

- MPC commercial schemes introduced
  - DMC
    - Step response model
    - Quadratic cost function
  - MAC
    - Impulse response model
    - Quadratic cost function
  - PFC
    - State space model
    - Structured control law
    - Quadratic cost function