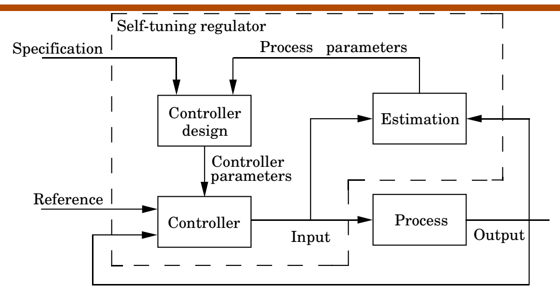
Adaptive control

Tore Hägglund

Dept. of Automatic Control
Lund University

Contents

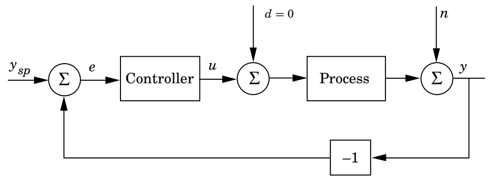
Adaptive control



Many versions:

- Many control structures
- Many estimation methods
- Many design methods

Process model



Note - No load disturbances!

Process model

$$y(t) = a_1 y(t-1) + a_2 y(t-2) + \ldots b_1 u(t-1) + b_2 u(t-2) + \ldots + n(t)$$

$$y(t) = \theta^T \varphi(t) + n(t)$$

where

$$\theta = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ b_n \end{pmatrix} \qquad \varphi(t) = \begin{pmatrix} y(t-1) \\ y(t-2) \\ \vdots \\ u(t-n) \end{pmatrix}$$

Note:

- Sampling period important!
- Simple model, no load disturbances

Least squares estimation

Find an estimate $\hat{\theta}$ of θ that minimizes

$$\sum_{i=1}^{n} (y(i) - \hat{y}(i))^2$$

where

$$\hat{y}(t) = \hat{\theta}^T \varphi$$

Recursive least-squares estimation

$$\hat{\theta}(t) = \hat{\theta}(t-1) + P(t)\varphi(t)\epsilon(t)$$
$$\epsilon(t) = y(t) - \hat{y}(t) = y(t) - \varphi(t)^T \hat{\theta}(t-1)$$
$$P(t) = P(t-1) - \frac{P(t-1)\varphi(t)\varphi(t)^T P(t-1)}{1 + \varphi(t)^T P(t-1)\varphi(t)}$$

$P(t)$ covariance matrix.

$$\lim_{t\to\infty} P(t) = 0$$

Recursive least-squares estimation

$$P(t) = P(t-1) - \frac{P(t-1)\varphi(t)\varphi(t)^T P(t-1)}{1 + \varphi(t)^T P(t-1)\varphi(t)}$$

Forgetting:

$$P(t) = \frac{1}{\lambda}\left(P(t-1) - \frac{P(t-1)\varphi(t)\varphi(t)^T P(t-1)}{\lambda + \varphi(t)^T P(t-1)\varphi(t)}\right)$$

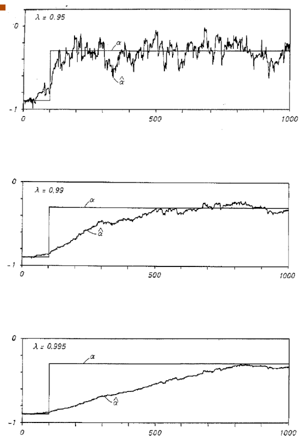
$\lambda < 1$

The information matrix

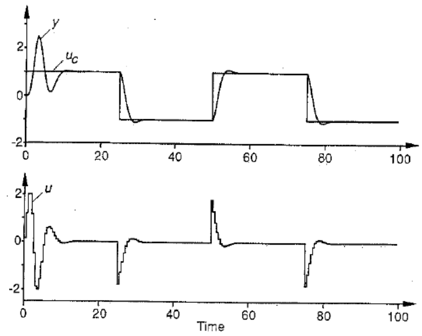
$$P(t)^{-1} = \lambda P(t-1)^{-1} + \varphi(t)\varphi(t)^T$$

$$P(t)^{-1} = P(t-1)^{-1} - (1-\lambda)P(t-1)^{-1} + \varphi(t)\varphi(t)^T$$

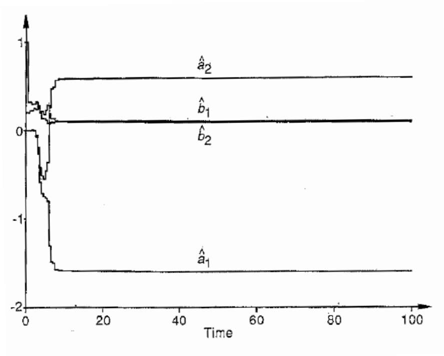
Simulation results



Simulation results



Simulation results



Research topics during the 70'ies

- stability
- convergence
- choice of λ
- colored noise $n(t)$

We believed that this was the final solution to control problems!

Parallell activities in industry around 1980

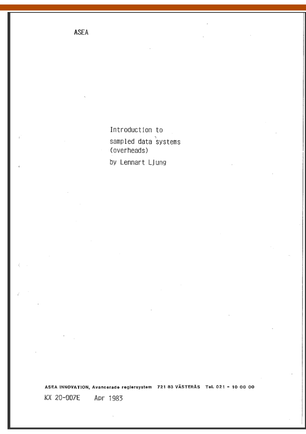
ABB prepared for the Novatune

Foxboro had their own strategy - the rule-based adaptive PID controller.

This was also the time when automatic tuning procedures started to appear

ABB Novatune

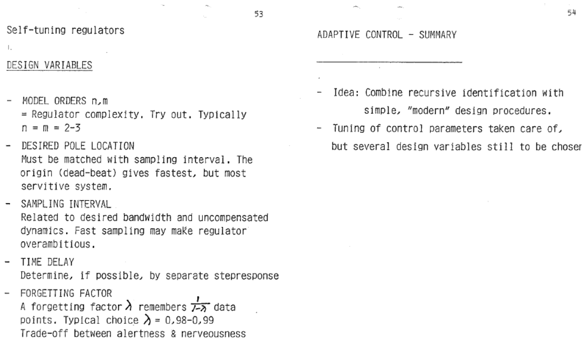
ABB Novatune


ASEA

Introduction to
sampled data systems
(overheads)
by Lennart Ljung

ASEA INNOVATION, Avancerade reglersystem 721 83 VÄSTERÅS Tel. 021 - 10 00 00
KX 20-007E Apr 1983

ABB Novatune



53

Self-tuning regulators

DESIGN VARIABLES

- MODEL ORDERS n,m
  = Regulator complexity. Try out. Typically
  n = m = 2-3
- DESIRED POLE LOCATION
  Must be matched with sampling interval. The origin (dead-beat) gives fastest, but most servitive system.
- SAMPLING INTERVAL
  Related to desired bandwidth and uncompensated dynamics. Fast sampling may make regulator overambitious.
- TIME DELAY
  Determine, if possible, by separate stepresponse
- FORGETTING FACTOR
  A forgetting factor λ remembers $\frac{1}{1-\lambda}$ data points. Typical choice λ = 0,98-0,99
  Trade-off between alertness & nerveousness

54

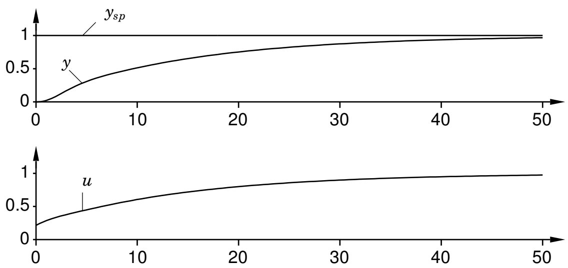
ADAPTIVE CONTROL - SUMMARY

- Idea: Combine recursive identification with simple, "modern" design procedures.
- Tuning of control parameters taken care of, but several design variables still to be choser

Adaptive control – Rule-based methods

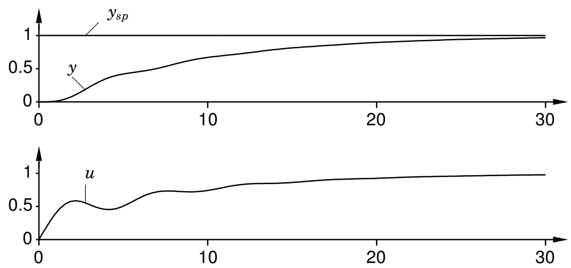
1. Wait for setpoint or load disturbances
2. Note the characteristics of the responses
3. Adjust the controller based on well-known rules
4. Wait for the next response

Adaptive control – Rule-based methods



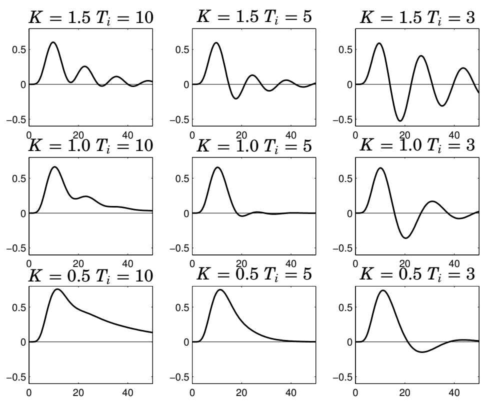
Rule: Increase gain, decrease integral time

Adaptive control – Rule-based methods

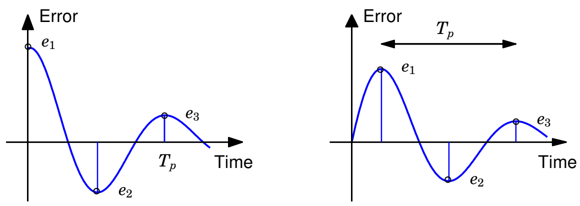


Rule: Decrease gain, decrease integral time

Adaptive control – Rule-based methods



Foxboro EXACT



- Handles load disturbances efficiently
- Assumes isolated step disturbances

Collaboration University – Industry

- 60ies: Good collaboration.
- 70ies: Collaboration interrupted.
- 80ies: Slowly improving.
- 90ies: Good collaboration.

70ies

- Collaboration between process industry and university bad
- Collaboration between suppliers and process industry bad
- Simulators
- $G(s) = B(s)/A(s)$
- Adaptive control is developed in this environment
- "Adaptive control solves all problems"
- "The end of automatic control research?"

80ies

Discovery:

- The process is not $G(s) = B(s)/A(s)$
- Disturbances are not always step changes in set point
- There are load disturbances
- Processes are nonlinear, e.g. friction and hysteresis
- Products must be automatic – no parameters to tune

ABB ECA600

Automatic tuning

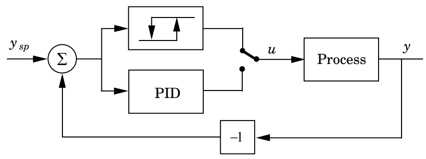
When a controller is to be tuned, the following tasks are performed:

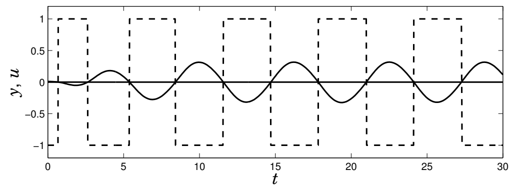
1. Disturb the process.
2. Derive a process model.
3. Determine controller parameters based on the model.

Automatic tuning:

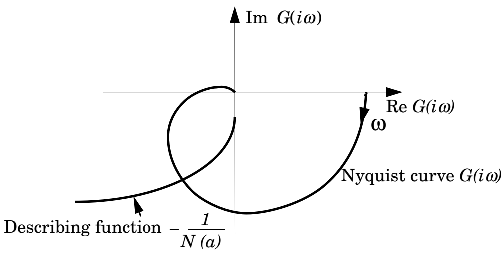
An aid where these tasks are performed automatically.

The relay method





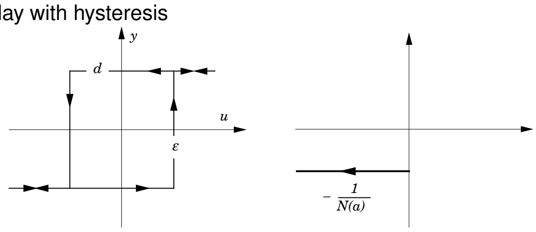
Describing function analysis



$$N(a)G(i\omega) = -1$$

Describing function analysis

Relay with hysteresis



$$-\frac{1}{N(a)} = -\frac{\pi}{4d}\sqrt{a^2 - \epsilon^2} - i\,\frac{\pi\epsilon}{4d}$$

Design ECA600

Normal:

$$P(i\omega_0)C(i\omega_0) = 0.5e^{-i135\pi/180}$$

$$T_i' = 4T_d'$$

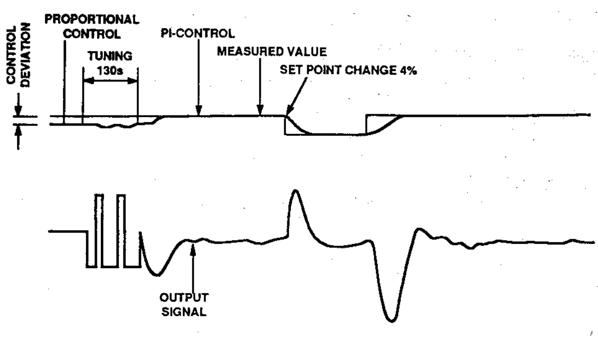
Lag-dominant processes:

$$K = 0.5/|P(i\omega_0)|, \quad T_i = 4/\omega_0$$

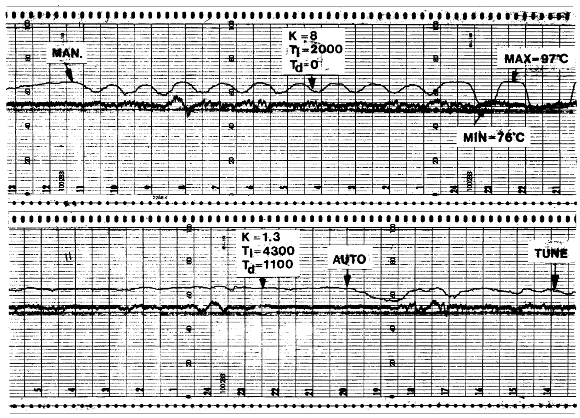
Delay-dominant processes:

$$K = 0.25/|P(i\omega_0)|, \quad T_i = 1.6/\omega_0$$

Example – Level control



Example – Temperature control

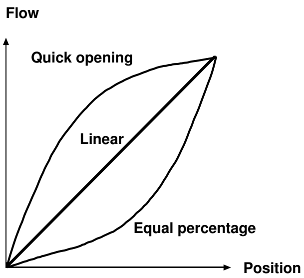


ECA600 Experience

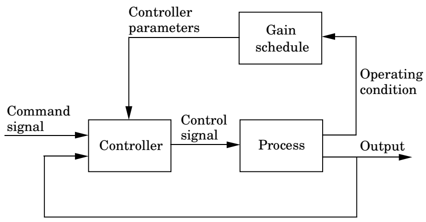
- Why was it well received?
- Must be fully automatic.
- Keep it simple!

Nonlinearities

Example – Nonlinear valve



Gain scheduling



Example – Gain scheduling references

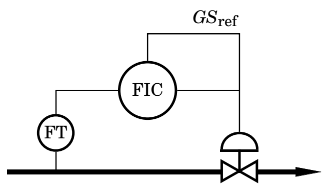
- Production level
- Machine speed
- Speed and hight (air planes)

Gain scheduling

A table with different controller parameter for different operating conditions.

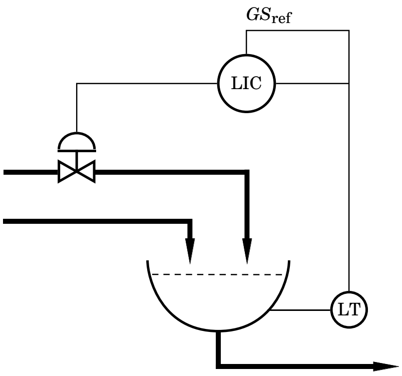
GS_{ref} (%)	0–33	33–67	67–100
K	0.43	0.28	0.20
T_i	1.81	1.89	1.71
T_d	0.45	0.47	0.43

Gain scheduling



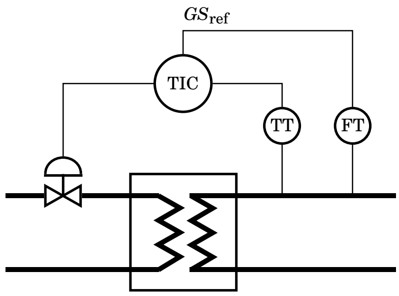
Gain scheduling based on the control signal.

Gain scheduling



Gain scheduling based on the measurement signal.

Gain scheduling

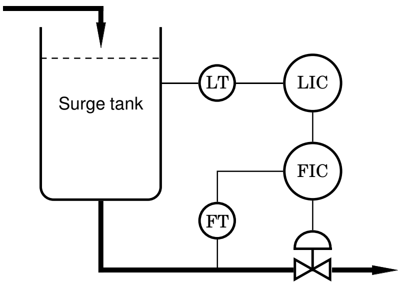


Gain scheduling based on an external signal (the flow).

Gain scheduling

Gain scheduling can also be used to treat linear processes with production-dependent specifications.

Example – Surge tank control:



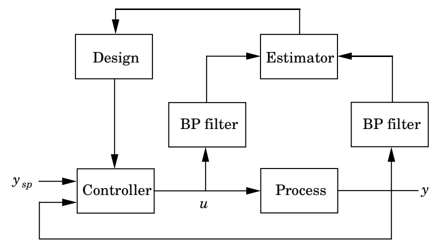
Conclusions

- Gain scheduling is simple and robust
- Important when controllers are tuned for high performance
- Good complement to automatic tuning procedures

Adaptive problems to handle

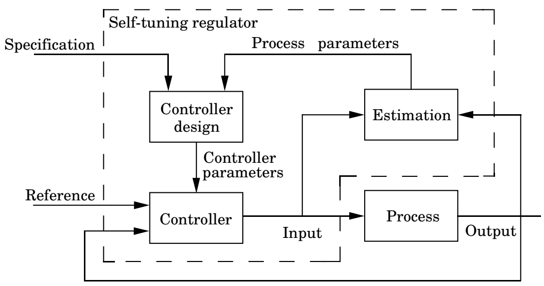
- Initialization (fully automatic)
- Tracking - forgetting - excitation
- Load disturbances
- Stiction in valves
- Mode switches

ABB ECA 600



Idea: Track the point on the Nyquist curve identified by the relay autotuner!

Adaptive controllers



Key problem: The parameter estimation must be based on relevant data.

Recursive least-squares estimation

$$\hat{\theta}(t) = \hat{\theta}(t-1) + P(t)\varphi(t)\epsilon(t)$$

$$\epsilon(t) = y(t) - \varphi(t)^T\hat{\theta}(t-1)$$

$$P(t) = P(t-1) - \frac{P(t-1)\varphi(t)\varphi(t)^T P(t-1)}{1+\varphi(t)^T P(t-1)\varphi(t)}$$

Forgetting:

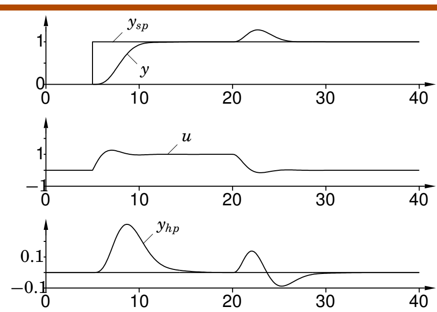
$$P(t) = \frac{1}{\lambda}\left(P(t-1) - \frac{P(t-1)\varphi(t)\varphi(t)^T P(t-1)}{\lambda+\varphi(t)^T P(t-1)\varphi(t)}\right)$$

Excitation

1. Ensure that excitation always is present by adding excitation signals to the process input.
2. Ensure that estimation is performed only when there is enough natural excitation of the process.

The last alternative is preferred.

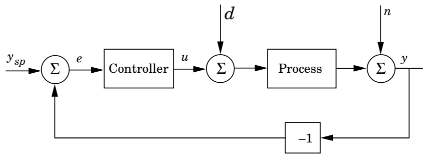
Excitation detection



$$Y_{hp} = \frac{s}{s+\omega_{hp}}Y \qquad \omega_{hp} \sim \frac{1}{T_p}$$

Estimate when $|y_{hp}|$ is large.

Problems with load disturbances



The process output y is given by

$$Y(s) = P(s)\,(U(s) + D(s)) + N(s)$$

Noise no problem, but loads are.

$$y(t) = y_u(t) + y_d(t),$$

Problems with load disturbances

The estimation error is

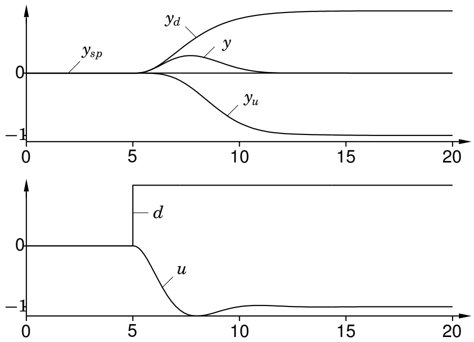
$$\epsilon(t) = y(t) - \hat{y}(t) = y(t) - \varphi(t)^T \hat{\theta}(t-1)$$

where it is assumed that

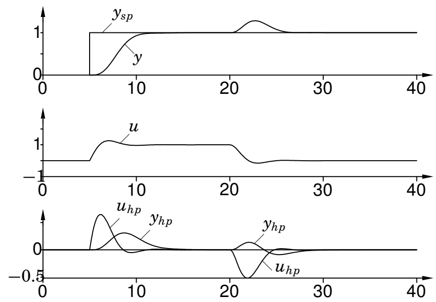
$$y(t) = y_u(t) = \varphi(t)^T \theta(t-1),$$

Problems with load disturbances

Solution: Avoid adaptation when y_d dominates over y_u.



Load disturbance detection



Set-point changes: $|y_{hp}|$ and $|u_{hp}|$ large, move in the same direction

Load disturbances: $|y_{hp}|$ and $|u_{hp}|$ large, move in opposite direction

Oscillation detection

Oscillations near ultimate frequency:

When generated by setpoint:

$y(t) = y_u(t) \Rightarrow$ Good excitation

When generated by friction or load disturbances:

$y(t) = y_d(t) \Rightarrow$ Bad excitation

Detect oscillations and avoid adaptation!

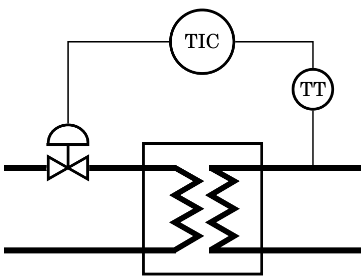
Signal saturation

$$y(t) = y_u(t) + (y_{\text{limit}} - y_u(t)).$$

Avoid adaptation during signal saturation!

Also important to have bumpless transfer between modes.

Example – Heat exchanger



Example – Heat exchanger

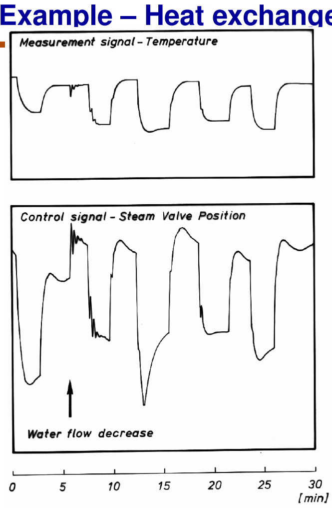


ABB ECA 600 – Adaptive feedforward

Process model:

$$y(t) = au(t-4h) + bv(t-4h),$$

where $h = T_0/8$, T_0 is the oscillation period

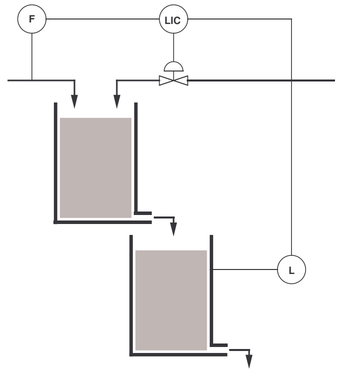
Controller

$$\Delta u_{ff}(t) = k_{ff}(t)\Delta v(t),$$

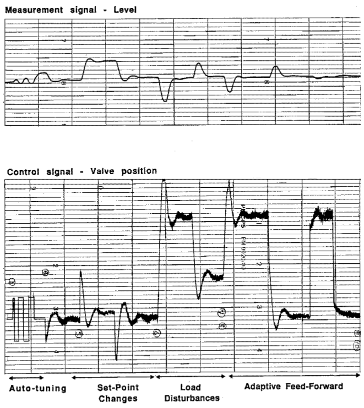
where

$$k_{ff}(t) = -0.8\,\frac{\hat{b}(t)}{\hat{a}(t)}.$$

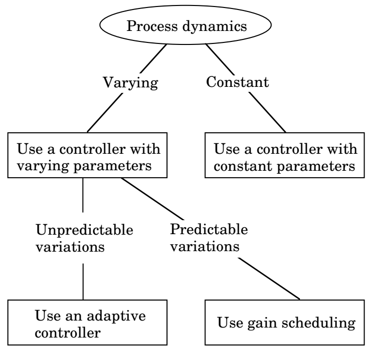
Example – Adaptive feed-forward



Example – Adaptive feed-forward



Use of adaptive techniques



“Revolutionizing” methods

- 80-talet: Adaptive control
- 90-talet: Fuzzy control
- 00-talet: MPC (Model Predictive Control)

hat went wrong with the adaptive control developmer

- Too complicated
  - must be fully automatic
- Too limited
  - could not be tuned manually
- Bad robustness properties
  - load disturbances and nonlinearities

Experiences

- Universities must collaborate with end users
- Universities must collaborate with system suppliers
- System suppliers must collaborate with end users
- End users in process control must keep or improve their competence