Bottom-Up Architectures

Bo Bernhardsson and K. J. Åström

Department of Automatic Control LTH, Lund University
Introduction

Basic Architectures

Large Parameter Variations

Otto J. M. Smith’s Specials

Miscellaneous

Soft Computing

Summary

Theme: Brick by brick.
Design

- Design is a part of all engineering disciplines: A, ME, EE, CE, CS, AA, ...

- Building complex systems from standard parts have been a standard procedure. Nuts, bolts standard assemblies. Transistors, boards, cabinets. VLSI, graphs, design rules, libraries. Subroutines, programs, component software.

- Many attempts to make design theories and design methods - not too successful

- CS has been an interesting proving ground because it is easy to experiment, but also easy to include realistic settings - Abelson Sussman Structure and Interpretation of Computer Programs

- Chip design is the role model
  - Abstractions, Layering, Design rules, Testing

- Can we imitate it?
Views on Control System Design

Holistic
- Requirements
- Specifications
- Modeling
- Analysis
- Simulation
- Design
- Implementation
- Commissioning
- Operation
- Upgrading

Reductionistic
- Stability
- Robustness
- Passivity
- Optimization
Build a system by combining a collection of building blocks. Key ingredients are building blocks and combination principles.

- What are the components (blocks, algorithms,...)?
- What are the rules for combining components? - Design principles.

Building blocks
- Linear Systems
- Controllers
- Estimators
- Nonlinearities
  - Limiters
  - Selectors
  - Logic
- Estimators
- Optimizers

System principles
- Feedback
- Feedforward
- Cascade
- Midranging
- Selector control
- Model following
- Gain scheduling
- Adaptation
Bottom-Up Architectures

1. Introduction
2. Basic Architectures
   - Feedback/Feedforward
   - Generalized PI Control
   - Cascade Control
   - Midranging
   - Selector Control
3. Large Parameter Variations
4. Otto J. M. Smith’s Specials
5. Miscellaneous
6. Soft Computing
7. Summary

Theme: Brick by brick.
Feedback and Feedforward

- Reduce effects of measured disturbances and improve command signal response
- Feedback and feedforward have nice complementary properties
- Feedback does not require accurate models (sensitivity can be less than 1), feedforward does (sensitivity is always one)
- Feedback can lead to instability, feedforward can never destabilize
- Windup protection
A nice separation of the different functions

The signals $x_m$ and $u_{ff}$ can be generated from $r$ in real time or from stored tables (robotics)

Integral action and windup protection
Improved Command Signal Response

- Essentially a problem of computing inverses or approximate inverses of systems $PM_u = M_y$ with constraints
- Make reasonable demands, time delays and RHP zeros of $P$ must be included in $M_y$
- The feedforward parts $M_u$ and $M_y$ can be nonlinear. Modelica can deliver inverse models

PI control with $M_s = 1.4$ (dashed) and $M_s = 2.0$ (dotted) for $P(s) = 1/(s + 1)^4$ and non-linear feedforward (solid)
Feedforward - Measured Disturbance

Basic scheme for reducing effects of measured load disturbance

\[ G_{ff} = P_u^{-1} P_d = P_u^t P_d \]

\[ G_{yd} = \frac{P_d - P_u G_{ff}}{1 + P_u C} \]

TAT: Can you see some drawbacks if perfect cancellation is not possible?
Only feedback dashed line FB+FF red full line dotted Brosilo's version
A 2DOF scheme where controller is told what the feedforward is doing

\[ G_{yd} = \frac{P_d - P_u G_{ff}}{1 + P_u C}, \quad H = P_d - P_u G_{ff}, \quad G_{yd} = P_d - P_u G_{ff} \]
An Example - Drum Level Control

The shrink and swell effect
System inversion is to compute the input that gives the output for a given system.

The concept of flat output

A flat output $y$ is an output signal such that the state $x$ and the input $u$ can be generated from $y$ and its derivatives.

$$\frac{d^n y}{dt^n} + a_1 \frac{d^{n-1} y}{dt^{n-1}} + \cdots + a_n y = b_n u$$

Useful for feedforward design by making the flat output behave in the desired way.

Can be applied to nonlinear systems.
Generalized Integral Control

- Integral control was a real breakthrough
  Maxwell (1868) mentioned that Siemens (1866) had distinguished between governors (PI) and moderators (P)
- Automatic removal of steady state errors (automatic reset)
- Integral control eliminates a disturbance that is constant with unknown amplitude
- Can it be generalized to other types of disturbances?
  - Ramps \( v(t) = a_0 + a_1 t \) where \( a_0, a_1 \) are unknown
  - Jerks \( v(t) = a_0 + a_1 t + a_2 t^2 \) where \( a_0, a_1, a_2 \) are unknown
  - Sinusoidal disturbances with known frequency but unknown amplitude and phase
  - Sinusoidal disturbances with unknown frequency, amplitude and phase
  - Periodic disturbances with known period
- Idea: Build a model of the disturbance in the controller!
Generalized Integral Control

- Constant but unknown
- Ramps with unknown levels and rates
- Sinusoidal with known frequency but unknown amplitude and phase
- Periodic with known period but unknown shape

\[ G_f(s) = \frac{1}{1 + sT_f} \]
\[ G_f(s) = \frac{2\zeta\omega_0 s}{s^2 + 2\zeta\omega_0 s + \omega_0^2} \]
\[ G_f(s) = e^{-sT} \]

\[ C(s) = \frac{k}{1 - G_f(s)} \]
\[ C_{\text{const}}(s) = 1 + \frac{1}{sT} \]
\[ C_{\text{sine}}(s) = \frac{s^2 + 2\zeta\omega_0 s + \omega_0^2}{s^2 + \omega_0^2} \]
\[ C_{\text{periodic}}(s) = \frac{1}{1 - e^{-sT}} \]
Elimination of Periodic Disturbances

\[ G_f(s) = e^{-sT}, \quad C_{\text{periodic}}(s) = \frac{1}{1 - e^{-sT}} \]

Control law

\[ u(t) = e(t) + u(t - T) \]

Transfer function from disturbance to output

\[ G_{yd}(s) = \frac{P(s)}{1 + P(s)C(s)} = \frac{P(s)(1 - e^{-sT})}{1 - e^{-sT} + P(s)}. \]

The relation between load disturbance and output

\[ (1 - e^{-sT} + P(s))Y(s) = P(s)(1 - e^{-sT})D(s). \]

Notice that the time function corresponding to \((1 - e^{-sT})D(s)\) is \(d(t) - d(t - T)\), which is zero if \(d\) has period \(T\). Compare with PI control.
No difficulties with infinite gain for a PI controller
Difficulties with controller that have infinite gain at high frequencies
The remedy is to lower the gain and introduce high frequency roll-off

Replacing $G_f$ by $\alpha G_f$ gives $C = \frac{1}{1-\alpha G_f}$, with $\alpha$ close to 1

$C_{\text{const}}(s) = \frac{1 + sT}{1 - \alpha + sT}$

$C_{\text{sine}}(s) = \frac{s^2 + 2\zeta \omega_0 s + \omega_0^2}{s^2 + 2(1 - \alpha)\zeta \omega_0 s + \omega_0^2}$

$C_{\text{per}}(s) = \frac{1}{1 - \alpha e^{-sT}}$
Bode Plots for $\alpha = 0.99$
Cascade Control - Several Sensors

- Using several sensors - state feedback is the ultimate
- State feedback ultimate case
- Tight feedback around disturbances and uncertainty
- Linearize a nonlinear actuator
- Integral action and windup
When is Cascade Control Useful?

Bo Bernhardsson and K. J. Åström  
Bottom-Up Architectures
Cascade Control - Example

Process dynamics

\[ P_1(s) = \frac{1}{s + 1} \quad P_2(s) = \frac{1}{(s + 1)^3} \quad P(s) = P_1P_2(s) = \frac{1}{(s + 1)^4} \]

- PI Controller outer loop \( K = 0.37, T_i = 2.2 \)
- P Controller inner loop \( K = 5 \), PI outer \( K = 0.55, T_i = 1.9 \)
Examples of Cascade Control

Motordrive

Three cascaded loops
- Current loop
- Velocity loop
- Position loop
Output temperature is the primary variable
Three-way valve
Flow measurement is used to mix primary water
Use parallel actuators to obtain high actuation precision and wide actuation range.

Fine actuation through $v_1$, course actuation through $v_2$.

Try to keep the valve $v_1$ in the mid range.

Course actuation can also be discrete (chillers).

Separate time scales.
The optical package is light with a voice coil drive.

The sledge drive is slower and provides the coarse motion.
$P_1$ provides precise control of $y$ but the range of $u_1$ is limited.

$P_2$ attempts to control the output $y$ so that the control signal $u_1$ is in mid range.

Windup protection
Similar to basic scheme but with coupling that tells first loop what second loop is doing

Windup protection
Duality

Cascade Control

Midrange Control

Controller

Controller
Split Range Control

Using one controller for two actuators, typically heating and cooling.

Nonlinearity can be asymmetric
Selector Control

- Control with constraints
- Mixing objectives
- Elegant way to handle logic
- Other selectors: median 2 of 3
  - Windup protection via tracking input!!
- Stability analysis

Bo Bernhardsson and K. J. Åström
Bottom-Up Architectures
Air-fuel Controller

Make sure the boiler always runs with excess air
Discuss increase and decrease in power demand
Compare with use of logic
Windup protection
Air-fuel Controller - Always air excess
A common problem is to mix flows in given proportions. Ratio controllers is one way to do this selector control is an alternative (see Air-Fuel control later)

$$r_2 = ay_1$$  \hspace{1cm}  $$r_2(t) = a (\gamma r_1(t) + (1 - \gamma)y_1(t))$$
Top curves ratio control bottom curves blend station
Introduction

Basic Architectures

Large Parameter Variations
  Gain Scheduling
  Recursive Least Squares Estimation
  Model Reference Control
  The Self Tuning Regulator

Otto J. M. Smith’s Specials

Miscellaneous

Soft Computing

Summary

Theme: Brick by brick.
Parameter Variations

- Robust control
  Find a control law that is insensitive to parameter variations
- Gain scheduling
  Measure variable that is well correlated with the parameter variations and change controller parameters
- Adaptive control
  Design a controller that can adapt to parameter variations
  - Many different schemes
    - Model reference adaptive control
    - The self-tuning regulator
    - $L_1$ adaptive control (later in LCCC)
- Dual control
  Control should be directing as well as investigating!
Gain Scheduling

Example of scheduling variables

- Production rate
- Machine speed
- Mach number and dynamic pressure
- Room occupancy
Model Reference Adaptive Control

The MIT rule

- Idea of model following
- MIT rule: $\frac{d\theta}{dt} = -\gamma e \frac{\partial e}{\partial \theta}$
- Many other rules
Recursive Least Squares

\[ y_{t+1} = -a_1 y_t - a_2 y_{t-1} + \cdots + b_1 u_t + \cdots + e_{t+1} \]

\[ = \phi_t^T \theta + e_{t+1} \]

\[ \hat{\theta}_t = \hat{\theta}_{t-1} + K_t (y_t - \phi_t \hat{\theta}_{t-1}) \]

\[ K_t = P_{t-1} \phi_t (\lambda + \phi_t^T P_{t-1} \phi_t)^{-1} = P_t \phi_t \]

- Many versions: directional forgetting, resetting, ...
- Square-root filtering (good numerics!)
The Self-Tuning Regulator (STR)

Certainty equivalence

Many estimation and control design methods
Minimum Variance Control and the STR

\[ y_{t+1} + ay_t = bu_t + e_{t+1} + ce_t \]

\[ u_t = \frac{1}{b}(ay_t - ce_t) = \frac{a - c}{b}y_t \]

In general \( A(q)ty = B(q)u_t + C(q)e_t \)

\[ u_t = \frac{a(q) - c(q)}{b(q)}(y_t - r_t) \]

The self-tuning controller

\[ y_t = \beta(u_t - \theta y_t) + \epsilon \]

\[ u_t = \text{sat}\left(\hat{\theta}(y_t - r_t)\right) \]

Estimate \( \theta \) by least squares for fixed \( \beta \), \( 0.5 < \beta/b < \infty \), \( B(z) \) stable + order conditions. Local stability: real part of \( C(z) \) positive for all zeros of \( B(z) \)
Ship Steering - Performance

Adaptive

PID

Heading

Rudder
Steermaster

NORTHROP GRUMMAN
Using Different Methods

Process dynamics

- Varying
  - Use a controller with varying parameters
    - Unpredictable variations
      - Use an adaptive controller
    - Predictable variations
      - Use gain scheduling
- Constant
  - Use a controller with constant parameters

Bo Bernhardsson and K. J. Åström Bottom-Up Architectures
Extremal Seeking - Optimization

- Draper-Lee Optimize jet engine performance
- Bacteria searching for light or food
- Optimize production rate or quality
- Many perturbation and optimization techniques can be used

Perturb input, correlate with output
Introduction

Basic Architectures

Large Parameter Variations

Otto J. M. Smith’s Specials

The Smith Predictor
Tore’s PPI Controller
$C_{\text{pred}}$ as a lead compensator
Posicast Control

Miscellaneous

Soft Computing

Summary

Theme: Brick by brick.
The Smith Predictor 1958

- O. J. M. Smith UC Berkeley 1958
- Idea: Use model to create output without delay
- \( P(s) = P_0(s)e^{-sT} \)

Controller and closed loop transfer function

\[
C(s) = \frac{C_0}{1 + C_0 \hat{P}_0 (1 - e^{-sL})} \quad T = \frac{P_0 C_0}{1 + P_0 C_0 e^{-sL}}
\]

TAT: When can you expect trouble?
https://www.youtube.com/watch?v=1zKLakF9dwg start 7:30
Process $P(s) = e^{-sL}/(s + 1)$, $L = 1$ and $8$

PI controller designed for $\omega_c = 2$ and $\zeta_c = 0.7$
Origin of Phase Advance

\[ C = \frac{C_0}{1 + C_0 \bar{P}_0 (1 - e^{-sL})} = C_0 C_{\text{pred}}, \quad C_{\text{pred}} = \frac{1}{1 + C_0 \bar{P}_0 (1 - e^{-sL})} \]

Near the crossover frequency \( C_0 P_0 \approx -1 \) and \( C_{\text{pred}} \approx e^{sL} \)

Bode plot of \( C_{\text{pred}}(s) \) and \( e^{sL} \) for \( L = 8 \)
Unstable oscillatory modes can give phase advance!
Nyquist Plot of Loop Transfer Function

$L = 1$

$L = 2$

$L = 4$

$L = 8$
Smith predictor of loop transfer function for $L = 2$ (full) and for an increase of the time delay by 30%.

The delay margin is the percentage increase of the time delay that makes the system unstable. For $L = 2$ plots to the left the delay margin is 27%. Notice that it is the second peak that creates instability. The delay margin is only 7% for $L = 8$. 
PI and Smith Predictor for Pure Delay

PI control full lines

\[ G = e^{-sL} \]
\[ k = 0.25 \]

Smith predictor dashed lines

\[ G_p G_c = \frac{5}{sL} \]
\[ \frac{T_{cl}}{L} = 0.2 \]
Design a PI controller for FOTD process by canceling the process pole and choosing the gain to give a closed loop time constant $T_{cl}$. The controller then becomes

$$C(s) = \frac{1 + sT}{K_p sT_{cl}} \frac{1}{1 + \frac{1}{sT_{cl}} (1 - e^{-sL})}$$

$$U(s) = \left( k_p + \frac{k_i}{s} \right) E(s) - \frac{1}{sT_{cl}} (1 - e^{-sL}) U(s)$$

Predicts by accounting for past control actions that have not yet shown up in the output. Better than to predict by derivatives of the output! Particularly simple for $T_{cl} = T = T_i$ (Foxboros version)
Phase Advance of PPI Predictor

\[ C_{\text{pred}}(s) = \frac{1}{1 + \frac{1}{sT}(1 - e^{-sL})} \approx \frac{T}{T + L - sL^2/2} \]

\[ \approx \frac{T}{T + L} (1 + sT_{\text{pred}}), \quad T_{\text{pred}} = \frac{1}{2} \frac{L^2}{T + L} \]

\( C_{\text{pred}} \) is a nice phase advance network, better phase advance than with derivative action (dotted)!
Moving a Hanging Container - Posicast Control

O. J. M. Smith Posicast control of damped oscillatory systems, Proc. IRE. (45) 1957, 1249-1255

Has been used successfully for cranes and micro systems

What is the transfer function?
Transfer Function

Step response

Transfer function for posicast control

\[ G_{ff}(s) = \frac{1}{2} \left( 1 + e^{-sT} \right). \]

Sinusoidal signals of frequencies \( \omega = \omega_0, 3\omega_0, 5\omega_0 \ldots \), where \( \omega_0 = \frac{2\pi}{T} \).

Nonrational transfer function

Easy to implement posicast control using digital control.
Introduction

Basic Architectures

Large Parameter Variations

Otto J. M. Smith’s Specials

Miscellaneous

- Internal Model Control IMC 3
- Complementary Filtering 5
- Ratio Control 2
- Linearization by Feedback and Jitter 2
- Limiters 8

Soft Computing

Summary

Theme: Brick by brick.
The signal $v$ does not depend on the control actions.

The signal $v$ represents an output equivalence of all disturbances acting on the process.

Disturbance attenuation is done by design of $Q$.

TAT: What is the loop transfer function?

Ideal response if $Q = P^{-1}$, or approximately $Q = P^\dagger$.

TAT: $P$ must be stable! Why? Modifications?
A Pure Delay Process

\[ P(s) = e^{-sL} \quad Q(s) = 1 \quad T(s) = e^{-sL} \quad L(s) = \frac{e^{-sL}}{1 - e^{-sL}} \]

Frequency response of loop transfer function

\[ L(i\omega) = -\frac{1}{2} - i\frac{\sin \omega L}{2(1 - \cos \omega L)} = -\frac{1}{2} - i\frac{1}{\tan(\omega L/2)} \]

The Gang of Four

\[ S(s) = 1 - e^{-sL} \quad P S(s) = e^{-sL}(1 - e^{-sL}) \]
\[ C S(s) = e^{-sL} \quad T(s) = e^{-sL} \]

- Nyquist plot (Discuss!)
- Gain margin \( g_m = 2 \) phase margin \( \phi_m = 60^\circ \)
- Maximum sensitivities \( M_s = 2 \) and \( M_t = 1 \)
- Looks OK BUT!!!
Delay Margin

Using standard criteria the system looks robust, but what about parametric changes. Assume that the time delay of the process changes from $L$ to $L + \delta$.

$$e^{-s(L+\delta)} = e^{-sL} e^{-s\delta L} = e^{-sL} + e^{-sL}(e^{-s\delta L} - 1)$$

$$\Delta P(s) = e^{-sL}(e^{-s\delta L} - 1)$$

Hence $|\Delta P(i\omega)| = |e^{-i\omega\delta L} - 1|$. The stability criterion

$$\frac{|\Delta P(i\omega)|}{|P(i\omega)|} = |e^{-i\omega\delta L} - 1| < \frac{1}{|T(i\omega)|} = 1.$$

is not satisfied for any $\delta L > 0$ because the left-hand side is 2 for some $\omega$ and the right hand side is 1. Hence unstable for arbitrary small perturbation in the time delay. Sketch Nyquist plot.
Complementary Filtering

- A technique to combine information from several sensors
- A precursor to Kalman filtering
- Useful in its own right
- Requires only models of sensor systems

Signal model ($y_1$ slow but accurate, $y_2$ fast but drifting)

$$y_1 = x + n_1, \quad y_2 = x + n_2$$

Filter for recovering the variable $x$

$$X_f(s) = \frac{1}{s+1}Y_1(s) + \frac{s}{s+1}Y_2(s)$$

Choose $G_1$ as low pass filter, $G_2$ then becomes high pass.
A Kalman Filter Solution

Model the measured value $x_1$ and the drift of the second sensor as unknown constants

\[
y_1 = x_1 + n_1, \quad y_2 = x_1 + x_2 + n_2, \quad \dot{x}_1 = 0, \quad \dot{x}_2 = 0
\]

The Kalman filter

\[
\frac{d}{dt} \begin{pmatrix} \hat{x}_1 \\ \hat{x}_2 \end{pmatrix} = \begin{pmatrix} k_{11}(y_1 - \hat{x}_1) + k_{12}(y_2 - \hat{x}_1 - \hat{x}_2) \\ k_{21}(y_1 - \hat{x}_1) + k_{22}(y_2 - \hat{x}_1 - \hat{x}_2) \end{pmatrix}
\]

After some calculations

\[
\hat{X}_1(s) = \frac{k_{11}s + k_{11}k_{22} - k_{12}k_{21}}{s^2 + (k_{11} + k_{22} + k_{12}) + k_{11}k_{22} - k_{12}k_{21}} Y_1(s)
\]

\[
+ \frac{k_{12}s}{s^2 + (k_{11} + k_{22} + k_{12}) + k_{11}k_{22} - k_{12}k_{21}} Y_1(s)
\]
Velocity and Acceleration Measurements

Determine estimates of velocity and acceleration from measurements of the same quantities.

\[
V_f(s) = \frac{s}{s + k} V(s) + \frac{k}{s + k} V(s)
\]

\[
V_f(s) = \frac{1}{s + k} A(s) + \frac{k}{s + k} V(s)
\]

\[
A_f(s) = \frac{s}{s + k} A(s) + \frac{ks}{s + k} V(s)
\]

Use of an accelerometer and a rate gyro to determine tilt for the Segway is a similar problem.
I tidigare projekt hade man ju stött på behovet av filter, speciellt för att ta hand om brusiga radarsignaler. Man upptäckte då att tex antennvinklarna från egen flygradar mot ett radarföljt mål på grund av målets och det egna flygplanets fart, accleration och rotation vairerade starkt på grund av grundläggande kinematiska samband. ... Jag lärde mig ju snart att inse att dessa praktiska åtgärder helt enkelt bottnade i att man måste tvinga sina filtrerade variabler att satisfiera en modell för sambanden mellan accelerationer, farter och positioner hos eget flygplan och mål. Dessa modellsamband sattes då upp i vektorform varvid det oftast visade sig praktiskt att arbeta i olika koordinatsystem som oftast roterade. ... På detta sätt uppstod vad jag då efter viss vånda valde att kalla “komplementära filter”.

Complementary filtering is a well established field
Complementary Filters or Observers

Both

- Generate estimates of signals that are not measured directly
- Unify information from different sensors (sensor fusion)
- Can be optimized if noise information is available

Complementary filters

- Require models of sensor systems only not process dynamics

Observers

- Require models of process dynamics that typically involves command signals.
- Process inputs provide phase lead.
Both sensors and actuators can be linearized in open loop by feedforward. Feedback can also be used effectively when sensors are available.

Process model: \( y = f(u) \)
Feedforward: \( u = f^{-1}(u_c) \)
Hence: \( y = f(f^{-1}(u_c)) = u_c \)
Requires model
Sensitivity = 1

Requires sensor
Less sensitive
Linearization by Using Jitter Signals

When a triangular jitter signal is added to the error signal the average relay output is

$$\frac{T_p}{4} \left(1 + 2\frac{e}{a}\right) - \frac{T_p}{4} \left(1 - 2\frac{e}{a}\right) = \frac{e}{a}$$

The combination of a relay with a jitter signal thus acts like a saturated linearity.
Limiters

Limiters are used to avoid windup and to limit levels and rates for command signals (never ask the system to do more than it can). Kurt Nicolin Asea (legendary Swedish industrialist): “To add more workorders to an overloaded production unit increases confusion but not productivity.”

- Avoid actuator saturation
- Match demand to process capabilities
- Windup protection

A simple limiter
Simple Rate Limiter

Notice that it creates phase lag

The JAS Gripen problem show video
Jump and Rate Limiters

Less phase lag than with rate limiter

Bo Bernhardsson and K. J. Åström Bottom-Up Architectures
Rate limiters give phase lag, JAS Gripen
Jump-and-rate limiters are commonly used in power systems
Lars Rundqwist’s JAS Gripen Fix

- Assignment of authority for manual and automatic control
- Rate saturation in hydraulic servos causes phase lag
- Commissioning of flight control systems
- Rundqwist’s Rlim inspired by windup protection
Bengt Sjöberg Strikes Again

- Phone calls in the night
- Rlim can be improved!
- Project opportunities
Theme: Brick by brick.
Soft Computing

- Technical and biological systems
- Cybernetics - Control and communication in the animal and the machine. Wiener 1948, Ashby 1956
- Neural systems and the Perceptron (Neural Network)
  - McCulloch Pitts 1943
  - Rosenblatt 1958, Widrow-Hoff 1961 (Addaline)
  - Collapse due to Minsky and Papert 69
  - Survivors: Anderson, Grossberg, Kohonen
- Emergence of Artificial Intelligence
  - Dartmouth Conference 1956, Minsky,
- Revival of Neural Networks
  - Hopfield 1982 and The Snowbird Conference
  - The parallel distributed process group
- Neuro Fuzzy - Zadeh and Japan
- IBM Watson (Kasparov 1997, Jeopardy 2010)
- Autonomous Cars
- Machine Learning
Artificial neuron: \( y(t) = f\left( \sum a_i u_i(t) \right) \)

Kolmogorov’s Theorem: There exist fixed continuous increasing functions \( \phi_{ij}(x) \) so that any continuous function \( f \in \mathbb{R}^n \) can be written in the form

\[
f(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} g_i \left( \sum_{j=1}^{n} \phi_{ij}(x_j) \right)
\]

A function of many variables \( f : \mathbb{R}^n \rightarrow \mathbb{R}^n \) can be represented as a combination of MISO functions \( f : \mathbb{R}^n \rightarrow \mathbb{R} \)
Neural Networks

A nonlinear function with a learning mechanism!
Multilayer Networks

- Feedforward nets
- Nets with feedback (Kohonen)
- Nets with feedback and dynamics (Boltzmann)
Training Neural Networks

Both functions and inverse functions can be generated
Assume that all states can be measured or estimated using simple estimators.

The control law is then a nonlinear function \( f : \mathbb{R}^x \rightarrow \mathbb{R}^u \).

Difficult to represent functions of many variables make neural networks and fuzzy attractive.

Quantize the states \( \mathcal{N}^n \) large for large \( n \).

Use soft computing to construct or learn the control law.

Typical example: swinging up a pendulum or balancing a pole.

\[
\begin{align*}
    u(x_1, x_2) &= 2a \sin x_1 + bx_2 F(x_1, x_2) \cos x_1 \\
    F(x_1, x_2) &= \frac{2a + 1}{4a}(2a \cos x_1 - 1) + \frac{x_2^2}{2}.
\end{align*}
\]

Michie Chambers Boxes 1968 - a never-ending story.


- Measure $\theta$, $\frac{d\theta}{dt}$, $x$, $\frac{dx}{dt}$
- Quantize crudely: position and angle 5 levels, velocities 3 levels, gives 225 boxes
- On-off control
- Scoring method LL: left life, RL right life, LU left usage, RU right usage, ...
- Many followers
  - Quantization and smoothing
  - Neural networks, many versions
  - Genetic optimization
  - Fuzzy, neural, neuro-fuzzy
  - Machine learning
- Tobias Glück https://www.youtube.com/watch?v=Lt-KLtDlh8
Lotfi Zadeh Fuzzy Logic 1965
Mamdani Fuzzy Control 1974
F. L. Smith Fuzzy control of cement kilns 1981
Blue Circle Cement Linkman
Hitachi subway system 1987
Laboratory for International Fuzzy Engineering Tokyo 1988
Japan Society for Fuzzy Theory and Systems SOFT 1989
Fuzzy Logic Systems Institute 1990
Center for promotion of Fuzzy Engineering TIT 1991
Lots of products from Japan with Fuzzy Omron, Sharp
Fuzzy Logic

- Rule based control - one way to describe nonlinearities
- Linguistic variables *high, low, medium* and membership functions
- If temperature *high* then increase flow *a little*

![Diagram showing membership functions for temperature categories: cold, moderate, hot, cold and moderate, cold or moderate.](image-url)
Rule 1: If $e$ is $N$ and $\frac{de}{dt}$ is $P$ then $u$ is $Z$
Rule 2: If $e$ is $N$ and $\frac{de}{dt}$ is $Z$ then $u$ is $NM$
Rule 3: If $e$ is $N$ and $\frac{de}{dt}$ is $N$ then $u$ is $NL$
Rule 4: If $e$ is $Z$ and $\frac{de}{dt}$ is $P$ then $u$ is $PM$
Rule 5: If $e$ is $Z$ and $\frac{de}{dt}$ is $Z$ then $u$ is $Z$
Rule 6: If $e$ is $Z$ and $\frac{de}{dt}$ is $N$ then $u$ is $NM$
Rule 7: If $e$ is $P$ and $\frac{de}{dt}$ is $P$ then $u$ is $PL$
Rule 8: If $e$ is $P$ and $\frac{de}{dt}$ is $Z$ then $u$ is $PM$
Rule 9: If $e$ is $P$ and $\frac{de}{dt}$ is $N$ then $u$ is $Z$

<table>
<thead>
<tr>
<th>$e$</th>
<th>$P$</th>
<th>$Z$</th>
<th>$N$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N$</td>
<td>$Z$</td>
<td>$NM$</td>
<td>$NL$</td>
</tr>
<tr>
<td>$Z$</td>
<td>$PM$</td>
<td>$Z$</td>
<td>$NM$</td>
</tr>
<tr>
<td>$P$</td>
<td>$PL$</td>
<td>$PM$</td>
<td>$Z$</td>
</tr>
</tbody>
</table>
The fuzzy statement

\( \text{If } x \text{ is } A \text{ and } y \text{ is } B \)

is interpreted as the crisp variable

\[ z^0 = \min(f_A(x_0), f_B(y_0)) \]

and is equivalent to minimization of the membership functions. The linguistic variable \( u \) defined by

\( \text{If } x \text{ is } A \text{ or } y \text{ is } B \text{ then } u \text{ is } C \)

or is interpreted as a linguistic variable with the membership function

\[ f_u(x) = z^0 f_C(x). \]
Defuzzification

Consider a linguistic variable $A$ with the membership function $f_A(x)$. Defuzzification by mean values gives the value

$$x_0 = \frac{\int x f_A(x) \, dx}{\int f_A(x) \, dx}.$$

Defuzzification by the centroid gives the real variable $x_0$ that satisfies

$$\int_{-\infty}^{x_0} f_A(x) \, dx = \int_{x_0}^{\infty} f_A(x) \, dx.$$
Fuzzy Control

If $e$ large positive and $\frac{de}{dt}$ large positive then $u$ large
If $e$ med positive and $\frac{de}{dt}$ med negative then $u$ zero
Two Views of Fuzzy Control

- Good language for translating manual operating practice (control of cement kilns)
- Snake-oil salesmen
- Software for generating nonlinearities

Picture from Karl-Erik Årzen
A knowledge bases system is used for monitoring, process supervision and switching of control and estimation algorithms.

ABB’s notion of state-based control
Theme: Brick by brick.
Some Useful Blocks

- PID
- Linearities
  - Low pass, band pass, high pass
  - Smith type of predictors
  - Notch filters
  - Delays
  - Posicast
- Static nonlinearities
  - Saturation
  - Selectors
  - Jump and rate
- State estimators
- Parameter estimators
- Optimizers
- Neural networks
  - Nonlinear multivariable function with learning mechanism
Layering, abstraction and formal definition of building blocks for control systems is a major research issue. It may bring control design to the level of VLSI design!

A rich collection of methods and ideas
- Generalized integral control
- Cascade control and Midranging (duals)
- Selectors
- Delay related, Smith predictors, posicast
- Internal model control
- Special course on adaptive control