Knowledge-Based Control Systems (SC4081)

Lecture 1: Introduction

Alfredo Núñez
Section of Railway Engineering
CiTG, Delft University of Technology
The Netherlands

a.a.nunezvicencio@tudelft.nl
http://staff.tudelft.nl/A.A.NunezVicencio
tel: 015-27 89355

Robert Babuška
Delft Center for Systems and Control
3mE, Delft University of Technology
The Netherlands

r.babuska@tudelft.nl
http://www.dcsc.tudelft.nl/~rbabuska
tel: 015-27 85117

Lecture Outline

1. General information about the course
2. Conventional control – a refresher
3. Intelligent control
4. Introduction to fuzzy sets

Course Information

Knowledge-Based Control Systems (SC4081)

- **Lecturers**:
  - Alfredo Núñez, weeks 3.01, 3.02 and 3.03
  - Jens Kober, week 3.04
  - Hans Hellendoorn, Monday week 3.05
- **Assistants**:
  - Sachin Navalkar
  - Vahab Rostampour
- **Lectures**: (9 lectures = 18 hours)
  - Monday (15:45 – 17:30) in lecture hall Chip at EWI
  - Wednesday (15:45 – 17:30) in lecture hall Chip at EWI

Knowledge-Based Control Systems (SC4081)

- **Examination**: (check yourself the dates and times!):
  - 13 April 2016, 9:00-12:00.
  - 24 June 2016, 9:00-12:00.

  Exam constitutes 60% of the final grade, remaining 40% are two assignments: **Literature and Matlab assignment**
- To obtain the credits of this course:
  Each activity must be approved.

Matlab Assignment

Objectives:

- Get additional insight through Matlab implementation.
- Apply the tools to practical (simulated) problems.

The assignment consists of three problems: fuzzy systems, fuzzy control and neural networks modeling.

Work in groups of two students, more information later.

Will be handed out on **February 24, 2016**
Report deadline **April 6, 2016**

Literature Assignment

Objectives:

- gain knowledge on recent research results through literature research
- learn to effectively use available search engines
- write a concise paper summarizing the findings
- present the results in a conference-like presentation

Deadlines – **March 16, March 23 and March 30 2016**
Symposium: Reserve the whole day **March 30, 2016**,
Rooms I and J, 3mE

Work in groups of four students.

Choose subject via Blackboard → SC4081 → Literature assignment – Do it this week!

Experiment for this quarter

Limited number of groups, you need to deliver literature assignment related to the competition, and the Matlab assignment will be replaced by your App or codes for the competition.

CIS Mobile App competition and IEEE CIS competitions:

- CIS Mobile App competition
- Computation Intelligence in Forecasting
- Competition for real time anomaly detection
- The General Video Game AI competition

Deadlines and instructions are in general the same as in the assignments. Send Alfredo the files before deadlines.

Work in groups of four students.

Goals and Content of the Course

knowledge-based and intelligent control systems

1. Introduction to intelligent control
2. Fuzzy sets and systems
3. Data analysis and system identification
4. Knowledge based fuzzy control
5. Artificial neural networks
6. Control based on fuzzy and neural models
7. Basics of reinforcement learning
8. Reinforcement learning for control
9. Applications

Course Material

- Lecture notes: Robert Babuška: *Knowledge-Based Control Systems*. TU Delft, 2010 (available from NextPrint)
- Items available for download at: www.dcsc.tudelft.nl/˜SC4081
  - Transparencies as PDF files
  - Demos, examples, assignments with Matlab/Simulink
- Blackboard

The entire content of the lecture notes will be examined!

Where to run Matlab

- At your home PC: Matlab Classroom Kit (you can download it from Blackboard).
- Computer rooms at 3mE.
- Computer rooms of other faculties (e.g., at Drebbelweg)

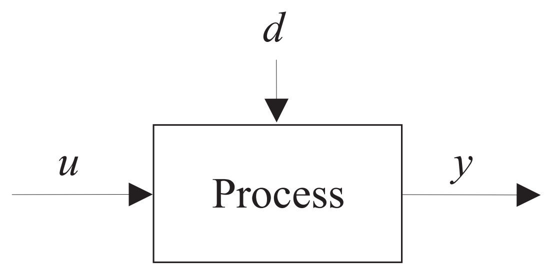
Prerequisites, Background Knowledge

- Mathematical analysis
- Linear algebra
- Basics of control systems (e.g., Control Systems)

Conventional Control

A Refresher

Process to Be Controlled



y : variable to be controlled (output)

u : manipulated variable (control input)

d : disturbance (input that cannot be influenced)

dynamic system

Examples of “Processes”

- technical (man-made) system

Examples of “Processes”

- technical (man-made) system
- natural environment

Examples of “Processes”

- technical (man-made) system
- natural environment
- organization (company, stock exchange)

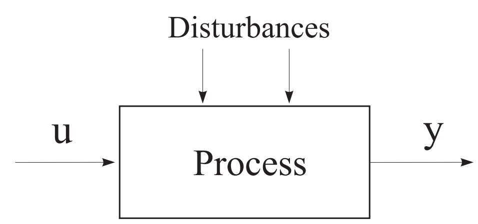
Examples of “Processes”

- technical (man-made) system
- natural environment
- organization (company, stock exchange)
- human body

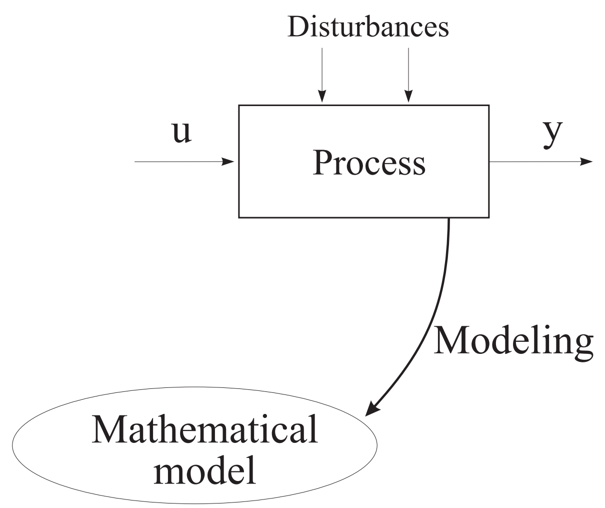
Examples of “Processes”

- technical (man-made) system
- natural environment
- organization (company, stock exchange)
- human body
- . . .

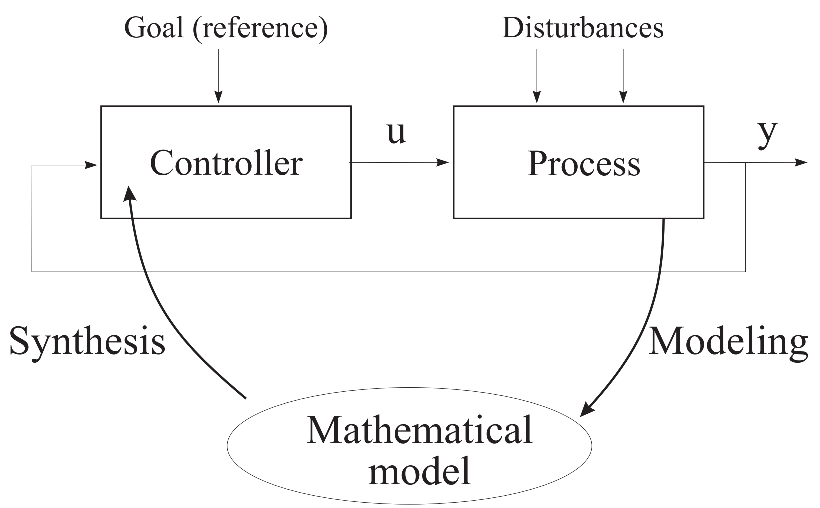
Classical Control Design



Classical Control Design



Classical Control Design



How to Obtain Models?

- physical (mechanistic) modeling
  1. first principles $\rightarrow$ differential equations (linear or nonlinear)
  2. linearization around an operating point

- system identification
  1. measure input–output data
  2. postulate model structure (linear–nonlinear)
  3. estimate model parameters from data (least squares)

Modeling of Dynamic Systems

$x(t)$ . . . state of the system

summarizes all history such that if we know $x(t)$ we can predict its development in time, $\dot{x}(t)$, for any input $u(t)$

linear state-space model:

$$\dot{x}(t) = Ax(t) + Bu(t)$$

Modeling of Dynamic Systems

$x(t)$ ... state of the system

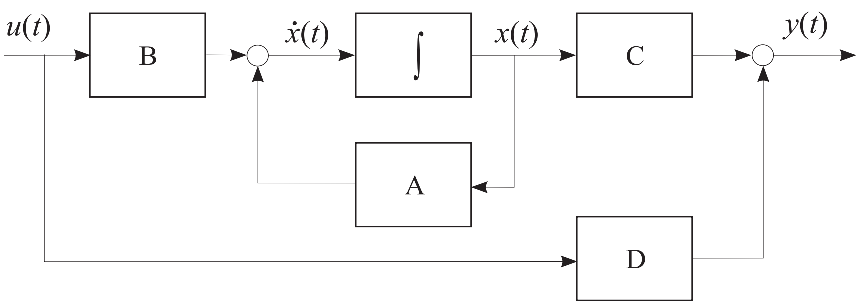
summarizes all history such that if we know $x(t)$ we can predict its development in time, $\dot{x}(t)$, for any input $u(t)$

linear state-space model:

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t)\\ y(t) &= Cx(t) + Du(t)\end{aligned}$$

Continuous-Time State-Space Model

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t)\\ y(t) &= Cx(t) + Du(t)\end{aligned}$$

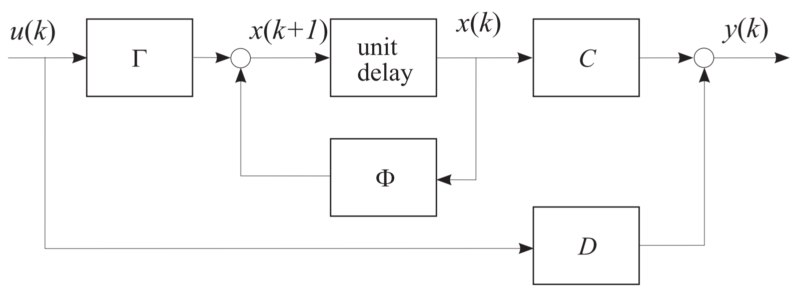


Discrete-Time State-Space Model

$$\begin{aligned}x(k+1) &= \Phi x(k) + \Gamma u(k)\\ y(k) &= Cx(k) + Du(k)\end{aligned}$$

Discrete-Time State-Space Model

$$\begin{aligned}x(k+1) &= \Phi x(k) + \Gamma u(k)\\ y(k) &= Cx(k) + Du(k)\end{aligned}$$



Input–Output Models

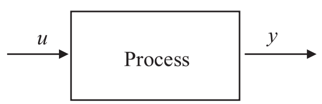
Continuous time:

$$y^{(n)}(t) = f\left(y^{(n-1)}(t), \ldots, y^{(1)}(t), y(t), u^{(n-1)}(t), \ldots, u^{(1)}(t), u(t)\right)$$

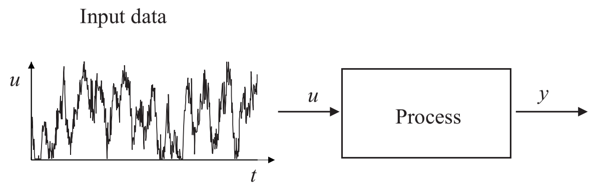
Discrete time:

$$\begin{aligned} y(k+1) \;=\; & f(y(k), y(k-1), \ldots, y(k-n_y+1), \ldots, \\ & u(k), u(k-1), \ldots, u(k-n_u+1)) \end{aligned}$$

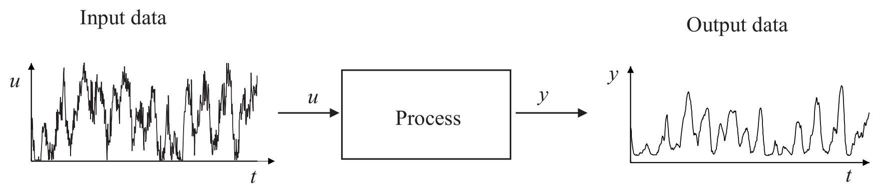
System Identification



System Identification



System Identification



$u(1), u(2), \ldots, u(N)$ $\qquad$ $y(1), y(2), \ldots, y(N)$

System Identification

Given data set $\{(u(k), y(k)) \mid k = 1, 2, \ldots, N\}$:

1. Postulate model structure, e.g.:

$$\hat{y}(k+1) = ay(k) + bu(k)$$

System Identification

Given data set $\{(u(k), y(k)) \mid k = 1, 2, \ldots, N\}$:

1. Postulate model structure, e.g.:

$$\hat{y}(k+1) = ay(k) + bu(k)$$

2. Form regression equations:

$$\begin{aligned} y(2) &= ay(1) + bu(1) \\ y(3) &= ay(2) + bu(2) \\ &\vdots \\ y(N) &= ay(N-1) + bu(N-1) \end{aligned}$$

in a matrix form: $\mathbf{y} = \boldsymbol{\varphi}[a\ b]^T$

System Identification

3. Solve the equations for $[a\ b]$ (least-squares solution):

$$\mathbf{y} = \boldsymbol{\varphi}[a\ b]^T$$

System Identification

3. Solve the equations for $[a\ b]$ (least-squares solution):

$$\mathbf{y} = \boldsymbol{\varphi}[a\ b]^T$$

$$\boldsymbol{\varphi}^T\mathbf{y} = \boldsymbol{\varphi}^T\boldsymbol{\varphi}[a\ b]^T$$

System Identification

3. Solve the equations for $[a\ b]$ (least-squares solution):

$$\mathbf{y} = \varphi[a\ b]^T$$

$$\varphi^T\mathbf{y} = \varphi^T\varphi[a\ b]^T$$

$$[a\ b]^T = [\varphi^T\varphi]^{-1}\varphi^T\mathbf{y}$$

System Identification

3. Solve the equations for $[a\ b]$ (least-squares solution):

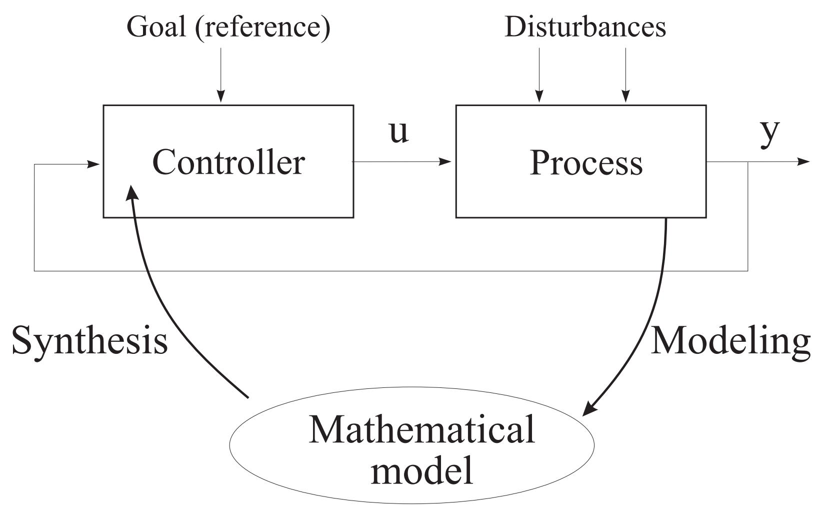
$$\mathbf{y} = \varphi[a\ b]^T$$

$$\varphi^T\mathbf{y} = \varphi^T\varphi[a\ b]^T$$

$$[a\ b]^T = [\varphi^T\varphi]^{-1}\varphi^T\mathbf{y}$$

Numerically better methods are available
(in MATLAB `[a b] =` φ `\ y`).

Classical Control Design



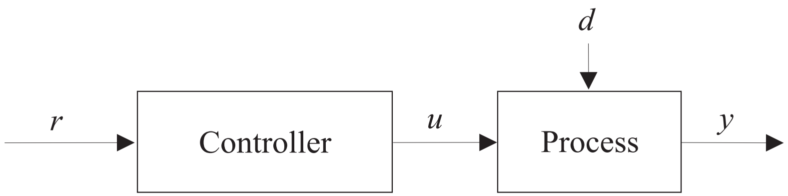
Design Procedure

- Criterion (goal)
  - stabilize an unstable process
  - suppress influence of disturbances
  - improve performance (e.g., speed of response)
- Structure of the controller
- Parameters of the controller (tuning)

Taxonomy of Controllers

- Presence of feedback: feedforward, feedback, 2-DOF
- Type of feedback: output, state
- Presence of dynamics: static, dynamic
- Dependence on time: fixed, adaptive
- Use of models: model-free, model-based

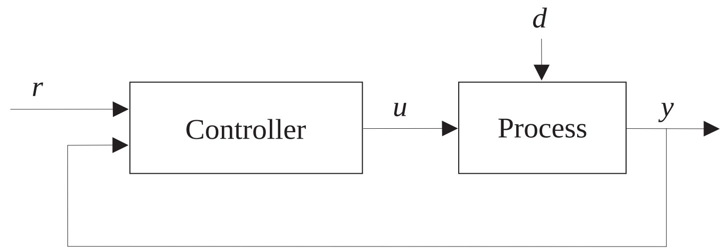
Feedforward Control



Controller:

- (dynamic) inverse of process model
- cannot stabilize unstable processes
- cannot suppress the effect of d
- sensitive to uncertainty in the model

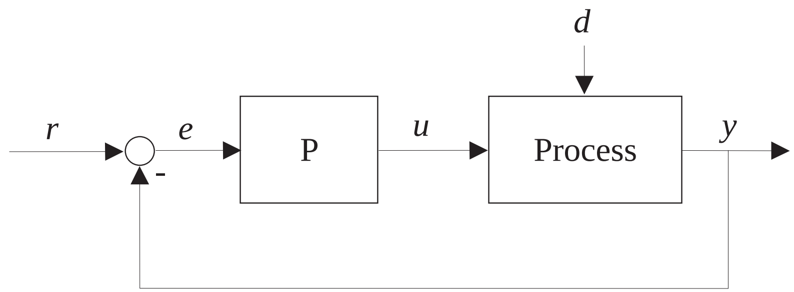
Feedback Control



Controller:

- dynamic or static ($\neq$ inverse of process)
- can stabilize unstable processes (destabilize stable ones!)
- can suppress the effect of d

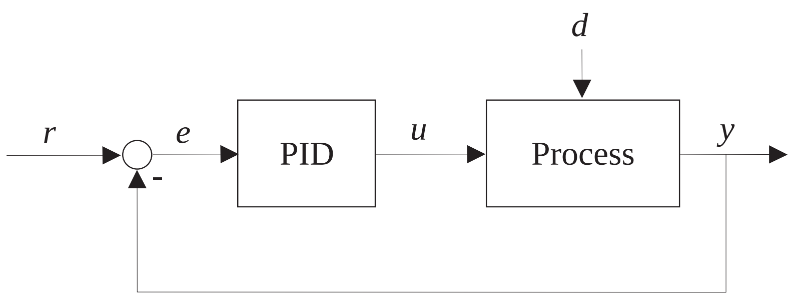
Proportional Control



Controller:

- static gain P: $u(t) = Pe(t)$

PID Control

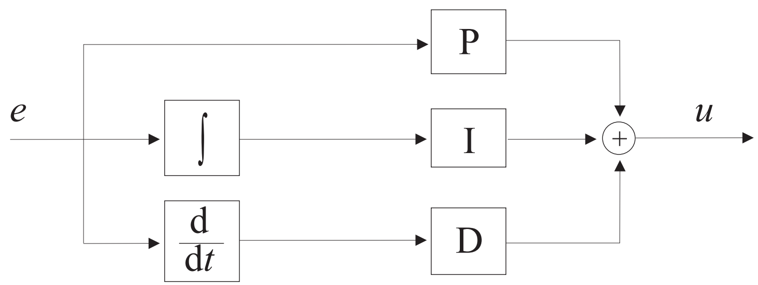


Controller:

- dynamic: $u(t) = Pe(t) + I\int_0^t e(\tau)d\tau + D\frac{de(t)}{dt}$
- P, I and D are the proportional, integral and derivative gains, respectively

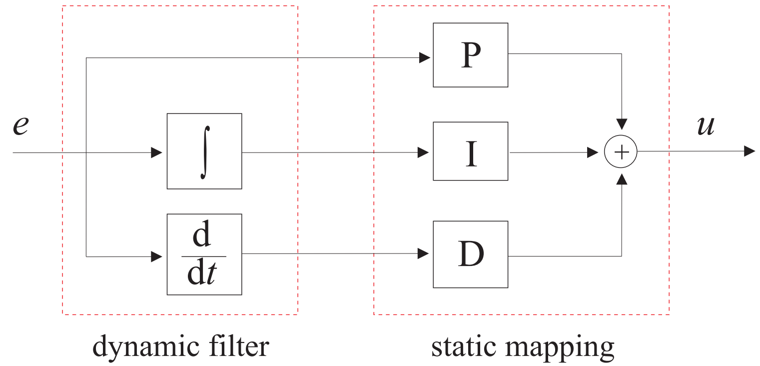
PID Control: Internal View

$$u(t) = Pe(t) + I\int_0^t e(\tau)d\tau + D\frac{de(t)}{dt}$$

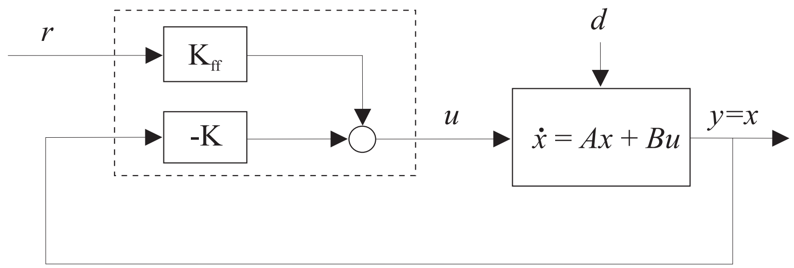


PID Control: Internal View

$$u(t) = Pe(t) + I\int_0^t e(\tau)d\tau + D\frac{de(t)}{dt}$$



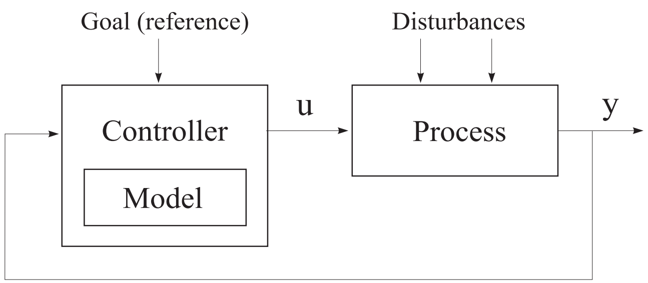
State Feedback



Controller:

- static: $u(t) = Kx(t)$
- K can be computed such that $(A + BK)$ is stable
- K_{ff} takes care of the (unity) gain from r to y

Model-Based Control



- state observer
- model-based predictive control
- adaptive control

Motivation for Intelligent Control

Pro's and Con's of Conventional Control

+ systematic approach, mathematically elegant
+ theoretical guarantees of stability and robustness

– time-consuming, conceptually difficult
– control engineering expertise necessary
– often insufficient for nonlinear systems

Additional Aspects

- control is a multi-disciplinary subject
- human factor may be very important
  - pilot
  - plant operator
  - user interface (e.g., consumer products)
- quest for higher machine itelligence

When Conventional Design Fails

- no model of the process available
  - $\rightarrow$ mathematical synthesis and analysis impossible
  - $\rightarrow$ experimental tuning may be difficult
- process (highly) nonlinear
  - $\rightarrow$ linear controller cannot stabilize
  - $\rightarrow$ performance limits

Example: Stability Problems

$$\frac{d^3y(t)}{dt^3} + \frac{d^2y(t)}{dt^2} + \frac{dy(t)}{dt} = y^2(t)u(t)$$

Use Simulink to simulate a proportional controller (nlpid.m)

Example: Stability Problems

$$\frac{d^3y(t)}{dt^3} + \frac{d^2y(t)}{dt^2} + \frac{dy(t)}{dt} = y^2(t)u(t)$$

Use Simulink to simulate a proportional controller (nlpid.m)

Conclusions:

- stability and performance depend on process output
- re-tuning the controller does not help
- nonlinear control is the only solution

Intelligent Control

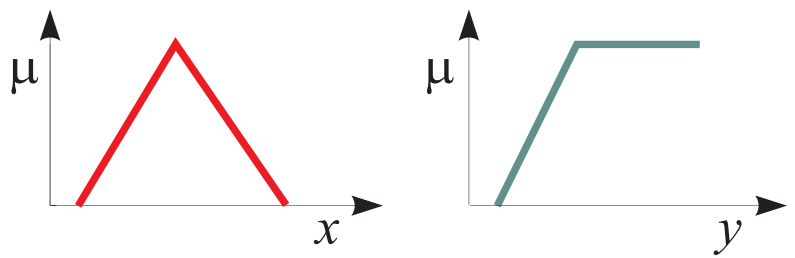
techniques motivated by human intelligence

- fuzzy systems (represent human knowledge, reasoning)
- artificial neural networks (adaptation, learning)
- genetic algorithms (optimization).

$\Rightarrow$ ***computational intelligence, soft computing***

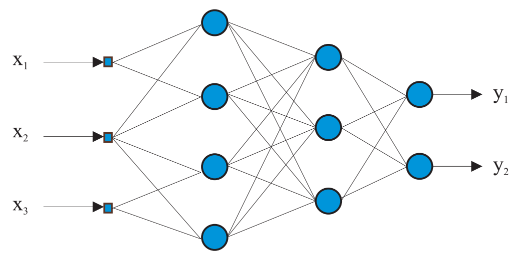
Knowledge Representation by If–Then Rules

If x is *Medium* **then** y is *Large*



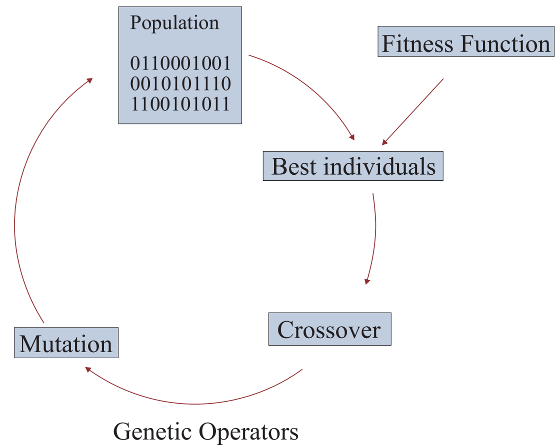
Artificial Neural Networks

Function approximation by imitating biological neural networks.



Learning, adaptation, optimization.

Genetic Algorithms

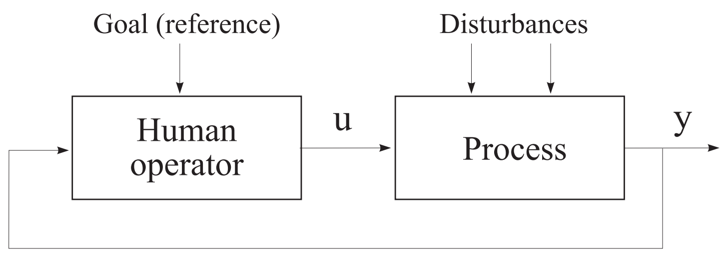


Optimization by imitating natural evolution.

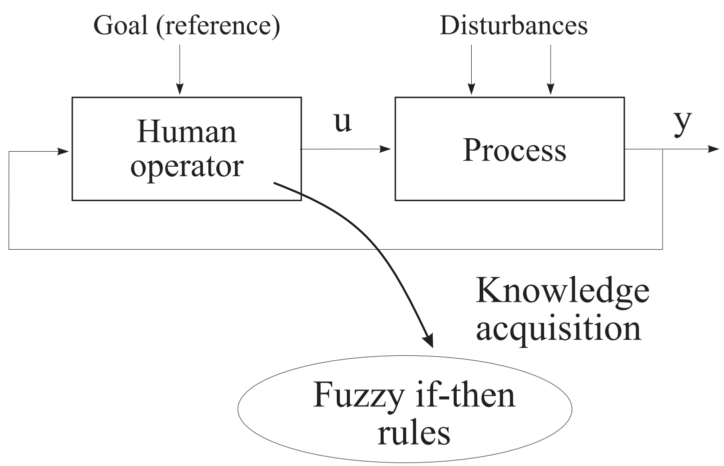
Intelligent Control

- Fuzzy knowledge-based control
- Fuzzy data analysis, modeling, identification
- Learning and adaptive control (neural networks)
- Reinforcement learning

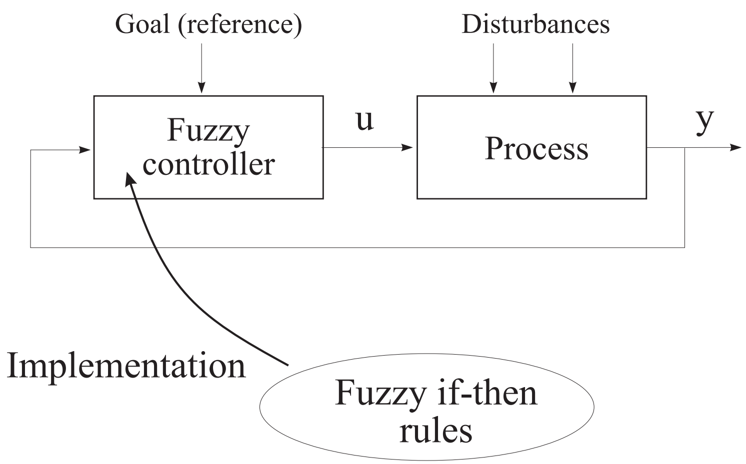
Direct Fuzzy Control



Direct Fuzzy Control



Direct Fuzzy Control



Fuzzy Sets and Fuzzy Logic

Relatively new methods for representing uncertainty and reasoning under uncertainty.

Types of uncertainty:

- chance, randomness (stochastic)
- imprecision, vagueness, ambiguity (non-stochastic)

Vagueness in If–Then Rules

If **temperature** in the burning zone *is OK*, and
oxygen percentage in the exhaust gases *is Low*, and
temperature at the back-end *is High*,
then reduce fuel *Slightly* and reduce fan speed *Moderately*.

Fuzzy Sets and Fuzzy Logic

Proposed in 1965 by L.A. Zadeh
(Fuzzy Sets, Information Control, vol. 8, pp. 338–353)

- generalization of ordinary set theory
- ’70 first applications, fuzzy control (Mamdani)
- ’80 industrial applications, train operation, pattern recognition
- ’90 consumer products, cars, special HW, SW.

The term “fuzzy logic” often also denotes fuzzy sets theory and its applications (e.g., fuzzy logic control).

Applications of Fuzzy Sets