LEARNING PREDICTIVE CONTROL USING PROBABILISTIC MODELS

Enso Ikonen
Systems Engineering Laboratory
Department of Process and Environmental Engineering, University of Oulu, Finland

Outline / Contents:

- Controlled Markov Chain models (CMC)
- Identification/prediction with CMC-ARX
- Predictive control
  - Maximize confidence in that the control action minimizes the costs
- Learning (dual) control
  - Maximize increase in confidence
  - Upper bound approximation
- Simulation illustrations
- Conclusions

CMC-ARX MODELS

- CMC-ARX discrete-time model:

$$y(k+1) = f(\mathbf{x}(k), \mathbf{u}(k)) + e(k+1)$$

- SISO (for simplicity): Discretized process variables:

$$\mathbf{x}(k) = \begin{bmatrix} y(k) \\ y(k-1) \\ \vdots \\ y(k-n_A+1) \\ u(k-1) \\ \vdots \\ u(k-n_B+1) \end{bmatrix}$$

$$i = \arg\min_l \left\| \mathbf{x} - \tilde{\mathbf{x}}_l \right\|; i \in \mathrm{I} = \{1,2,\ldots,I\}$$

$$j = \arg\min_l \left\| y - \tilde{y}_l \right\|, j \in \mathrm{J} = \{1,2,\ldots,J\}$$

$$u \in \tilde{u}_a; a \in \mathrm{A} = \{1,2,\ldots,A\}$$

- At instant *k*, the system is at state *i*(*k*), control action *a*(*k*) is applied and the system output will be *j*(*k*+1).

Formulate and solve:

I *Predictive control* II *'Optimal' learning*

IDENTIFICATION & PREDICTION

Frequentist model (*counters*):

$$C_{j,i,a}(k)=\begin{cases} C_{j,i,a}(k-1)+1 & \text{if } i(k-1)=i,\ a(k-1)=a \text{ and } j(k)=j \\ C_{j,i,a}(k-1) & \text{otherwise} \end{cases}$$

In general:

$$C_{j,i,a}(k)=C_{j,i,a}(k-1) + q_j(k)r_i(k-1)s_a(k-1)$$

where $q_j(k)$, $r_j(k)$ and $s_j(k)$ are elements of distributions (discrete densities) for output, state, and control.

(Controlled) Markov *transition probabilities*:

$$p_{j,i,a}=\frac{C_{j,i,a}(k)}{\sum_{j=1}^{J} C_{j,i,a}(k)}$$

Predicted distribution for next output (element *j*):

$$\hat{q}_j(k+h+1)=\sum_{i=1}^{I} p_{j,i,\overline{a}}(k)r_i(k+h)$$

and state (element *i*):

$$r_i(k+h)=q_{j_0}(k+h)\cdots q_{j_{n_A-1}}(k+h-n_A+1) \cdot s_{a_1}(k+h-1)\cdots s_{a_{n_B-1}}(k+h-n_B+1)$$

Apply recursively for *h* = 0, 1, 2, ...

PREDICTIVE CONTROL (1)

Simplified cost function (mean level set-point control, no costs for control):

$$J = \sum_{h=1}^{H_\mathrm{p}} \left[\overline{w}(k) - y(k+h)\right]^2$$

$$\Delta u(k+h) = 0 \, \forall h \geq 1$$

Distribution of plant output translates into distribution of costs.

Interpretation in frequentist framework:

$$J_j = \frac{\sum_{h=1}^{H_\mathrm{p}} \hat{q}_j(k+h)}{\sum_{j=1}^{J} \sum_{h=1}^{H_\mathrm{p}} \hat{q}_j(k+h)} \left[\overline{w}(k) - \tilde{y}_j\right]^2$$

$$= \hat{f}_j \left[\overline{w}(k) - \tilde{y}_j\right]^2$$

Notation:

$\hat{f}_j(\overline{a}) := \hat{f}_j \mid u(k+h) = \tilde{u} \, \forall h \geq 0_{\overline{a}}$

and $\hat{\mathbf{f}}(\overline{a})$ is ordered in ascending order.

PREDICTIVE CONTROL (2)

Solve predictive control problem by comparing distributions $\hat{\mathbf{f}}(\overline{a})$, for different $\overline{a} \in \mathsf{A}$.

Measure of confidence:

$$S(\overline{a}) = \Pr\{\hat{\mathbf{f}}(\overline{a}) \le \hat{\mathbf{f}}(1) \text{and} \ldots \text{and} \hat{\mathbf{f}}(\overline{a}) \le \hat{\mathbf{f}}(\overline{a}-1)$$
$$\hat{\mathbf{f}}(\overline{a}) \le \hat{\mathbf{f}}(\overline{a}+1) \text{and} \ldots \text{and} \hat{\mathbf{f}}(\overline{a}) \le \hat{\mathbf{f}}(A)\}$$

Simple to estimate:

$$\hat{R}_{a,b} = \sum_{j=1}^{J} \hat{f}_j(b) \left[\sum_{j=i}^{j} \hat{f}_i(\overline{a}) \right]$$

and

$$\hat{S}(\overline{a}) = \prod_{b=1, b \ne \overline{a}}^{A} \hat{R}_{b,\overline{a}}$$

Optimal *control action* is given by

$$a^* = \arg\max_{\overline{a}} \hat{S}(\overline{a})$$

LEARNING CONTROL

- If no full confidence on model **C**, more information must be gained.
- Use statistics **C** to 'optimally' design plant inputs (cf. dual control).
- Idea: Maximize increase in confidence $\hat{S}(a^*)$

- Estimate variability of $\hat{S}(a^*)$ using a bound of the largest possible change during one update (heuristic)
- Relative to:

$$V(c) = \begin{cases} 1 + \dfrac{\Delta(k+1)}{\hat{R}_{a^*,c}(k)} & \text{if } c \neq a^* \\ \prod_{b=1, b \neq a^*}^{A} \left[1 + \dfrac{\Delta(k+1)}{\hat{R}_{c,b}(k)} \right] & \text{if } c = a^* \end{cases}$$

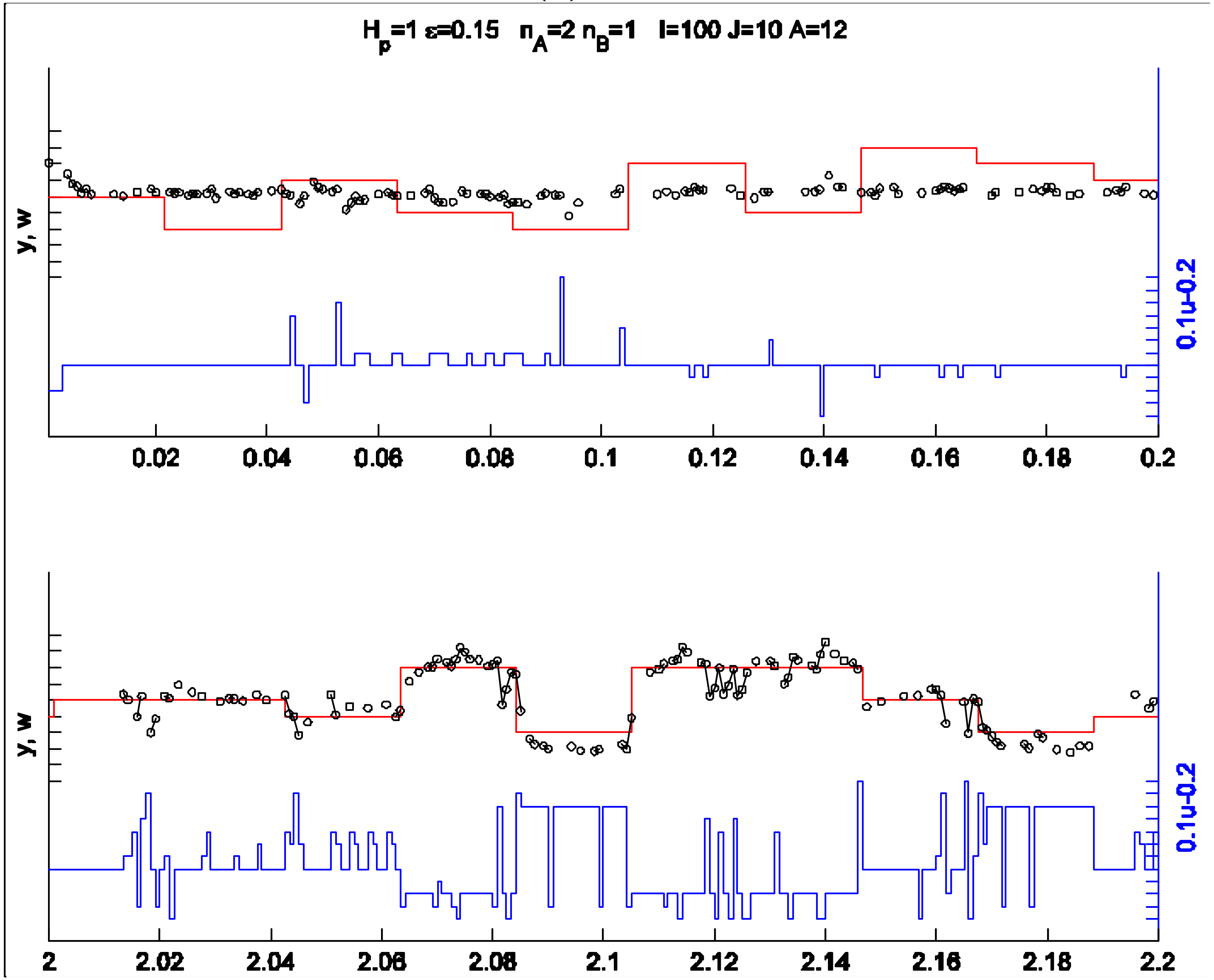
where $\Delta(k+1) = \sum_{a=1}^{A} \left[\sum_{i=1}^{I} \frac{1}{\sum_{j=1}^{J} C_{j,i,a}(k)} r_i(k) \right] s_a(k)$

(see paper for proof)

Proposition:
For 'optimal' learning, select:

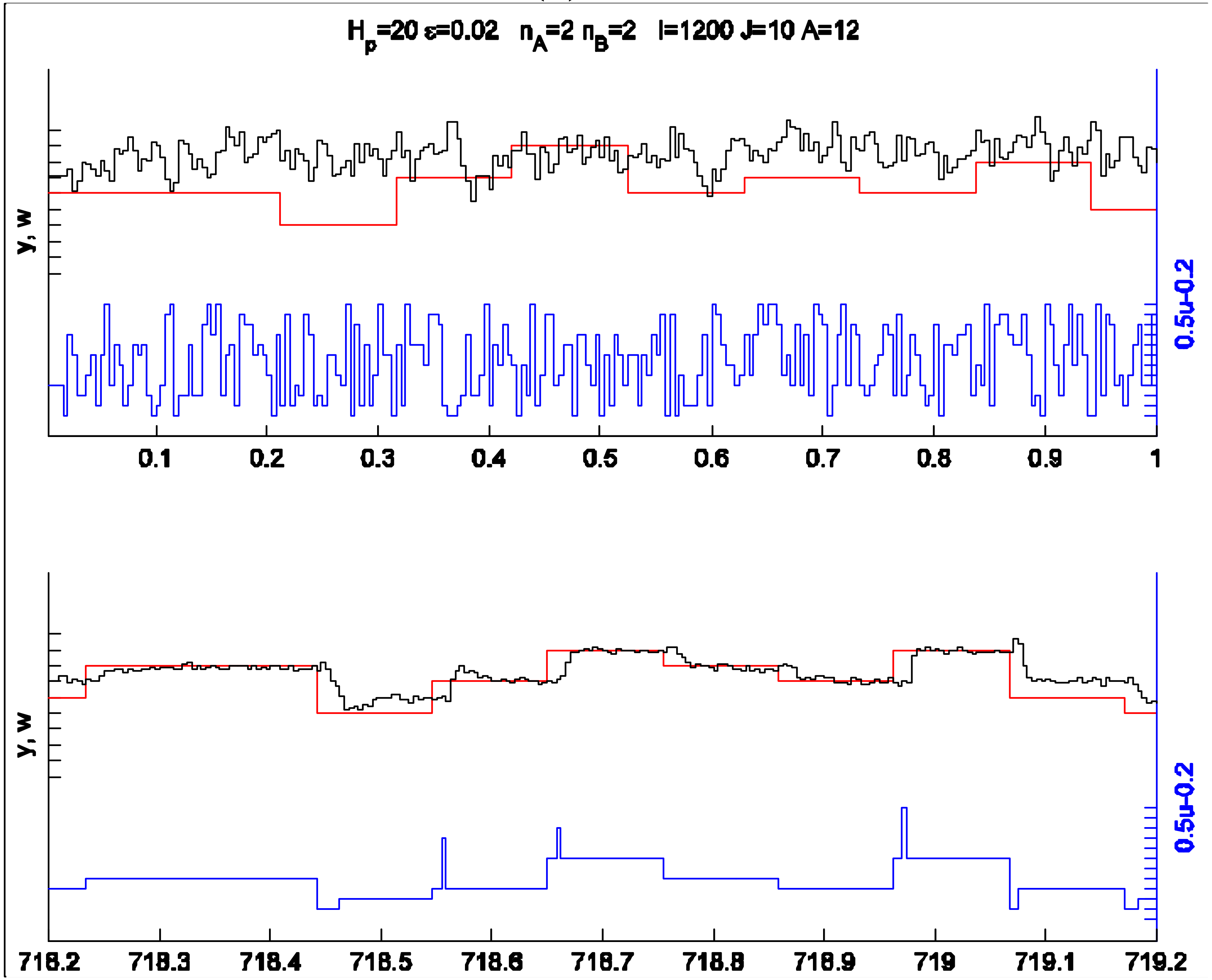
$$a(k) = \arg\max_{c} V(c)$$

NUMERICAL EXAMPLE (1)



Missing data (50% output data corrupted)

NUMERICAL EXAMPLE (2)



Nonminimum phase plant

CONCLUSIONS

Summary:
- CMC-ARX
- Predictive control
- Learning control
- Numerical illustration

- *Discussion*:
  - non-linear multiple-input stochastic systems
  - missing data problems
  - curse of dimensionality (lack of data)