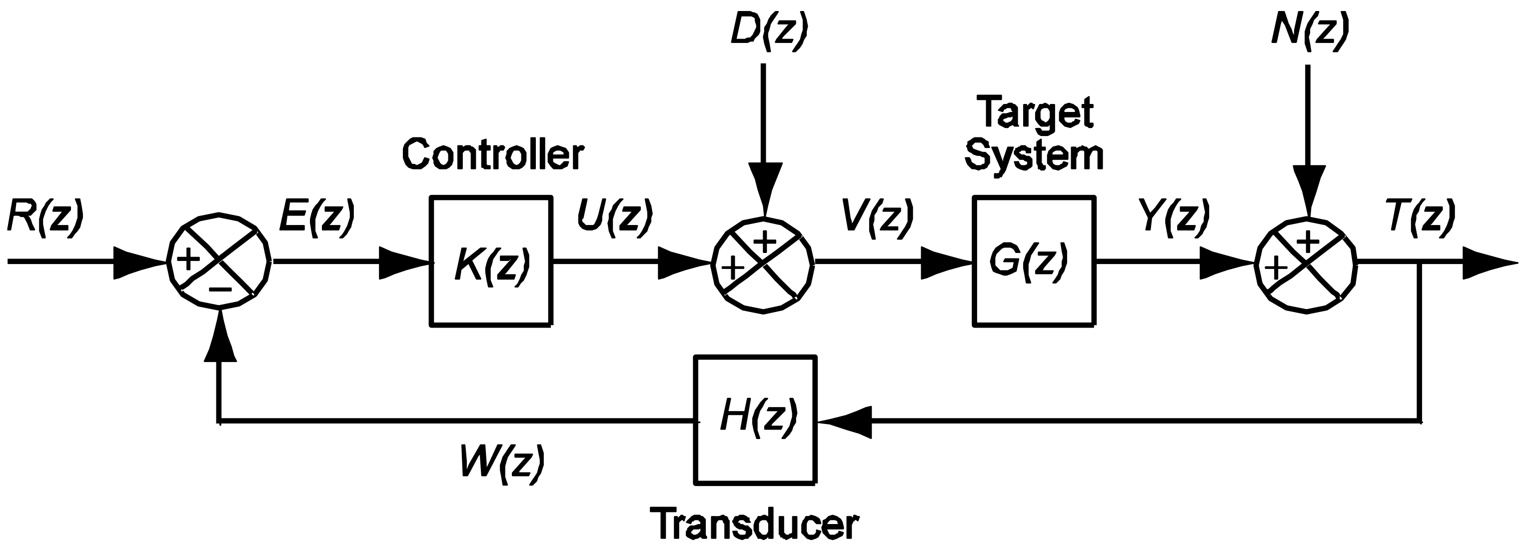
CSE 590K: Analysis and Control of Computing Systems Using Linear Discrete-Time System Theory

Advanced Topics

Joseph L. Hellerstein, Jie Liu
Microsoft Corporation

March 3, 2008

Challenges Not Addressed by Classical Control

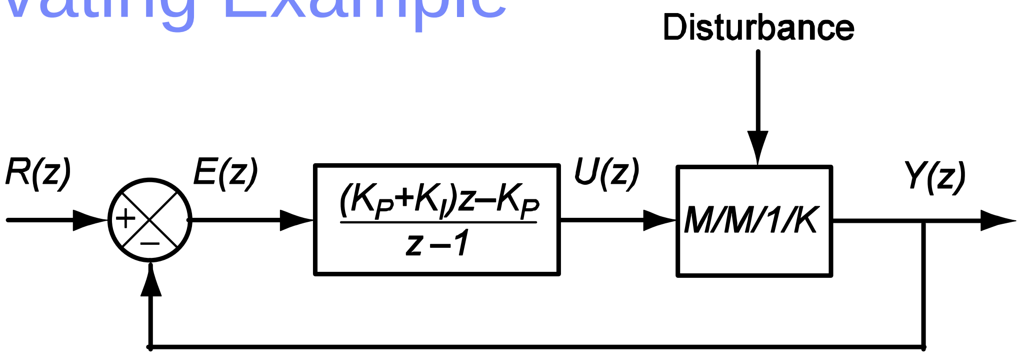


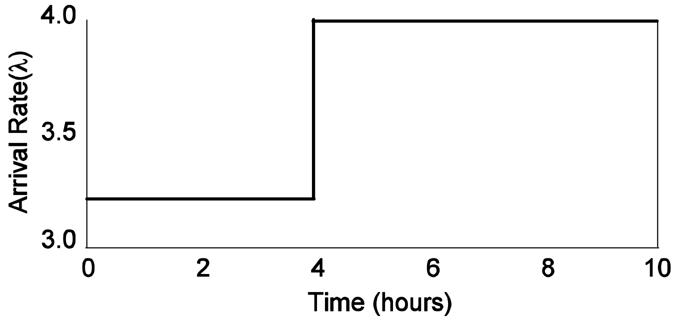
- Change in system model (e.g., workload changes)
- Stochastics
- Incomplete or imprecise model of target system

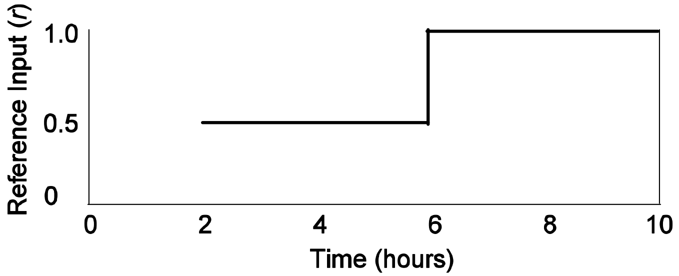
Agenda

- Motivating example
- Gain scheduling
- Self-Tuning regulators
- Minimum variance control
- Fuzzy control

Motivating Example



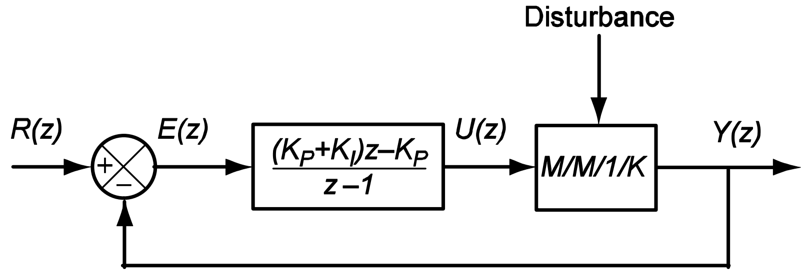




Approach to handling non-stationarities

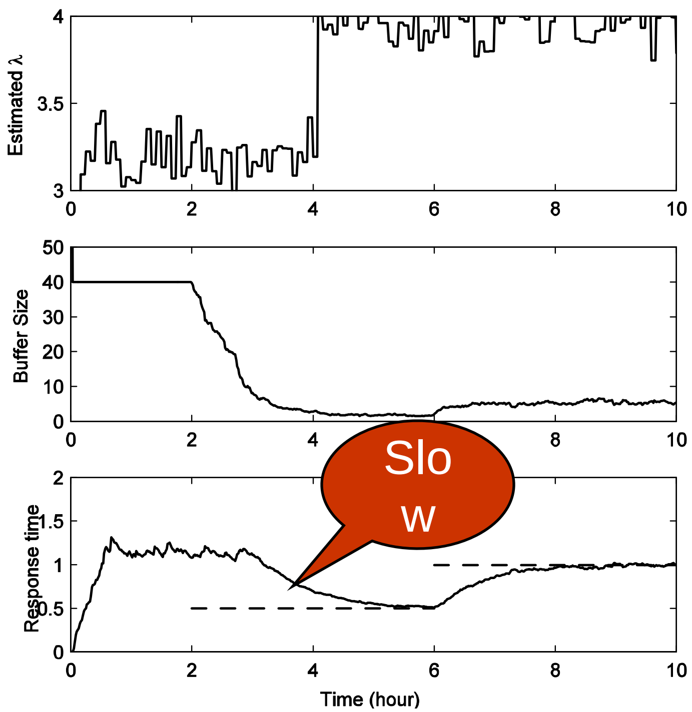
1. Design at one arrival rate
2. See if robust at other

Results

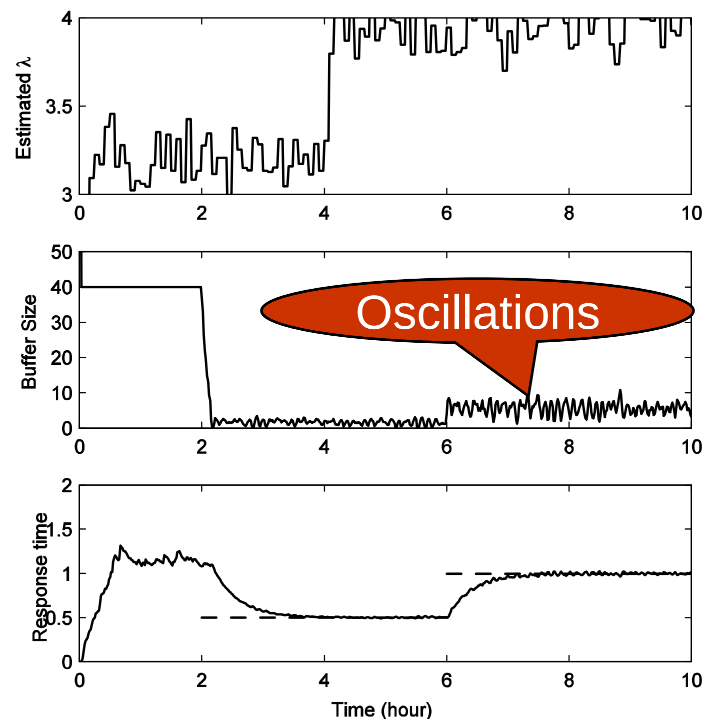


$$M/M/1/K\text{: Use } \frac{b}{z-a}$$

Designed using high arrival rates



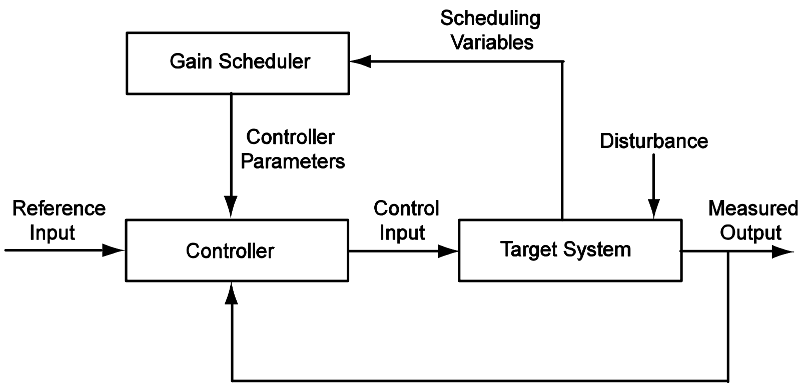
Designed using low arrival rates

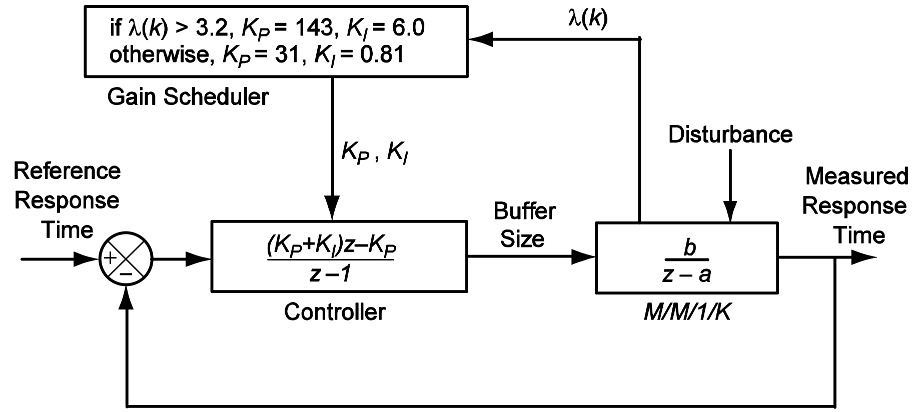


Gain Scheduling

- Addresses situations in which model changes
- Key concepts
  - Scheduling variable: measures environmental effects that determine the model to use
  - Gain scheduler: selects the model

Structure of Gain Scheduling



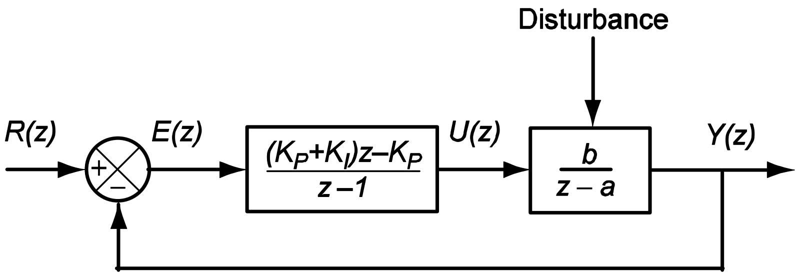


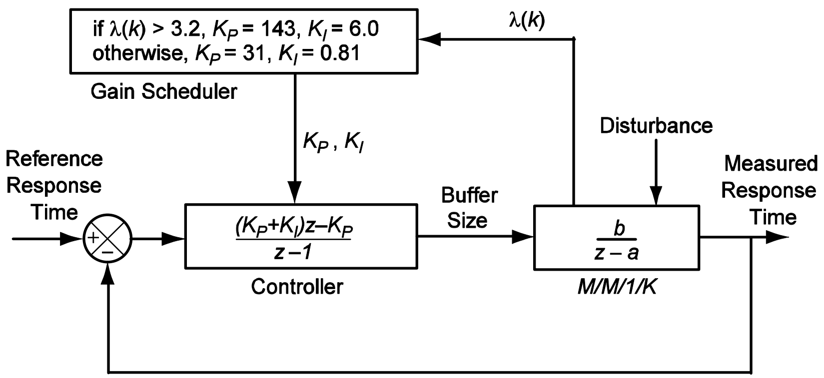
Principles for developing gain schedulers

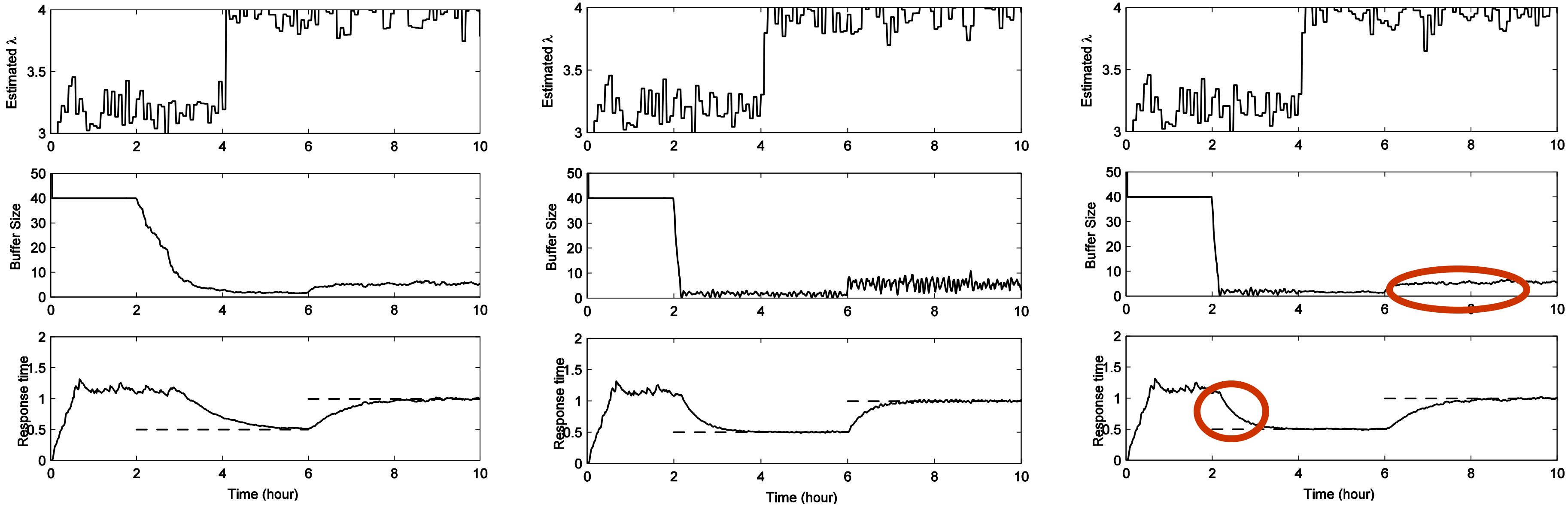
1. Identify the variables that characterize how the target system changes
2. Develop rules that determine which control gains to use with which values of scheduling variables. Conceptually, gain scheduler indexes into a table of the form:

Rule 1	1st Gains
Rule 2	2nd Gains
…	…
Rule n	n-th Gains

Performance of Gain Scheduling



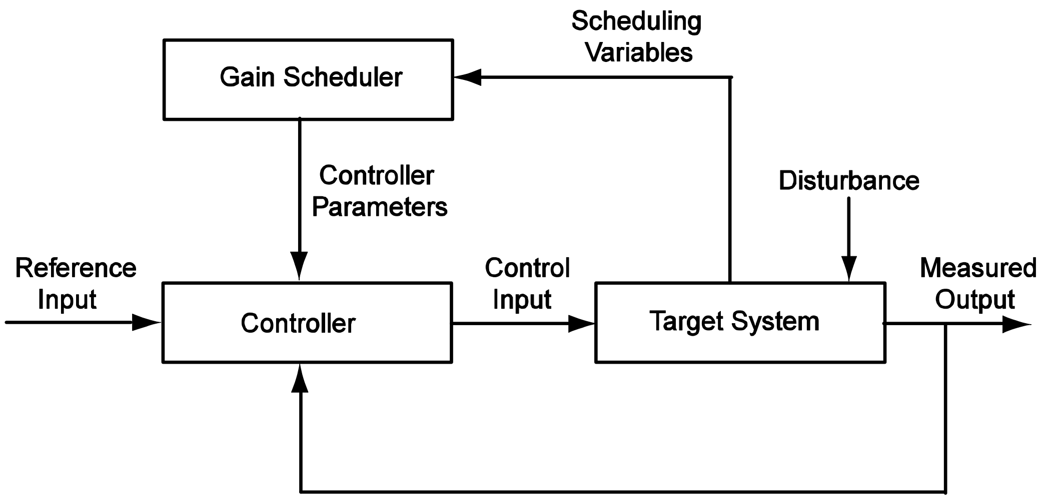




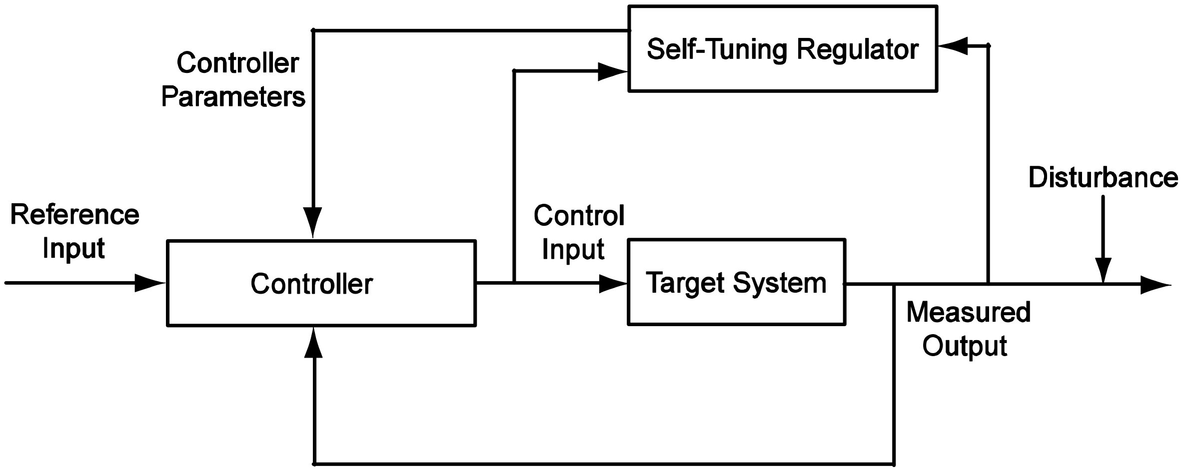
Self-Tuning Regulator (STR)

- Issue with gain scheduling
  - Discrete set of controllers
- Self-tuning regulators
  - Computes a continuous set of gains
  - Based on updated model of the target system

Structure of Self-Tuning Regulators

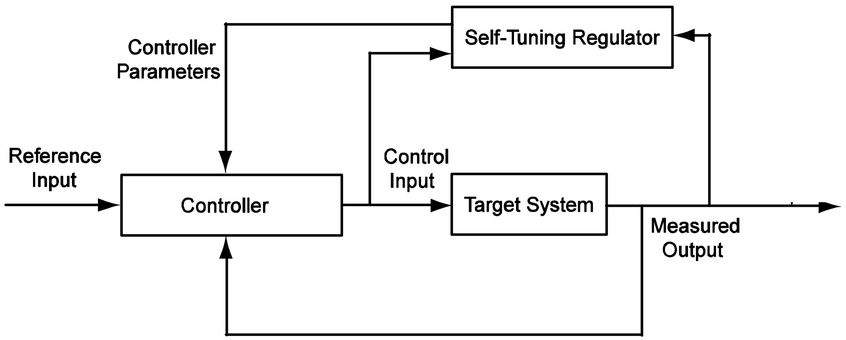


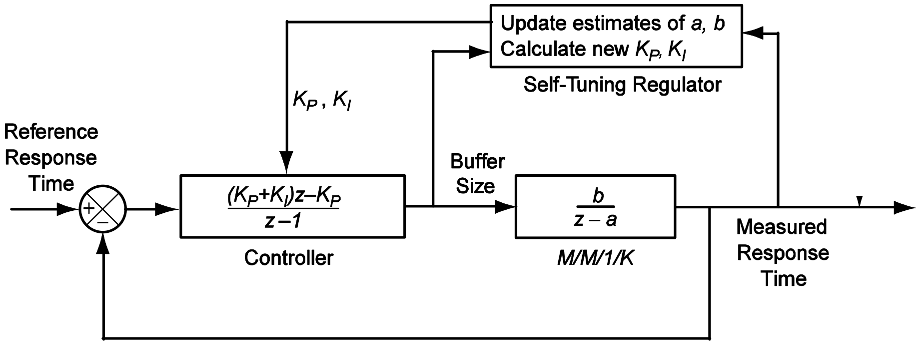
Gain Scheduler



STR

Structure of Self-Tuning Regulators





Principles for developing self-tuning regulators

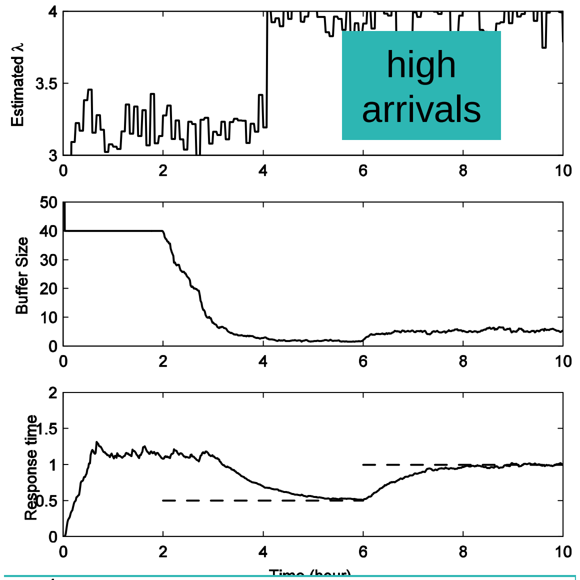
1. Express control gains in terms of the a_n and b_m and desired poles (p_1, p_2).
2. Use recursive least squares (or other techniques) to estimate a_n and b_m on-line
3. Compute new control gains whenever the a_n and b_m are updated

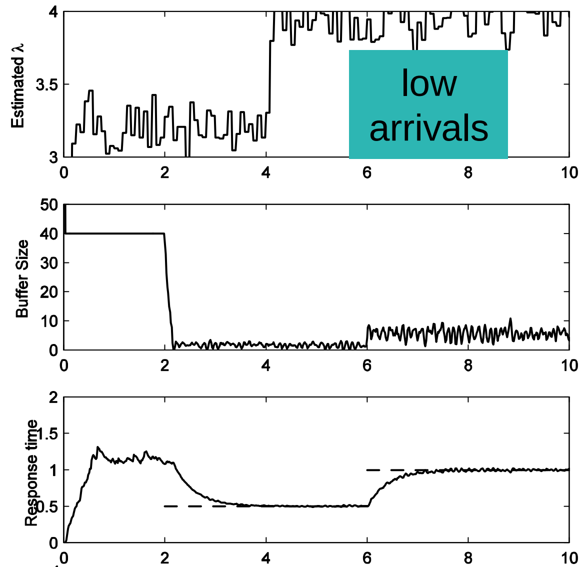
Example for a PI controller:

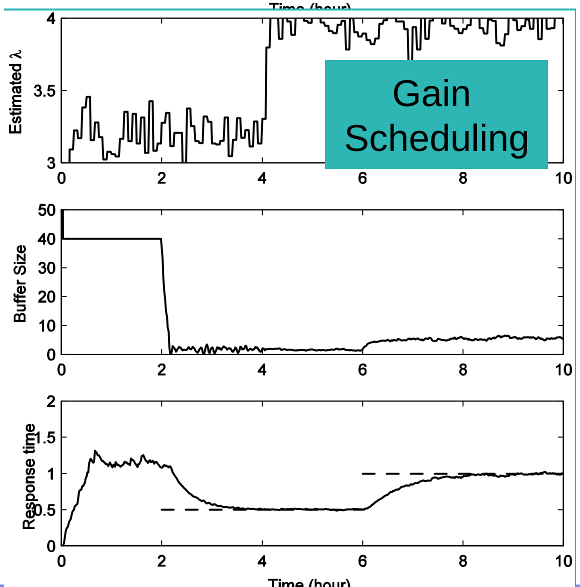
$$K_P = \frac{a - p_1 p_2}{b}$$

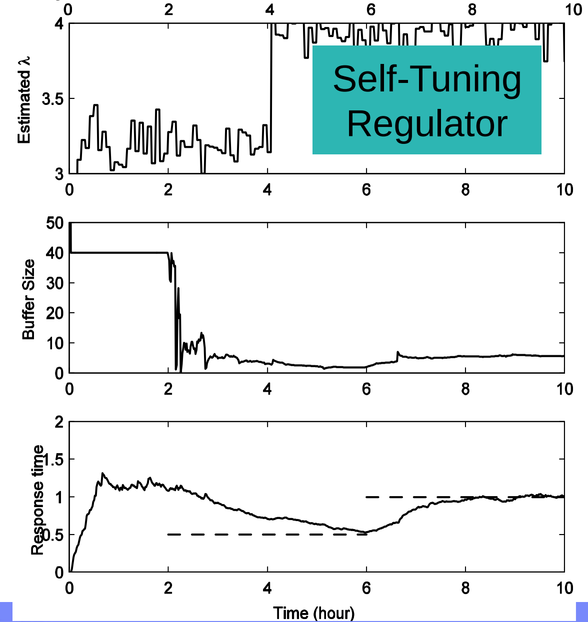
$$K_I = \frac{1 + p_1 + p_2 + p_1 p_2}{b}$$

Performance of Self-Tuning Regulator







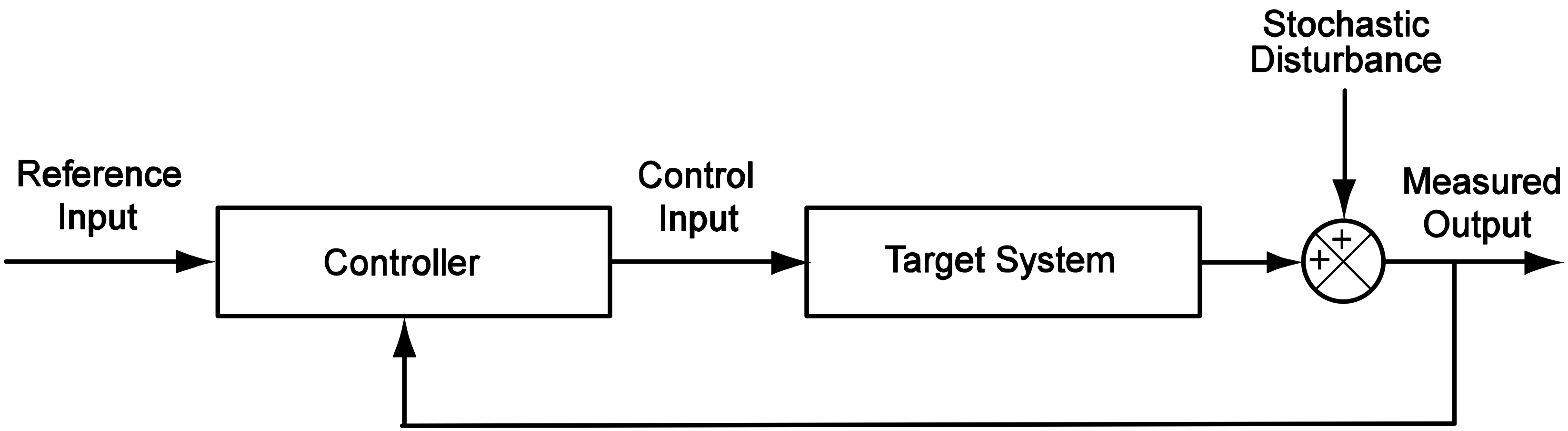


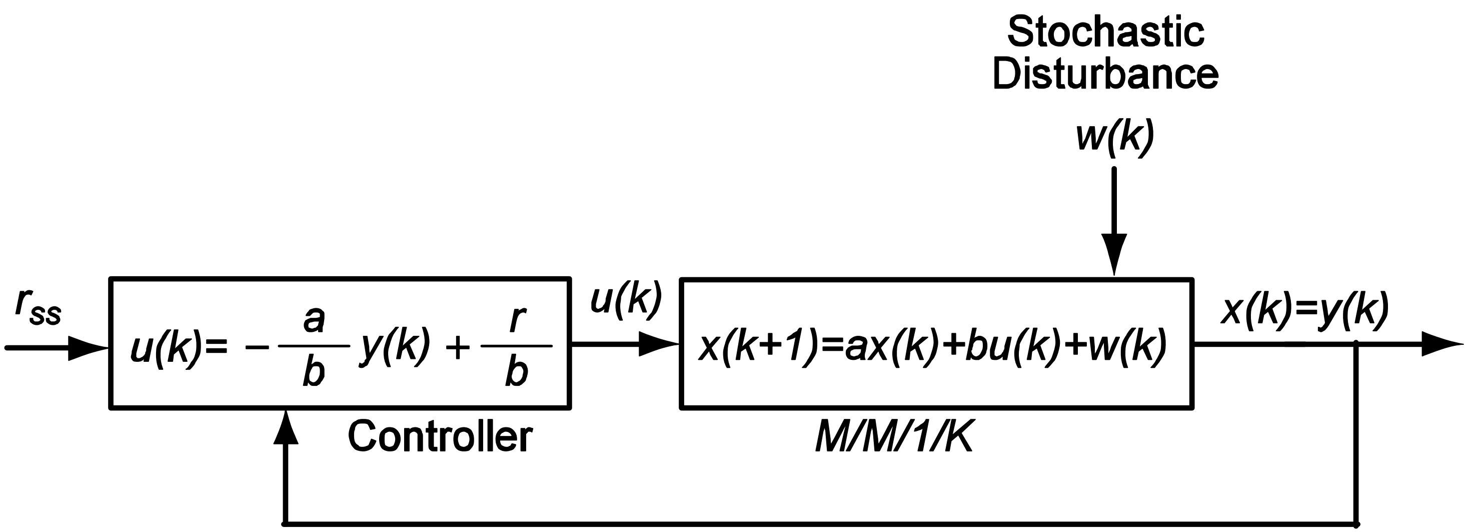
- STR is slower than gain scheduling
- But STR is easier to design

Minimum Variance Control (MVC)

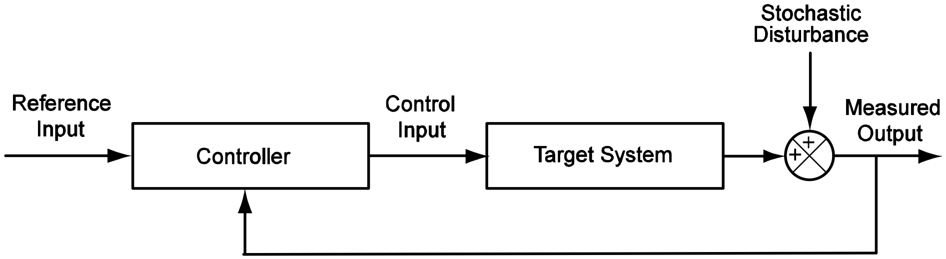
- Metrics of computing systems are often dominated by stochastics
- MVC provides an explicit way to control variance

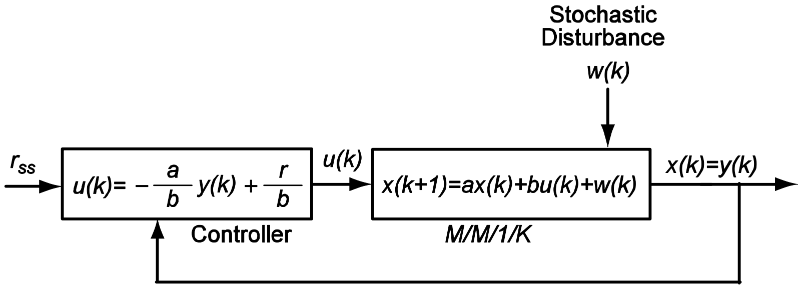
Structure of Minimum Variance Control





Structure of Minimum Variance Control





$$y(k+1) = ay(k) + bu(k) + w(k);\ w(k) \text{ is a random variable, } E[y(k)] = r_{ss}$$

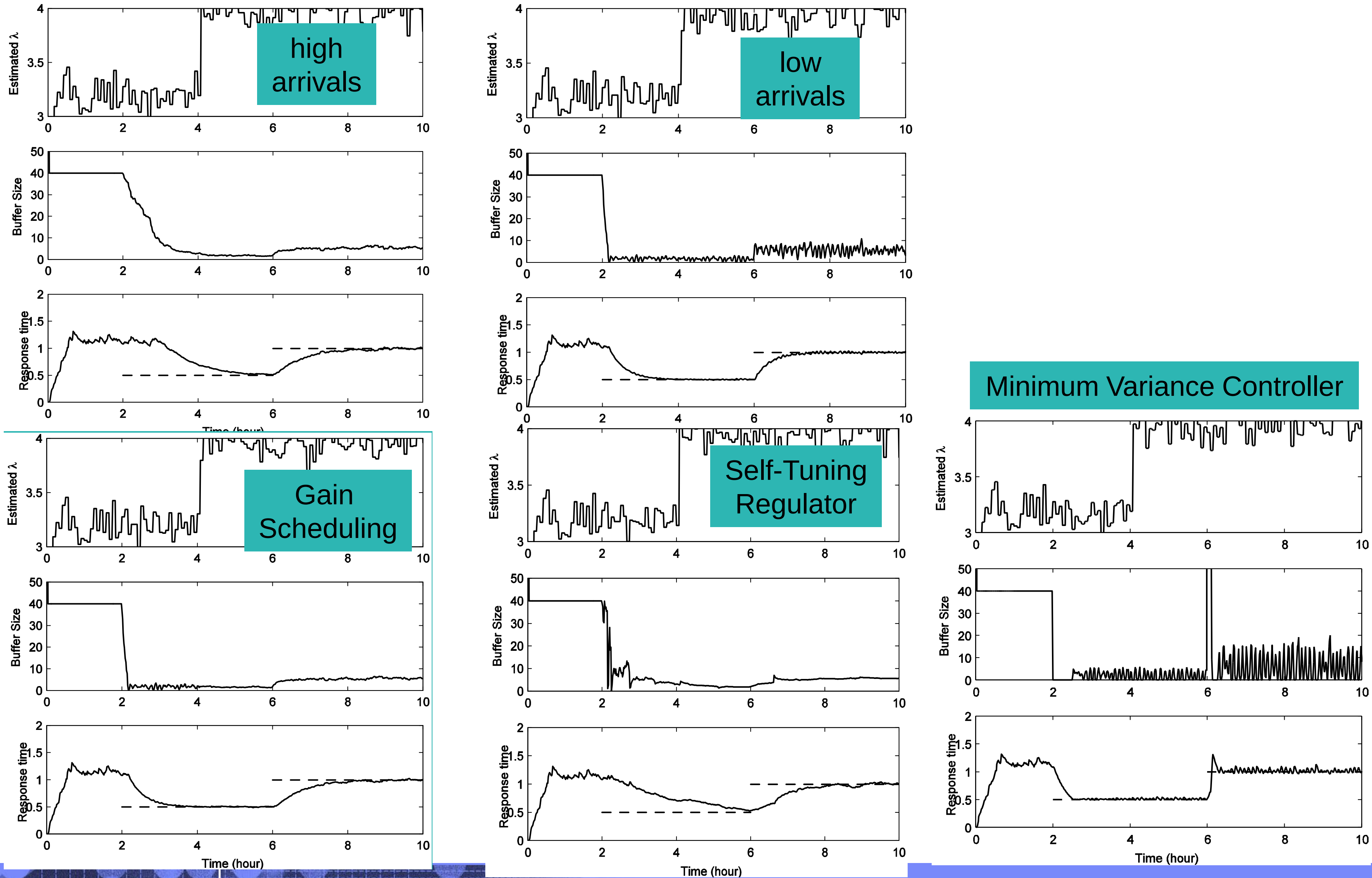
$$\begin{aligned}\operatorname{var}(y(k+1)) &= E(y(k+1) - E(y(k+1))^2 \\ &= E(ay(k) + bu(k) + w(k)) - r_{ss})^2 \\ &= E(ay(k) + bu(k) - r_{ss})^2 + \sigma_{ss}{}^2\end{aligned}$$

$$u(k) = \frac{r_{ss} - ay(k)}{b}, \text{ minimizes } \operatorname{var}(y(k+1))$$

Principles for developing minimum variance controllers

1. Express control gains in terms of the a_n and b_m
2. Use recursive least squares (or other techniques) to estimate a_n and b_m on-line
3. Compute new control gains whenever the a_n and b_m are updated

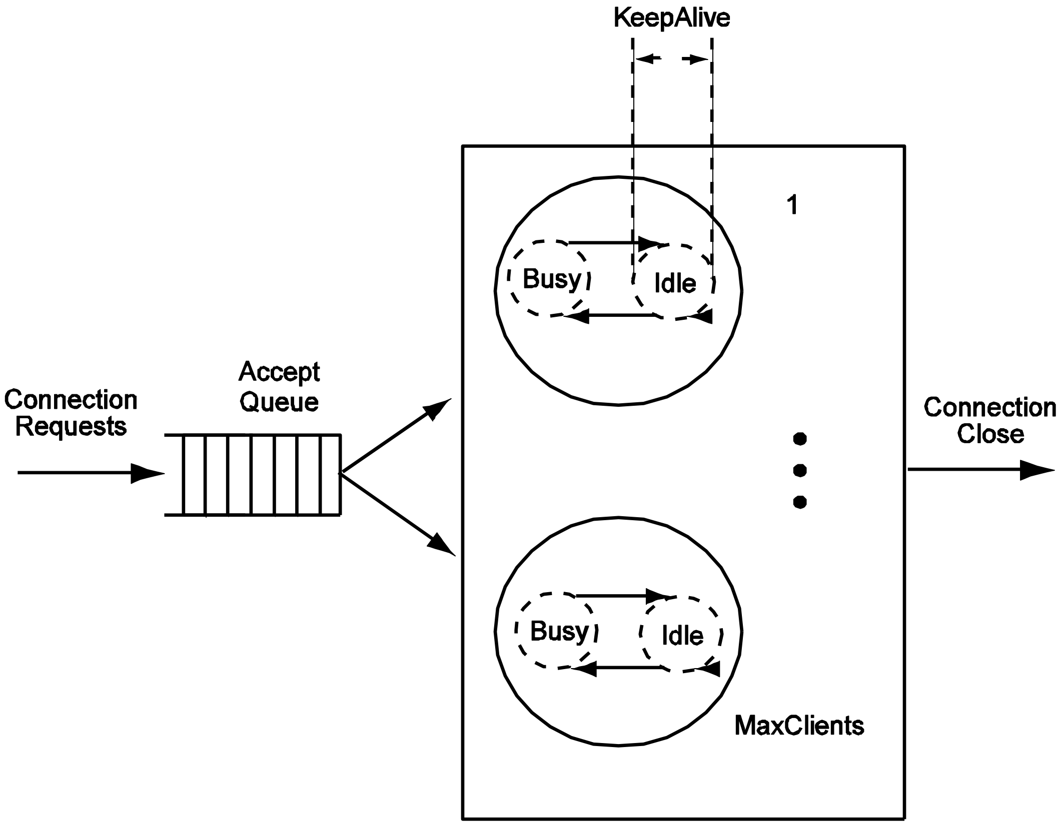
Performance of Minimum Variance Control

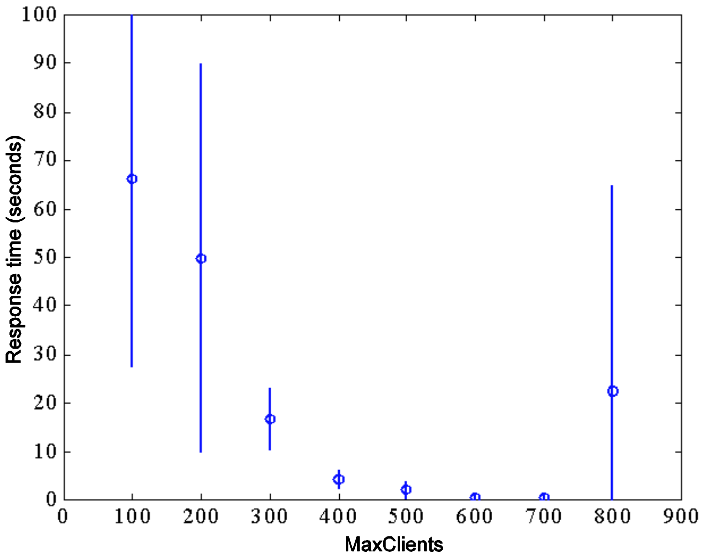


Fuzzy Control

- Simplified modeling target system
- Address optimization rather than regulation

The Control Problem

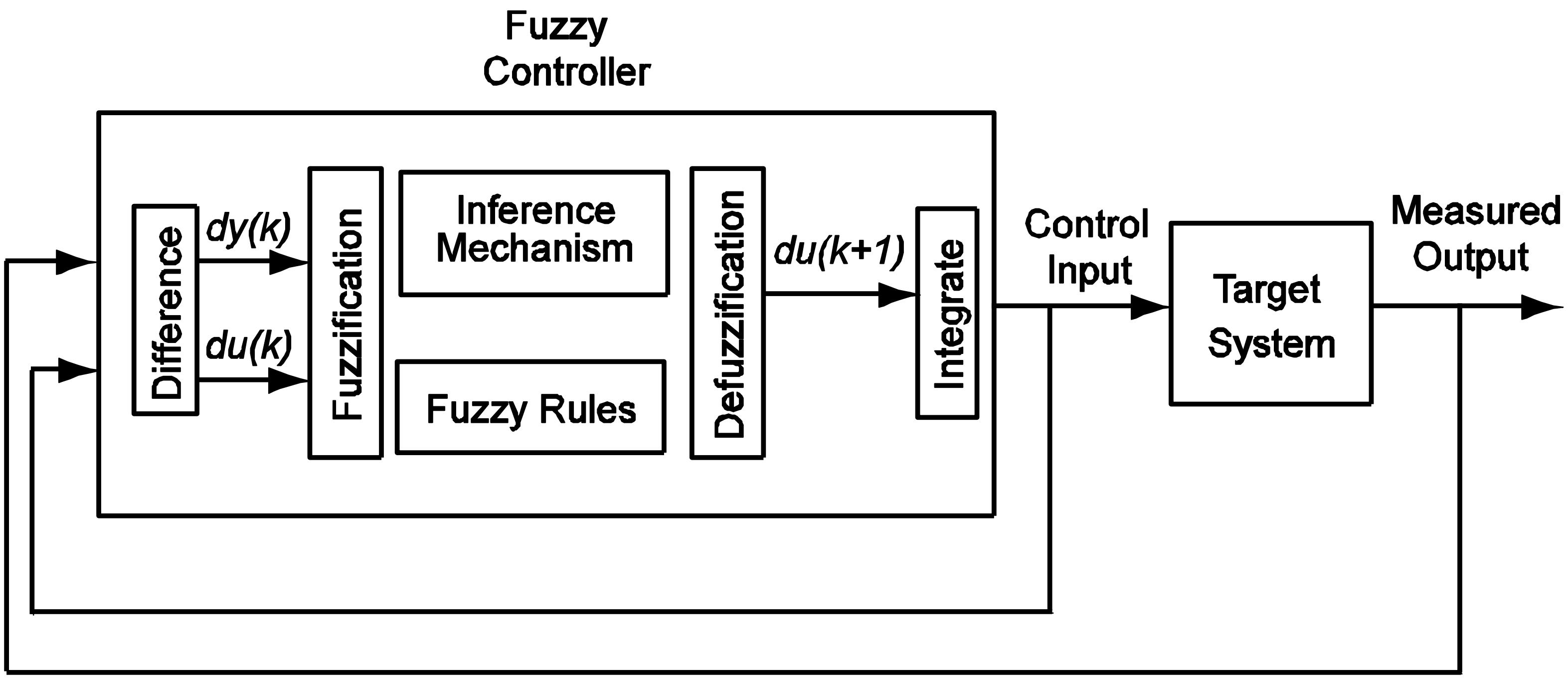




Control objective

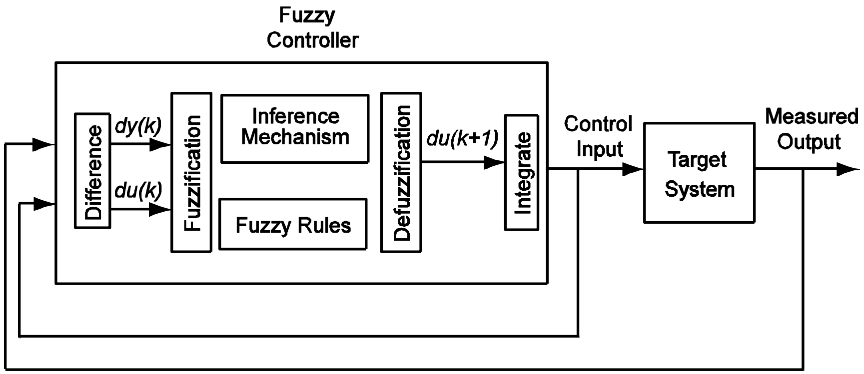
Select MaxClients that minimizes response time

Operation of Fuzzy Control



Difference: Compute change
Fuzzification: Translate into linguistic value
Inference mechanism: Interpret fuzzy rules
Fuzzy rules: Express how control input changes as other linguistic variables change
Defuzzification: Translate into quantitative values
Integrate: Compute non-differenced values

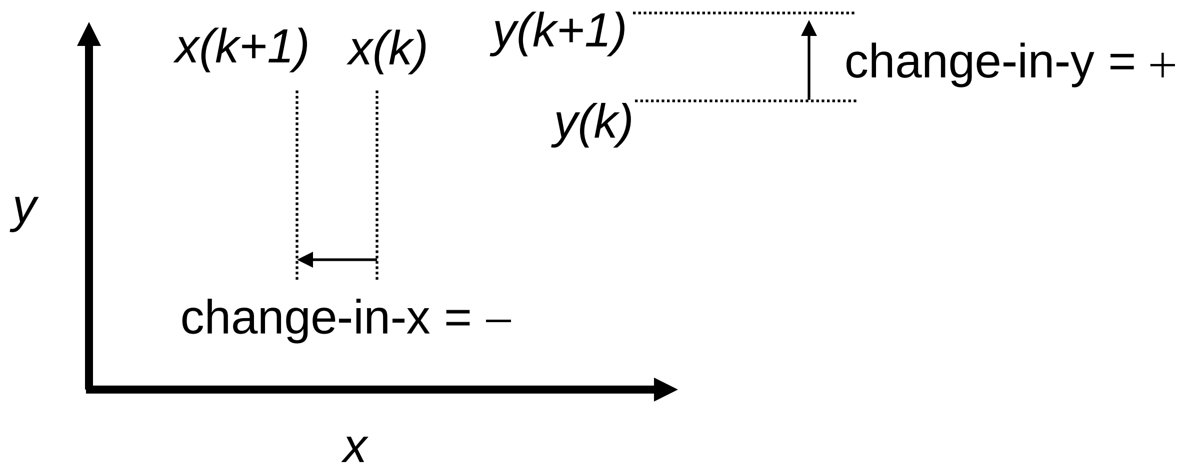
Linguistic Variables



Linguistic variables take on categorical values

Ex 1: positive, negative

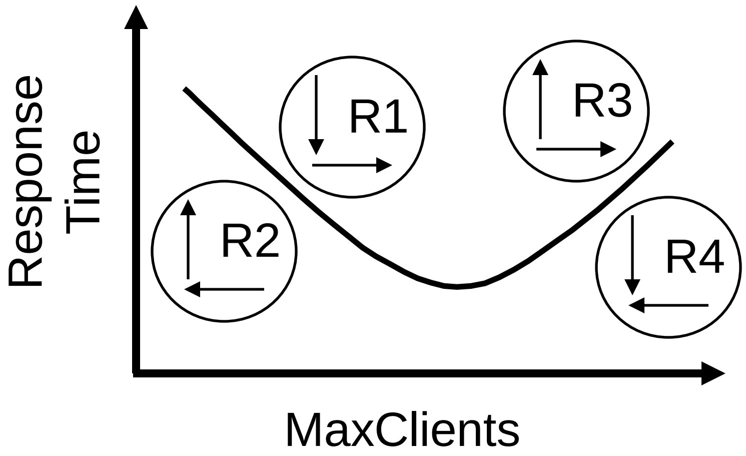
Ex 2: small, medium, large

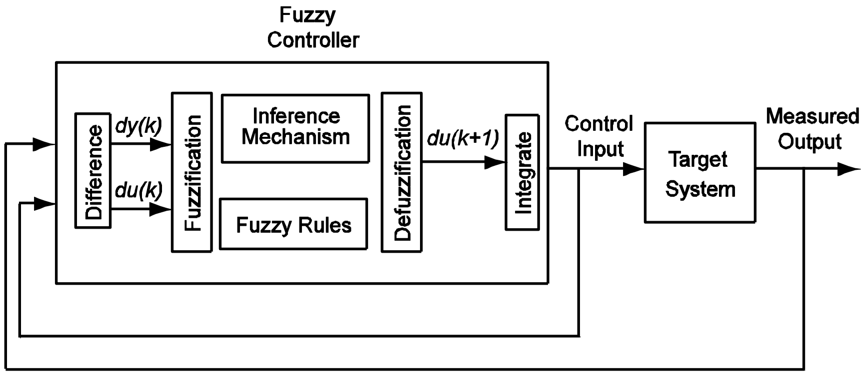


Linguistic variables provide a quality description of the system
Linguistic values are ordinal with a small number of distinct value

Ex 1: x(k)=45, x(k+1)=30 ➔ change-in-x = negative

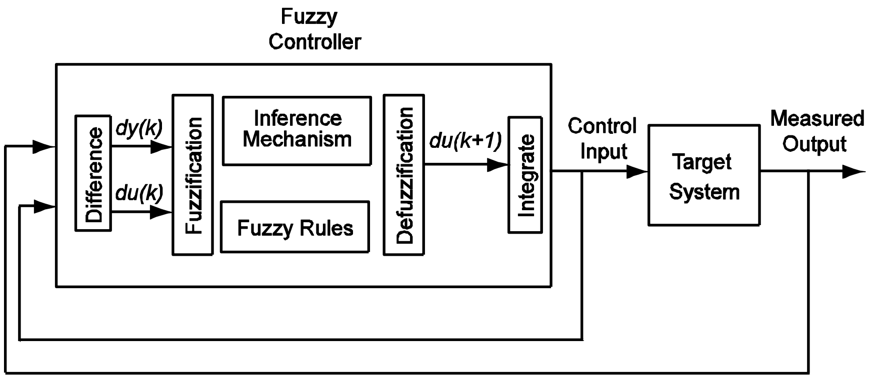
Fuzzy Rules

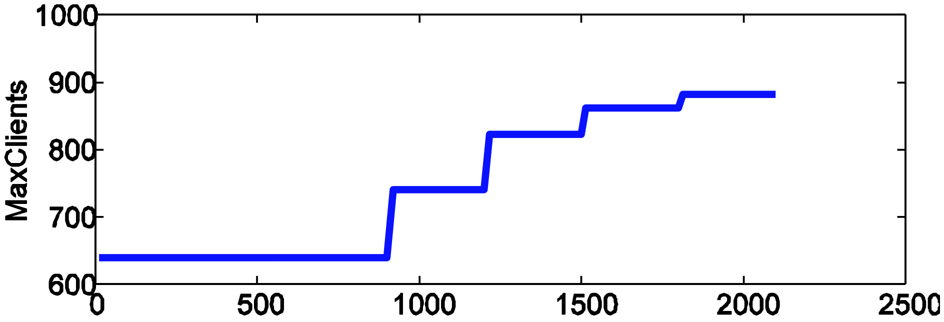


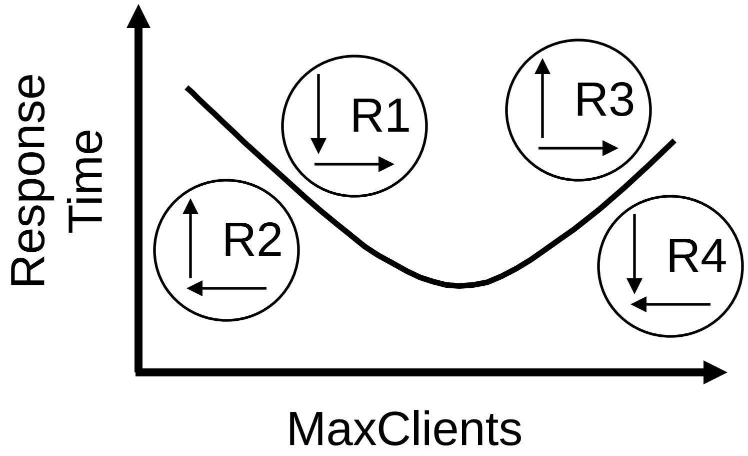


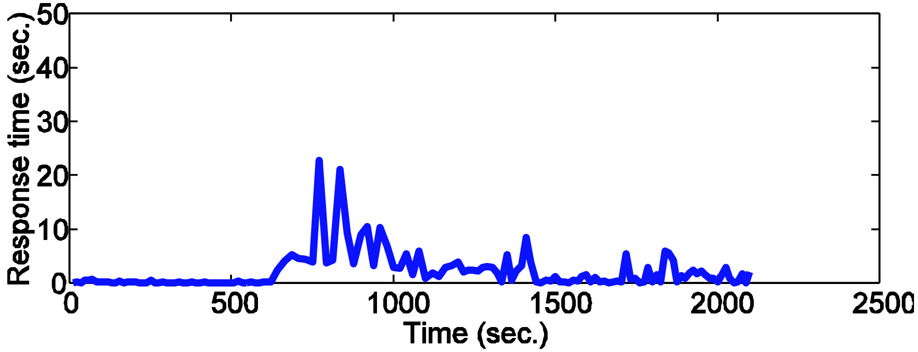
R1: If change-in-response-time is negative and last change-in-MaxClients was positive
Then increase MaxClients
R2: If change-in-response-time is positive and last change-in-MaxClients was negative
Then increase MaxClients
R3: If change-in-response-time is positive and last change-in-MaxClients was positive
Then decrease MaxClients
R4: If change-in-response-time is negative and last change-in-MaxClients was negative
Then decrease MaxClients

Operation of Fuzzy Control









Which rule(s) are being applied?

Summary

- Gain scheduling
  - Use scheduling variables to select among multiple controllers
- Self-tuning regulator
  - Continuous adjustment of controller gains based on updated model of target system
- Minimum variance controller
  - Control input compensates for variability
- Fuzzy control
  - Use rules to optimize performance