FACULTY OF ELECTRICAL ENGINEERING
UNIVERSITY OF WEST BOHEMIA

RICE

Introduction to Model Predictive Control

Vaclav Smidl

Outline

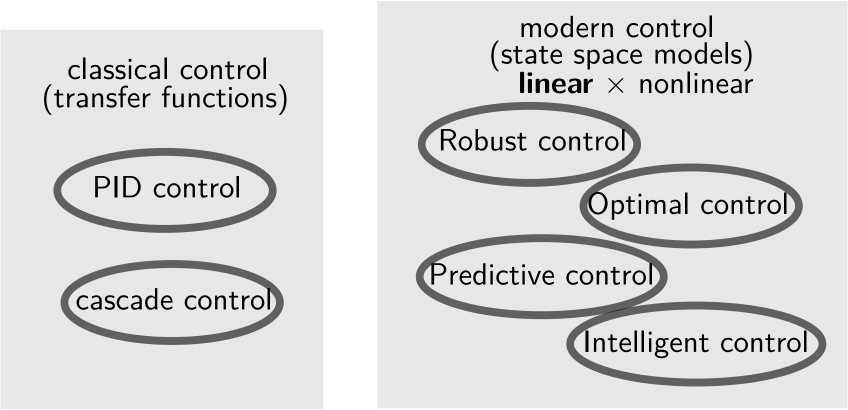
Control theory

- History
- PID control

Model predictive control

- Optimization background

Control Theory Landscape



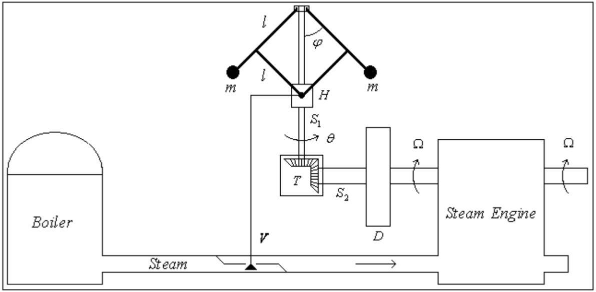
Â

History of control

History of control

Centrifugal Governor:

- Watt & Boulton in 1788
- First popular feedback controller
- **Speed controller**
- Negative feedback
- output too high -> decrease input



PID control

Formalization of the idea of control error suppression:

$$u_t = K_p e_t + K_I \int e_t dt + K_D \frac{de_t}{dt},$$

control gains, K_P, K_I, K_D.

P: Proportional:
- large - instability,
- small - rejection of small errors,

I: Integral:
- removes steady state error,
- causes overshoot,

D: differential
- improves settling time,
- rarely used (20%) - instability

PID control

Formalization of the idea of control error suppression:

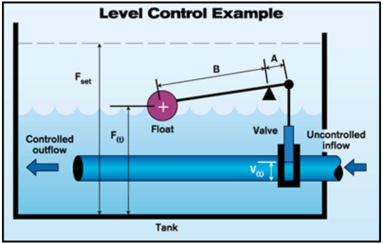
$$u_t = K_p e_t + K_I \int e_t dt + K_D \frac{de_t}{dt},$$

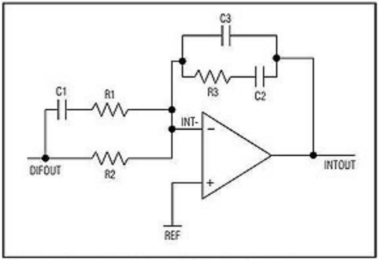
control gains, K_P, K_I, K_D.

P: Proportional:
- large - instability,
- small - rejection of small errors,

I: Integral:
- removes steady state error,
- causes overshoot,

D: differential
- improves settling time,
- rarely used (20%) - instability



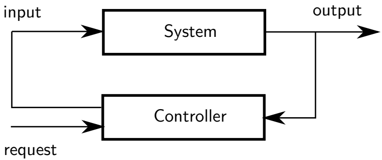


Proportional Integral Derivative Controller

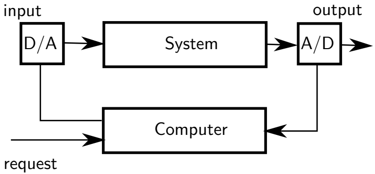
- parameter tuning

Digital control

Analog: direct connection



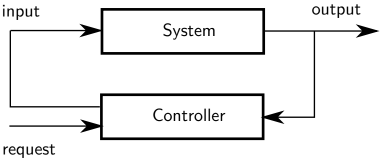
Digital: digital comm.



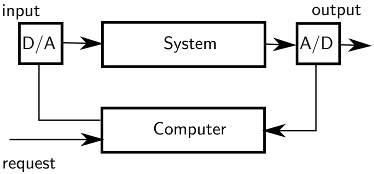
- far more flexible
- can we take advantage of it?

Digital control

Analog: direct connection



Digital: digital comm.



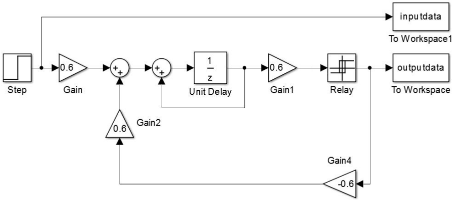
- far more flexible
- can we take advantage of it?

Design change:

- Laplace transfrom -> Z transfrom
- Discrete time controllers:

$$u_t = K_p e_t + K_I \sum_t e_t + K_D(e_t - e_{t-1}),$$

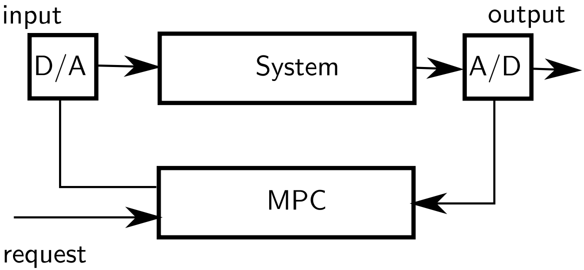
Block diagrams:



Classical analog intuition and thinking.

Model predictive control

Block diagram:



- no z transform,
- no links,
- no intuition?
  - Different language (optimal control theory)
  - Unique solution

Optimization background

- cost function
- minimum, argument minima
- constraint
- finite x continuous control set

Application to control

Keep current i_t of series RL circuit close to the requested value i_t^*.
Closeness is measured by square distance

$$(i_t - i_t^*)^2$$

We can not change the measured value of the current, but the future. The model of RL circuit is:

Application to control

Keep current i_t of series RL circuit close to the requested value i_t^*. Closeness is measured by square distance

$$(i_t - i_t^*)^2$$

We can not change the measured value of the current, but the future. The model of RL circuit is:

$$L\frac{di}{dt} + Ri = u \qquad\qquad i_{t+1} = \left(1 - \frac{\Delta t R}{L}\right) i_t + \frac{\Delta t}{L} u_t$$

simplified to

$$i_{t+1} = a i_t + u_t,$$

Application to control

Keep current i_t of series RL circuit close to the requested value i_t^*. Closeness is measured by square distance

$$(i_t - i_t^*)^2$$

We can not change the measured value of the current, but the future. The model of RL circuit is:

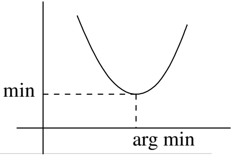
$$L\frac{di}{dt} + Ri = u \qquad i_{t+1} = \left(1 - \frac{\Delta t R}{L}\right) i_t + \frac{\Delta t}{L} u_t$$

simplified to

$$i_{t+1} = a i_t + u_t,$$

Using the finite control set approach:

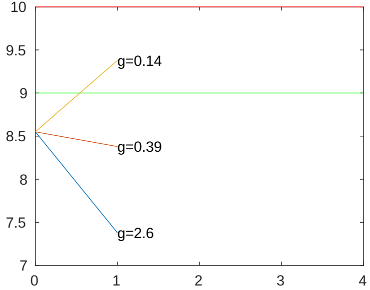
$$\begin{aligned} u_t^* &= \arg\min_{u_t}((i_{t+1} - i^*)^2), \\ s.t.: \quad u_t &\in \{-1, 0, 1\},\ i_{t+1} = a i_t + u_t, \end{aligned}$$



One step ahead MPC

$$i_t = 8.55 \qquad u_t = \arg\min(\underbrace{(i_{t+1} - i^*)^2}_{g})$$

$$i_{t+1} = 0.98 i_t + u_t, \qquad u_t \in \{-1, 0, 1\}$$

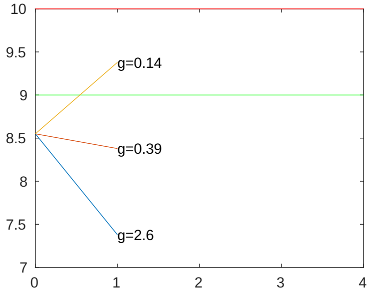


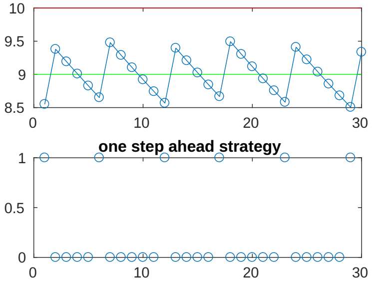
u_0	i_1	g_1
1	9.3	0.14
0	8.3	0.39
-1	7.3	2.6

One step ahead MPC

$$i_t = 8.55 \qquad u_t = \arg\min(\underbrace{(i_{t+1} - i^*)^2}_{g})$$

$$i_{t+1} = 0.98 i_t + u_t, \qquad u_t \in \{-1, 0, 1\}$$





Code for the first MPC

```
clear all
i(1) = 8.55; T = 30; u=-1:1;

for t=1:T
    % model predictive controller
    for j = 1:3
        ip =  0.98*i(t) + u(j);
        g(j) = (ip-9)^2;
    end
    [mi,imi]=min(g);
    U(t) = u(imi);

    % simulation of the real plant
    i(t+1) = 0.98*i(t) + u(imi);
end
```

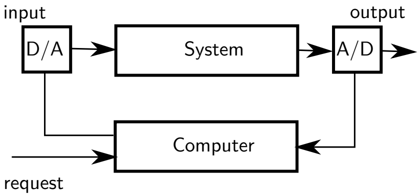
Code for the first MPC

```
clear all
i(1) = 8.55; T = 30; u=-1:1;

for t=1:T
    % model predictive controller
    for j = 1:3
        ip =  0.98*i(t) + u(j);
        g(j) = (ip-9)^2;
    end
    [mi,imi]=min(g);
    U(t) = u(imi);

    % simulation of the real plant
    i(t+1) = 0.98*i(t) + u(imi);
end
```

- Can we save some operations?

How it differs from PWM?

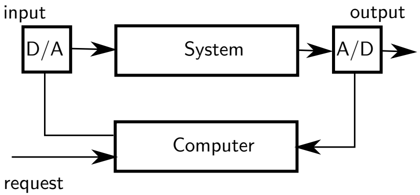
Zero-order hold



1. investigate spectrum of the generated current

How it differs from PWM?

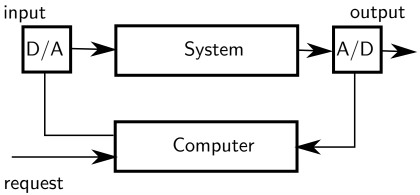
Zero-order hold



1. investigate spectrum of the generated current
2. try to track a sine wave

How it differs from PWM?

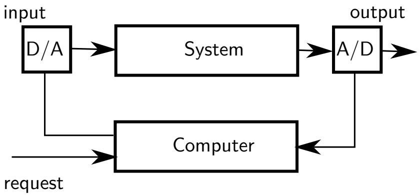
Zero-order hold



1. investigate spectrum of the generated current
2. try to track a sine wave
3. Introduce current limit, $i_t \leq 9.2$

How it differs from PWM?

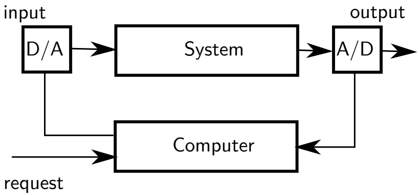
Zero-order hold



1. investigate spectrum of the generated current
2. try to track a sine wave
3. Introduce current limit, $i_t \leq 9.2$
4. change of switching frequency (penalization)

How it differs from PWM?

Zero-order hold



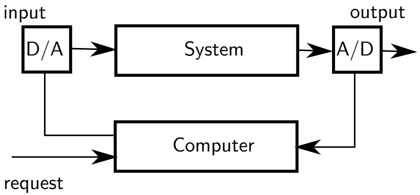
1. investigate spectrum of the generated current
2. try to track a sine wave
3. Introduce current limit, $i_t \leq 9.2$
4. change of switching frequency (penalization)

$$u_t^* = \arg\min_{u_t}((i_{t+1} - i^*)^2 + \alpha|u_t - u_{t-1}|),$$

Any control modification is achieved by change of model or cost.

How it differs from PWM?

Zero-order hold



1. investigate spectrum of the generated current
2. try to track a sine wave
3. Introduce current limit, $i_t \leq 9.2$
4. change of switching frequency (penalization)

$$u_t^* = \arg\min_{u_t}((i_{t+1} - i^*)^2 + \alpha|u_t - u_{t-1}|),$$

Any control modification is achieved by change of model or cost.

- can the cost be $(\omega - \omega^*)^2$?

Practicalities: time delay & delay compensation

- We can not compute the response arbitrarily fast
- When the controller computes action, the real system is still running

Practicalities: time delay & delay compensation

- We can not compute the response arbitrarily fast
- When the controller computes action, the real system is still running
- We need to assume that we compute future control action u_{t+1} that will have effect on the next current i_{t+2}.
  - The state i_{t+1} is computed using the model

    $$i_{t+1} = ai_t + u_t,$$

    where u_t is known from previous step.
  - the optimization is performed for

    $$\begin{aligned} u^*_{t+1} &= \arg\min_{u_{t+1}}((i_{t+2} - i^*)^2), \\ s.t.: \quad u_t &\in \{-1, 0, 1\},\ i_{t+2} = ai_{t+1} + u_{t+1} \end{aligned}$$

Exercise: RL control with delay compensation

```
clear all
i(1) = 8.55; T = 30; u=-1:1;

for t=2:T
    % model predictive controller
    for j = 1:3
        ip =  0.98*i(t-1) + u(j);
        g(j) = (ip-9)^2;
    end
    [mi,imi]=min(g);
    U(t-1) = u(imi);

    % simulation of the real plant
    i(t) = 0.98*i(t-1) + u(imi);
end
```

- Implement with 1-step time delay compensation

Long prediction horizon

- Why one-step-ahead may not be enough?

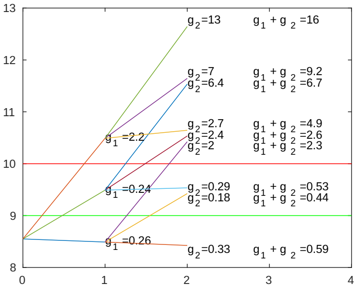
Long prediction horizon

- Why one-step-ahead may not be enough?
  - delay
  - high order dynamics,
- Correct solution for additive cost via dynamic programming
- Computational burden: grows exponentially
- Approximate MPC
  - preselection strategies
  - sphere decoding

Two step ahead MPC

$$i_t = 8.55 \qquad u_t, u_{t+1} = \arg\min((i_{t+1} - i^*)^2 + (i_{t+2} - i^*)^2)$$

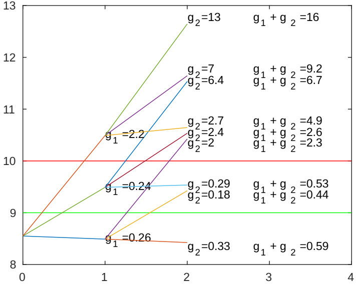
$$i_{t+1} = 1.11 i_t + u_t, \qquad u_t \in \{-1, 0, 1\}, u_{t+1} \in \{-1, 0, 1\}$$

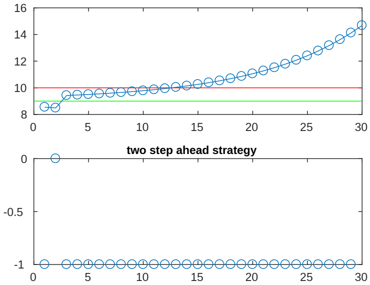


Two step ahead MPC

$$i_t = 8.55 \qquad u_t, u_{t+1} = \arg\min((i_{t+1} - i^*)^2 + (i_{t+2} - i^*)^2)$$

$$i_{t+1} = 1.11 i_t + u_t, \qquad u_t \in \{-1, 0, 1\}, u_{t+1} \in \{-1, 0, 1\}$$





General concept of MPC

Find a control action u^* from the set of admissible values $\mathcal{U}$ minimizing cost

$$
\begin{aligned}
u^*_{t:t+h} &= \arg\min g(y^*, x_t) \\
\text{s.t. } x_{t+1} &= f(x_t, u_t), \\
& u_\tau \in \mathcal{U},\ x_\tau \in \mathcal{X},\ \forall \tau = t, \ldots t+h
\end{aligned}
$$

where:

- $\mathcal{U}$ can be a finite set of states (FCS) or infinite (continuous) set,
- $\mathcal{X}$ is a set of admissible states
- h is the prediction horizon.

MPC of current loop of IM drive

Model in flux-linked d-q frame ($\Psi_{rq} = 0$)

$$\frac{di_{sd}}{dt} = -\alpha i_{sd} + \omega_{s\Psi} i_{sq} + \beta \Psi_{rd} + \delta u_{sd},$$
$$\frac{di_{sq}}{dt} = -\omega_{s\Psi} i_{sd} - \mu i_{sd} - \eta \omega_{s\Psi} \Psi_{rd} + \delta u_{sq},$$
$$\frac{d\Psi_{rd}}{dt} = R_r \frac{L_m}{L_r} i_{sd} - \frac{R_r}{L_r} \Psi_{rd},$$

with parameters $\lambda = L_{s\sigma} + L_{r\sigma}\frac{L_m}{L_r}$, $\alpha = \frac{R_s + R_r\left(L_m^2/L_r^2\right)}{\lambda}$, $\beta = \frac{R_r\left(L_m/L_r^2\right)}{\lambda}$, $\delta = \frac{1}{\lambda}$, $\mu = \frac{R_s}{\lambda}$, $\eta = \frac{L_m/L_r}{\lambda}$, $L_r = L_m + L_{r\sigma}$ are transformations of physical parameters of the drive (stator resistance R_s, stator leakage inductance $L_{s\sigma}$, magnetizing inductance L_m, rotor inductance L_r, rotor leakage inductance $L_{r\sigma}$, and rotor resistance R_r), and $\omega_{s\Psi}$ is angular speed of the rotor magnetic flux vector Ψ_r with respect to stator coordinates.

MPC of current loop of IM drive

Position of the rotor flux ϑ_r is computed using

$$\vartheta_\mathrm{r} = p_\mathrm{p}\vartheta_\mathrm{m} + \int_t \frac{L_\mathrm{m}}{T_\mathrm{r}} \frac{i_\mathrm{sq}}{\Psi_\mathrm{rd}} dt, \tag{1}$$

where rotor time constant $T_\mathrm{r} = L_\mathrm{r}/R_\mathrm{r}$, p_p is the number of pole pairs, and ϑ_m is the mechanical position of the rotor.

MPC of current loop of IM drive

Cost function

$$g_{is}(x_{\text{pred}}, S_i) = \left| i_{\text{sd}}^* - \hat{i}_{\text{sd},t+2}(S_i) \right|^2 + \left| i_{\text{sq}}^* - \hat{i}_{\text{sq},t+2}(S_i) \right|^2, \tag{2}$$

where $S_i \in \mathcal{S} = \{++, ++-, +-+, -++, +--, -+-, --+, ---\}$, is the switching combination.

- define input voltage vectors $u_{sd}(S_i) =$
- predict currents i_{t+1}
- update state (flux magnitude and angle)