Model predictive control of industrial processes

Vitali Vansovitš
Contents

- Industrial process (Iru Power Plant)
- Neural network identification
- Process identification – linear model
- Model predictive controller
- Results
- Future plans
Industrial process (Iru Power Plant)

- Three similar boilers in parallel
- Each boiler can work in:
  - basic mode (constant load)
  - control mode (variable load)
- Control mode is used to control plant output temperature
Industrial process control

Two cascade control loop
Cascade control loop
Control loop
Industrial process control (Iru Power Plant)

DH water flow → Boiler water flow → CALC₂

Setpoint → C₁ → CALC₁ → + → C₂ → - → C₃

Boiler output temperature

Gas flow → P₃ → P₂ → P₁ → Plant output temperature

C₃: kp = 0.25; ti = 20
C₂: kp = 0.9; ti = 240
C₁: kp = 0.5; ti = 180

Boiler output temperature

CALC₃ → + → C₄ → - → P₄

Air pressure

CALC₄ → + → C₅ → - → P₅

Air flow
Industrial process (Iru Power Plant)
Neural network model
Recursive prediction

\[ \hat{y}(k - 1) \]
\[ \hat{y}(k - 2) \]
\[ \hat{y}(k - 4) \]
\[ u(k - 1) \]
\[ u(k - 2) \]
\[ u(k - 4) \]

\[ \hat{y}(k) \]

\[ y \] – output temperature prediction
\[ u \] – vector of water, air and gas flows
Neural network 5 min prediction
Neural network first results

- 5 minutes predictions were good
- Simulation was not possible
  - After 10 cycles model output was not close enough to real process output
  - After 100 cycles model output was already somewhere in outer space
- So, it was decided to start from simple things – linear model identification with conventional methods.
Process identification

- Process is simple, but nonlinear:
  \[
  Q \approx r \cdot \Delta m_{\text{fuel}} \\
  Q \approx c \cdot \Delta m_{\text{water}} \Delta T \\
  \Delta T = \frac{r \cdot \Delta m_{\text{fuel}}}{c \cdot \Delta m_{\text{water}}}
  \]

- As water flow is more or less constant during normal operation, so mainly it can be seen as linear.

- Matlab Identification Toolbox was used for identification.

- Most suitable results were acquired from polynomial and state space models.
Polynomial model identification
State-space model
State space model with 4 states (ss2) was selected as:

- it is simpler than oe221 (6 states in state space representation)
- shows better results than ss1 (1 state)

4 state model showed better fit among all state space models (2, 3, 5 etc. states)
MPC (Model predictive controller)
Model identification

- To avoid perfect “theoretical results” model for MPC was identified from the data acquired one year later at the time when boiler load was different (~1/8 of previous example) and nonconstant.
MPC structure
MPC observer

\[
\hat{x}(k | k) = \hat{x}(k | k-1) + L[y(k) - \hat{y}(k | k-1)] \\
\hat{x}(k + 1 | k) = A\hat{x}(k | k) + Bu(k) \\
\hat{y}(k | k-1) = C\hat{x}(k | k-1)
\]

\[
\hat{x}(k + 1 | k) = A(I - L'C)\hat{x}(k | k-1) + Bu(k) + AL'y(k) \\
e(k) = x(k) - \hat{x}(k | k-1) \quad e(k + 1) = (A - LC)e(k)
\]

Observer is stable \(\Rightarrow\) estimation error converges to zero
MPC design (1)

- There is a cost function: \[ V(k) = \|Z(k) - T(k)\|_Q^2 + \|\Delta U(k)\|_R^2 \]

- where
  - \( Z(k) \) – predicted process output
  - \( T(k) \) – reference trajectory
  - \( \Delta U(k) \) – process input changes
  - \( Q, R \) – weight matrices
  - \( \|a\|_A^2 = a^T A a \)

\[
Z(k) = \Psi x(k) + Yu(k - 1) + \Theta \Delta U(k)
\]

\[
E(k) = T(k) - \Psi x(k) - Yu(k - 1)
\]

Tracking error

Free response with \( \Delta U = 0 \)
MPC design (2)

- From this function optimal input changes for unconstrained case are found:

\[ \Delta U(k) = \frac{1}{2} H^{-1}G \]

- where

\[ G = 2\Theta^T Q E(k) \]

\[ H = \Theta^T Q \Theta + R \]
MPC constraints

- Constraints are in the form:

\[
E\left[ \begin{bmatrix} \Delta U(k) \\ 1 \end{bmatrix} \right] \leq 0, \quad F\left[ \begin{bmatrix} U(k) \\ 1 \end{bmatrix} \right] \leq 0, \quad G\left[ \begin{bmatrix} Z(k) \\ 1 \end{bmatrix} \right] \leq 0
\]

- Which can be transformed to: \( W\Delta U(k) \leq w \)
- So, we solve constrained optimization problem:

\[
\text{minimize} \quad \Delta U(k)^T \ H \Delta U(k) - G^T \Delta U(k)
\]

subject to inequality constraint.
- Quadratic programming problem
MPC for water boiler

- 1 manipulated process input (gas flow)
- 2 measured process disturbances (water & air flows)
- 1 measured process output

- Prediction horizon 10 steps
- Control horizon 2 steps

- Gas flow constraints \((m^3/h)\):
  - \(\text{min} = 0\)
  - \(\text{max} = 16000\)
  - \(\text{max down rate} = 2000\)
  - \(\text{max up rate} = 2000\)

- Gas flow rate weight = 0.01
- Process output weight = 10
MPC for water boiler (Simulink)
Outputs comparison
Control quality
Gas consumption
Earn some money

- Same control quality, but less raw materials
Setpoint decreased 0,6 °C (1)

Output comparison
Setpoint decreased 0.6 °C (2)
Gas consumption difference

~8000 m³
Setpoint decreased 0,6 °C (3)
Sum of errors
Very rough estimation

- Gas price 400 € / 1000 m³
- Difference 8000 m³
- Period 16 hours
- Season length 50 days

- Economy
  - $\text{Economy} = 50 \times \left(\frac{24}{16}\right) \times 8000 \times 400 / 1000$
  - $= 240000$ € / season
Some results

- With MPC control became better:
  - Sum of errors of output are twice lower
- MPC with bad model is better than PI cascade
- Raw material consumption is on the same level
- If we change setpoint then we can decrease raw material consumption with the same control quality as PI loops have.
Future plans

- Compare MPC with simulated PI loop
- Minimize raw material consumption with MPC (not setpoint change)
- Find better way of identification
- Identify all parts of the process (including actuators)
- Make model based control closer to reality
- Try controller on real plant some day
- Etc.
References

- Real life
Thank you!

Questions???