Unit 7

Fuzzy Control with Examples

What is Fuzzy Control?

- Control is the continuous adaptation of parameters that influence a dynamic system with the aim to achieve a specific goal
- Usually, designing a controller requires a model of the dynamic system
- Often, only a vague description is available; fuzzy systems, therefore, are an appropriate alternative
- Fuzzy control is nothing else but applying fuzzy systems for such control tasks

Time-Discrete Closed-Loop Control

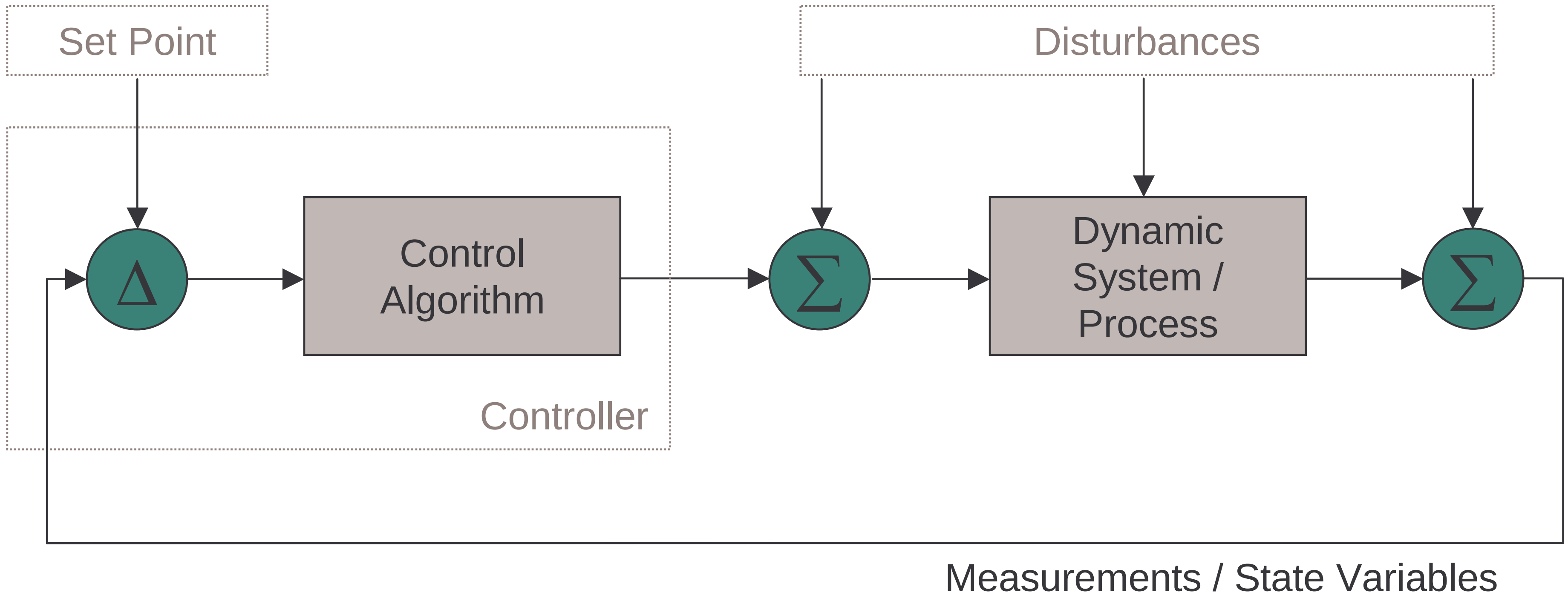
- Virtually all real-world systems are working in discrete time steps; let us assume that the time between two sampling cycles is Δt
- Ideally, the dynamic system has a state vector $(x_1(t), \ldots, x_k(t))$ and is modeled by a function f_S that computes a new state from the old state and a vector of control actions $(u_1(t), \ldots, u_l(t))$:

$$\big(x_1(t{+}\Delta t), \ldots, x_k(t{+}\Delta t)\big) = f_S\big(x_1(t), \ldots, x_k(t); u_1(t), \ldots, u_l(t)\big)$$

- The controller is a function f_C that computes the vector of control actions from the current state:

$$\big(u_1(t), \ldots, u_l(t)\big) = f_C\big(x_1(t), \ldots, x_k(t)\big)$$

A Typical Control Loop



Example: Cart Pole Centering Problem

State variables: angle and angular speed of pole, position and speed of cart

Control action: force/acceleration to be applied to cart

Fuzzy Systems in Control

Fuzzy systems are ready to be applied in control. However, the following issues have to be taken into account:

- Properties of the resulting control function f_C (e.g. smoothness)
- Stability
- Computational complexity (fuzzy controllers often have to be implemented on hardware with limited resources)

The Smoothness Issue

- Smoothness properties can only be guaranteed if the membership functions and all involved operators (T, S, N) and $(\tilde{I}, \tilde{T})$ or $(\tilde{T}, \tilde{S})$, respectively, fulfill these conditions
- Example 1: continuous membership functions and continuous operations guarantee a continuous control function f_C; note that MOM defuzzification is not continuous!
- Example 2: differentiable membership functions and differentiable operations guarantee a continuous control function f_C; note that this means that max-min inference may not yield a differentiable control function; sum-prod inference with COG defuzzification works (if all other conditions

The Stability Issue

- Nonchalantly speaking, stability is the property that the system behaves nice in a close proximity of the set point
- In particular, this means that no corrective control action should be taken if the system is already perfectly at the set point
- The stability question cannot be answered without explicit model knowledge
- Stability of fuzzy controllers is still an open issue

Shortcomings of Mamdani Max-Min Inference

Although common in practice and often sufficient, Mamdani max-min inference with COG defuzzification has shortcomings:

- The resulting control function f_C rarely has very nice properties
- Instable behavior can occur under some conditions
- Most often, the maximal values for output variables cannot be achieved

Sum-prod inference somehow solves the first of these issues, but not the other two.

An Alternative Approach: Sugeno Fuzzy Systems

Sugeno fuzzy systems use simplified rule actions:

$$\text{IF } \textit{cond}_i \text{ THEN } N_y = y_i$$

Assume that we have an input vector $(x_1, \ldots, x_n)$ and that the conditions *cond*$_i$ are fulfilled with degrees $t_i(x_1, \ldots, x_n)$. Then the output $f_C(x_1, \ldots, x_n)$ is defined as the sum of action values y_i weighted by the degrees of fulfillment:

$$f_C(x_1, \ldots, x_n) = \frac{\sum_{i=1}^{m} t_i(x_1, \ldots, x_n) \cdot y_i}{\sum_{i=1}^{m} t_i(x_1, \ldots, x_n)}$$

Sugeno fuzzy systems, in this sense, are piecewise constant step functions with fuzzy steps.

A Variant: Takagi/Sugeno/Kang (TSK) Fuzzy Systems

TSK fuzzy systems use rule actions like

IF *cond*$_i$ THEN $N_y = f_i(x_1, \ldots, x_n)$

where $f_i(x_1, \ldots, x_n) = a_i^0 + a_i^1 \cdot x_1 + \cdots + a_i^n \cdot x_n$. Assume that we have an input vector $(x_1, \ldots, x_n)$ and that the conditions *cond*$_i$ are fulfilled with degrees $t_i(x_1, \ldots, x_n)$. Then the output $f_C(x_1, \ldots, x_n)$ is defined as the sum of action functions f_i weighted by the degrees of fulfillment:

$$f_C(x_1, \ldots, x_n) = \frac{\sum_{i=1}^{m} t_i(x_1, \ldots, x_n) \cdot \left(a_i^0 + a_i^1 \cdot x_1 + \cdots + a_i^n \cdot x_n\right)}{\sum_{i=1}^{m} t_i(x_1, \ldots, x_n)}$$

TSK fuzzy systems, therefore, can be considered as piecewise affine linear functions with fuzzy transitions.

Advantages of Sugeno/TSK Fuzzy Systems

- Much less computational effort for "inference and defuzzification"
- Better analytical properties of the resulting control function
- Exact interpolation of output values y_i or $f_i(x_1, \ldots, x_n)$ can be achieved
- Stability easier to guarantee

Simplified Processing of Fuzzy Sets

- Images of fuzzy sets with respect to the fuzzy relation associated with the rule base (cf. Slides 179–181) provide an elegant theoretical framework for processing fuzzy inputs through a fuzzy rule base
- However, the computational effort to do so is enormous!
- A simplified variant is common:
  - Compute single compatibility degrees to which a fuzzy input matches a condition
  - Perform inference (Steps 3 and 4) as if the inputs were crisp

Simplified Processing of Fuzzy Sets in Detail

Assume that, instead of crisp input values, we have fuzzy sets A_i on X_i as inputs. Then the degree of fulfillment (compatibility) of A_i with a condition "N_i is l^i_j" is defined as

$$\text{sup}\{T'(\mu_{A_i}(x), \mu_{M_i(l^i_j)}(x)) \mid x \in X_i\},$$

where T' is some t-norm that is chosen a-priori. Analogous to the standard case (see Slide 168), we can perform Step 2.

That means we simplify the problem by computing a single degree of fulfillment to which the fuzzy sets fulfill the conditions of the rules in the rule base.

Note that the result will only be an approximation of what we would get

Demonstrator: Cart Pole Centering Problem

- Software demonstrator (simulated system)
- Done as assignment work by Christian Aistleitner at Johannes Kepler University Linz in Winter 2002/2003
- Implementation technology:
  - Fuzzy controller: *fuzzyTech*
  - Programming of simulation and GUI: *MS Visual C*
- Three separate control strategies for
  - swinging the pole up
  - stabilizing the pole
  - moving the cart with the centered pole to the target po-

Demonstrator: Hovering Ball

- Full-fledged hardware demonstrator
- Done as project work by Michael Landsiedl, Stefan Raiser, and Christian Grillberger at Johannes Kepler University Linz in 1999
- Hardware side:
  - DC ventilator for controlling the position of the ball
  - Sharp infrared distance sensor for detecting ball position
  - Rotation of ventilator controlled by Motorola 68HC11 chip; programmed in Assembler
  - Data exchange with PC via RS232

Demonstrator: Hovering Ball (cont'd)

- PC side:
  - Fuzzy set and rule editors implemented from scratch in C++/MFC
  - Fuzzy inference is performed on PC in a fully transparent way
  - Online modification of parameters and fuzzy system possible

Commercial Applications in Consumer Goods

- Washing machines
- Vacuum cleaners
- Rice cookers
- Showers
- Heating/air condition control
- . . .

Large-Scale Commercial Applications

- Cement kiln
- Sendai subway
- Danube water level control
- . . .