Autonomic Computer Systems CS321: Introduction to Control Theory

December 1, 2011

Prof. Dr. Christian Tschudin

Computer Science Department, University of Basel

Why Control Theory (in a network centric course)?

“Dynamics” usually a second thought in CS: functionality first. However, performance (beyond speed) matters in practice: unstable operation is useless.

- Dynamics is key for TCP
- Control Theory for engineering (using computer programs):
  – motor control of an inverted pendulum, e.g. Segway
  – Mindstorm version: see
  `http://www.brumann-art-production.ch/NXT_Standalone/`

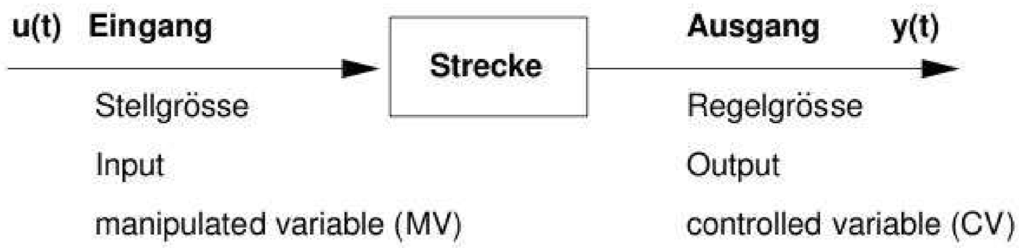
Can control theory be applied in a (autonomic) computing center?

Overview 2011-12-01

- Introduction to Control Theory
  problem setting, history, modeling
- Hellerstein et al slide set

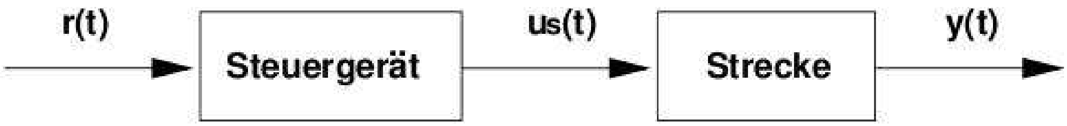
Terminology

- Control theory – *Theorie der Regelsysteme*
- Plant, or process – *Strecke*
  – "the thing to control"
  – has inputs: actuator, manipulated variable
  – has outputs: sensor, controlled variable



a) Feedforward Control (*Steuerung*)

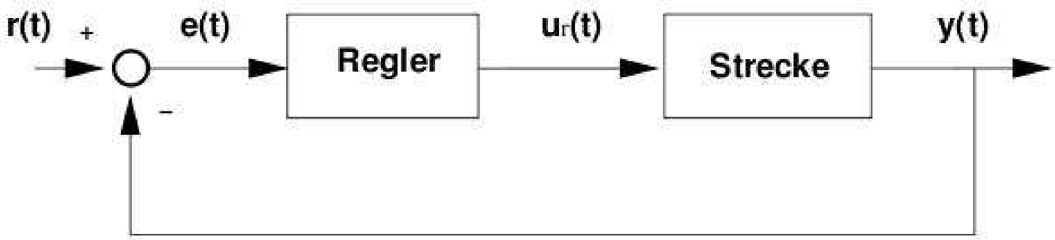
We "prepend" a controling device (controller)
that, given some reference value r,
generates the appropriate control signal u
such that $r \approx y$.



This is also called "**open loop** control".

b) Feedback Control (*Regelung*)

We add a control device (controller)
that, given an error signal $e = r - y$,
generates the appropriate control signal u
such that $e \approx 0$.



Question: why should an error signal (instead of r and y) be sufficient as input for the controller?

c) Learning how to Control (similar to Feedback)

- Using the observed output y, one can define a mapping from y to u based on trials.
- Process:
  – try out best u (for given y and r), done offline
  – implement as a lookup table
- This approach might be needed if system dynamics is too fast for actuating.
- Problem: cannot be subject to quantitative analysis.

Open Loop Example

Classical example: cruise control

- Car driver sets car speed manually, car keeps engine power steady
- y – actual speed
  r – desired speed
  u – angle of throttle (Drosselklappe)

Can be tuned for flat road, but will be off in the mountains.

Discussion of Open Loop Control

Simple: Compute control input without continuous variable measurement

- Need to know *everything accurately* to work right
  Cruise-control car: friction(t), ramp_angle(t), wind speed
- Open-loop control fails when
  – We don't know everything
  – We make errors in estimation/modeling
  – Things change

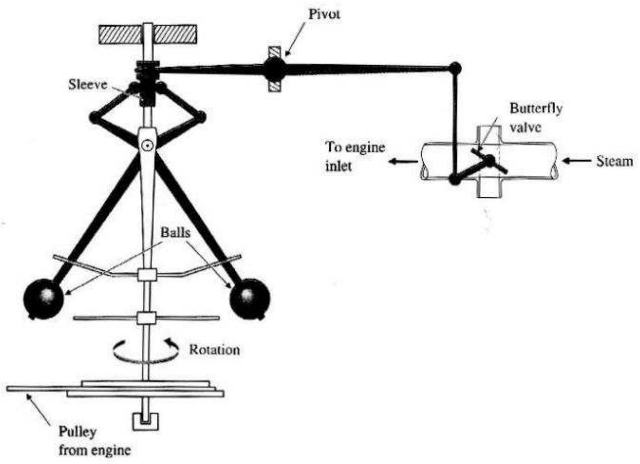
Feedback Control Theory

How to design the "optimal controller"?
No single solution, single approach: control theor**ies**.

- Linear vs. non-linear (differential equations)
- Deterministic vs. Stochastic
- Time-invariant vs. Time-varying control
  (are coefficients in controller functions of time)?
- Continuous-time vs. Discrete-time

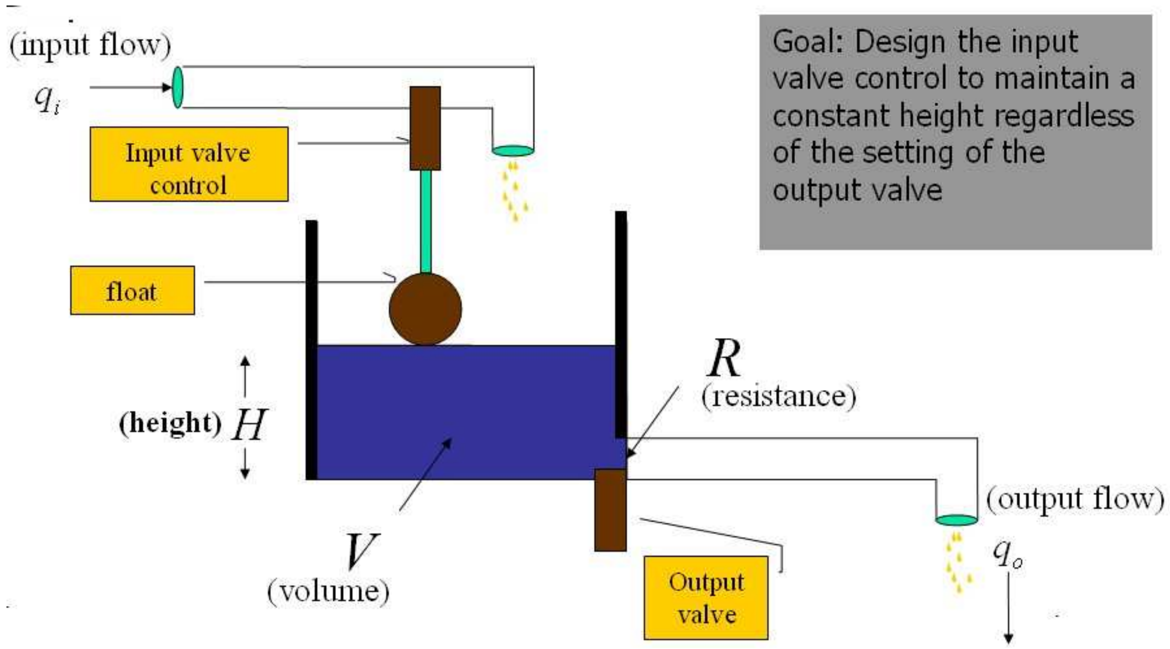
Even: mechanical controllers!

Historical controllers



- Breeding automaton (Drebbel, ca 1620) controls heating fire based on mercury thermometer
- "Governor" for steam engine (Watts, ca 1790) shown to the left

Another mechanical controler



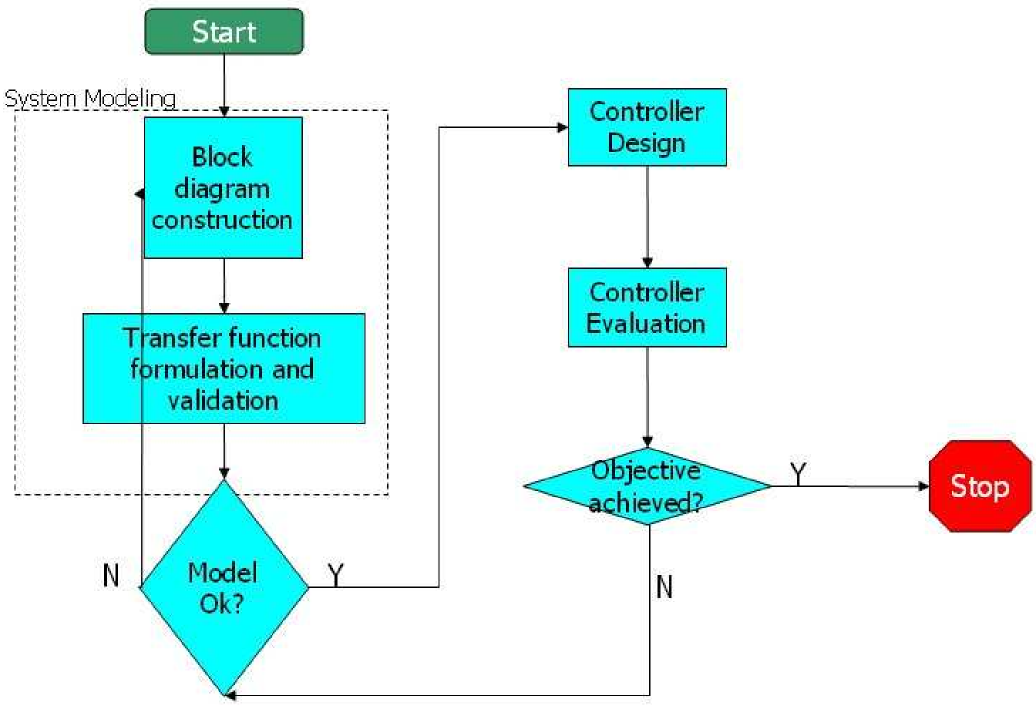
Goal: Design the input valve control to maintain a constant height regardless of the setting of the output valve

(Algorithmic) Controler Design: Basic Idea

- How does (uncontrolled) plant react?
  $\rightarrow$ capture behavior as a function (model)
- Design a (feedback) controller, to be added
  (controller have known dynamics, also a function)
- Goal: predict overall system (=combination of model and control function) behavior, including stablity etc

Problem: No closed theory how to find “optimal” controller.

Controler Design: Iterative Methodology



Mathematical Modelling

Introduce "system state" $x(t)$

- System state x is internal to the plant, time dependent
- x can be influenced (i.e., depends on) manipulated variable u
- A future system state is a function of its past state:
  $\dot{x} = f_1(x, u, t)$
- The observable state also such a function:
  $y = f_2(x, u, t)$

Note that x is usually not accessible, at all!

Mathematical Modelling (contd)

- Most interesting is how the system changes:
  $\frac{d}{dt}x(t) = a(x, t) + b(x, u, t)$
- and how this affects the controlled variable (observable):
  $y(t) = c(x(t), u(t), t)$

Usual assumption: linear behavior, time-invariant coefficients:

$$\frac{d}{dt}x(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t) + Du(t)$$

(A. . . D are matrices)

Systems can, sometimes, be described without reference to internal state x, for example: $\dot{y} + ky = u$