CS321 - Autonomic Computer Systems

Introduction to Control Theory 2008-10-02

Christian Tschudin, University of Basel

Using slides from:

J. L. Hellerstein, IBM
Z. Gao, Cleveland State University
C. Lu, University of Virginia

Overview

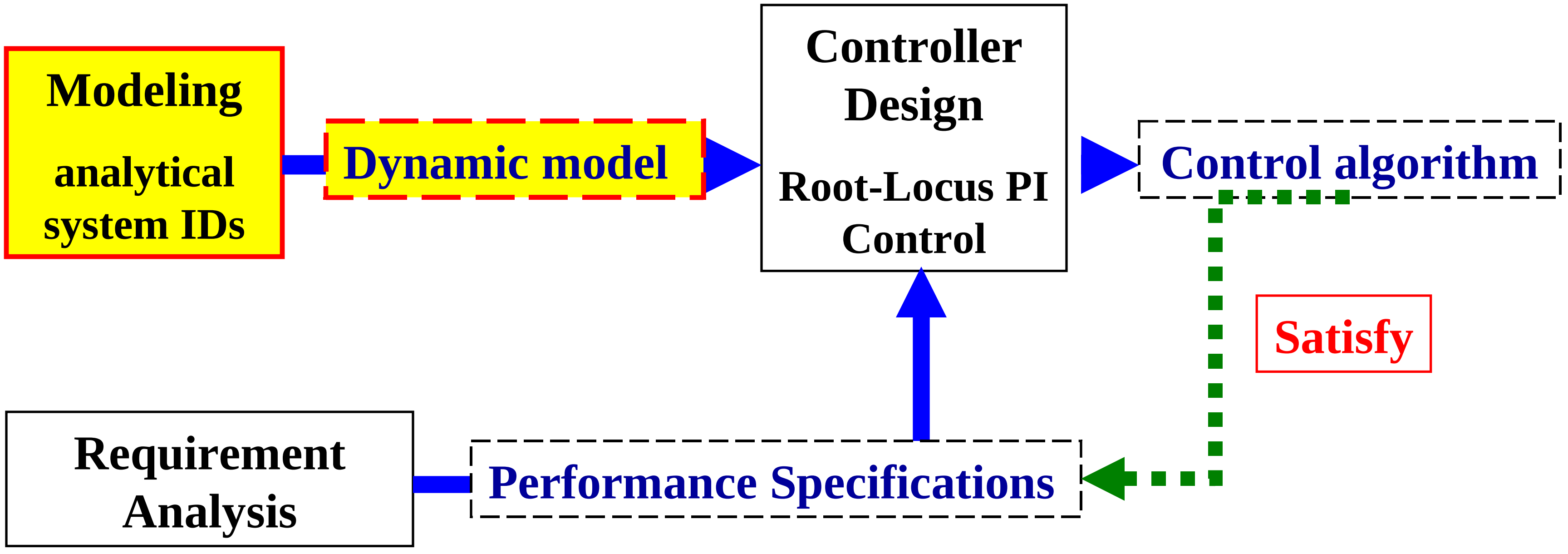
- Performance specifications (for expressing policies)
- Digital controlers
- z-Transform:
  - concept, signal composition
  - poles and stability
  - transfer function and system composition
  - proportional (P) and integral (I) control, PI control
- Effector example for a software system

Feedback control theory, vs …

- Adaptive resource management heuristics
  - Laborious design/tuning/testing iterations
  - Not enough confidence in face of untested workload
- Queuing theory
  - Doesn’t handle feedbacks
  - Not good at characterizing transient behavior in overload

- Feedback control theory
  - Systematic theoretical approach for analysis and design
  - **Predict system response and stability to input**

Control design methodology



System Models

- **System ID** vs. First Principle

Feedback Control of Computing Systems
M1: Introduction

Joseph L. Hellerstein
IBM Thomas J Watson Research Center, NY
hellers@us.ibm.com

September 21, 2004

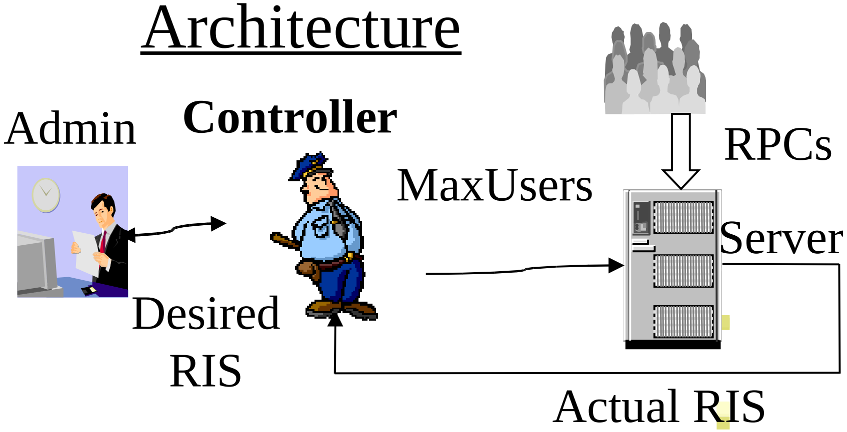
Book reference:
"Feedback Control of Computing Systems", Hellerstein, Diao, Parekh, Tilbury, Wiley, 2004.
Slides: http://swig.stanford.edu/~fox/cs444a/ct_tutorial/

Overview

- Open loop, closed loop
- Control system architecture
- Performance specifications (for expressing policies)
- Digital controlers
- z-Transform:
  - concept, signal composition
  - poles and stability
  - transfer function and system composition
  - proportional (P) and integral (I) control, PI control
- Effector example for a software system

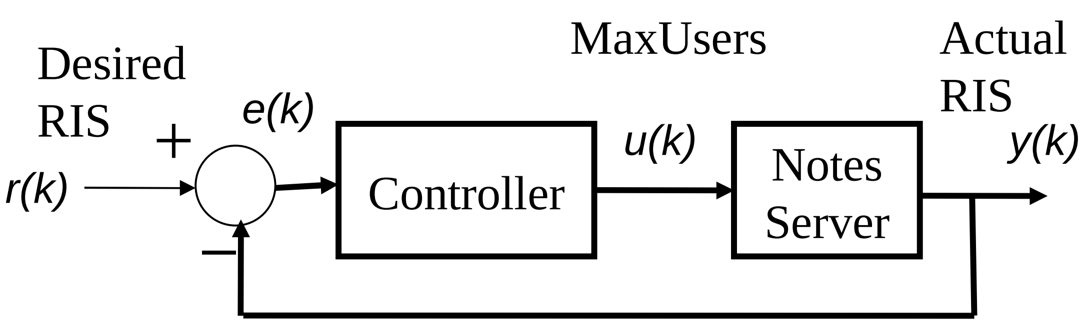
Example: Control of Lotus Notes

Architecture



RIS = RPCs in System

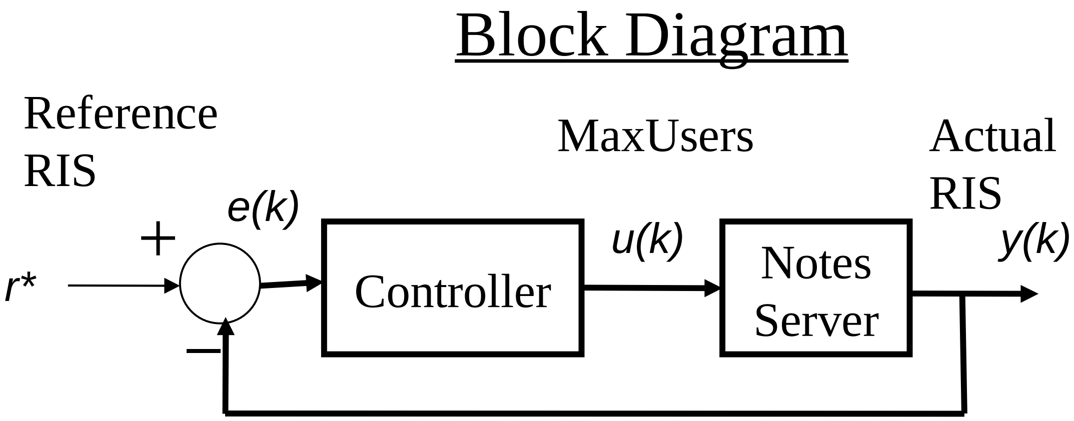
Control Model



Control error: $e(k)=r(k)-y(k)$

System model: $y(k)=(0.43)y(k-1) +(0.47)u(k-1)$

Lab Exercise



ARX* Models

Control error: $e(k)=r^*-y(k)$

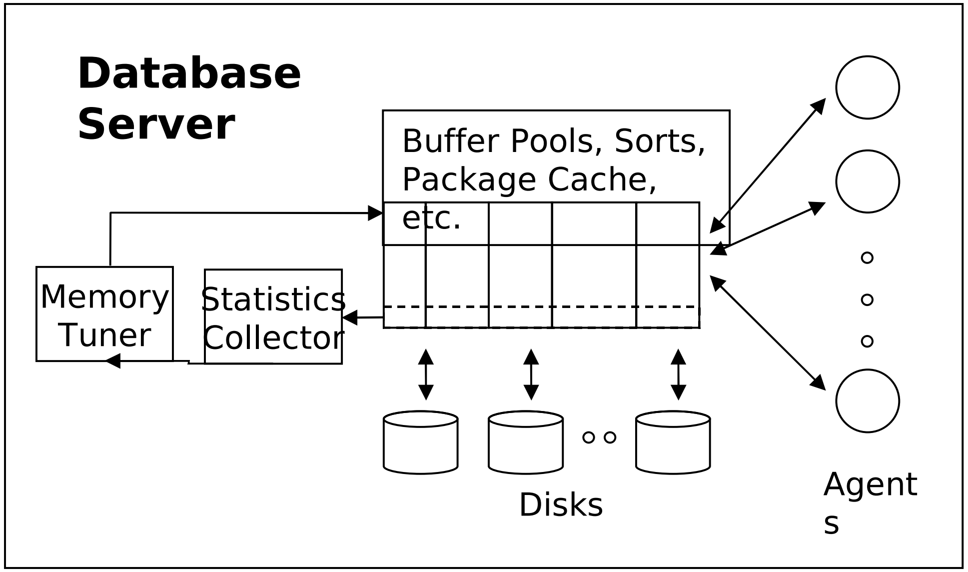
P controller: $u(k)=Ke(k)$

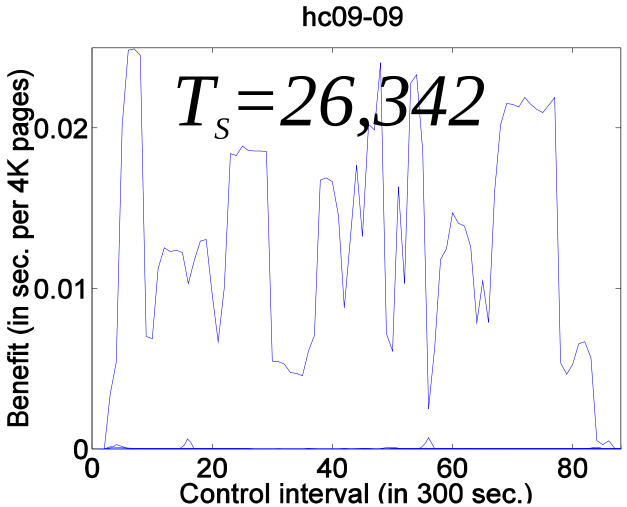
System model: $y(k)=(0.43)y(k-1)$
$+(0.47)u(k-1)$

- Spreadsheet file ctclass-M1
  - ***Proportional*** controller: *Maxusers(k+1) = K(r*--y(k))*
  - What is the effect of *K* on
    - Accuracy: (want $r^*=y(k)$=200)
    - Stability
    - Convergence rate
    - Overshoot

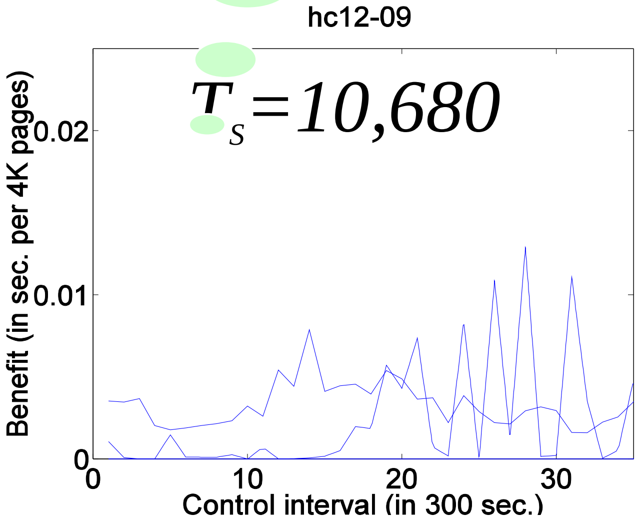
*ARX is autoregressive with an external input

Application of CT to a DBMS





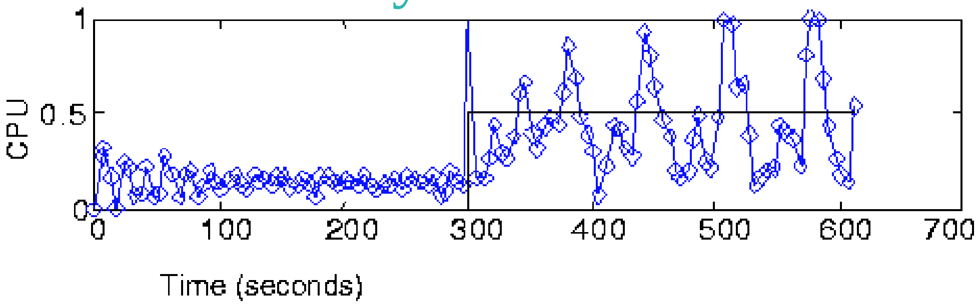
Without Controller

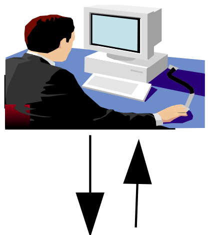


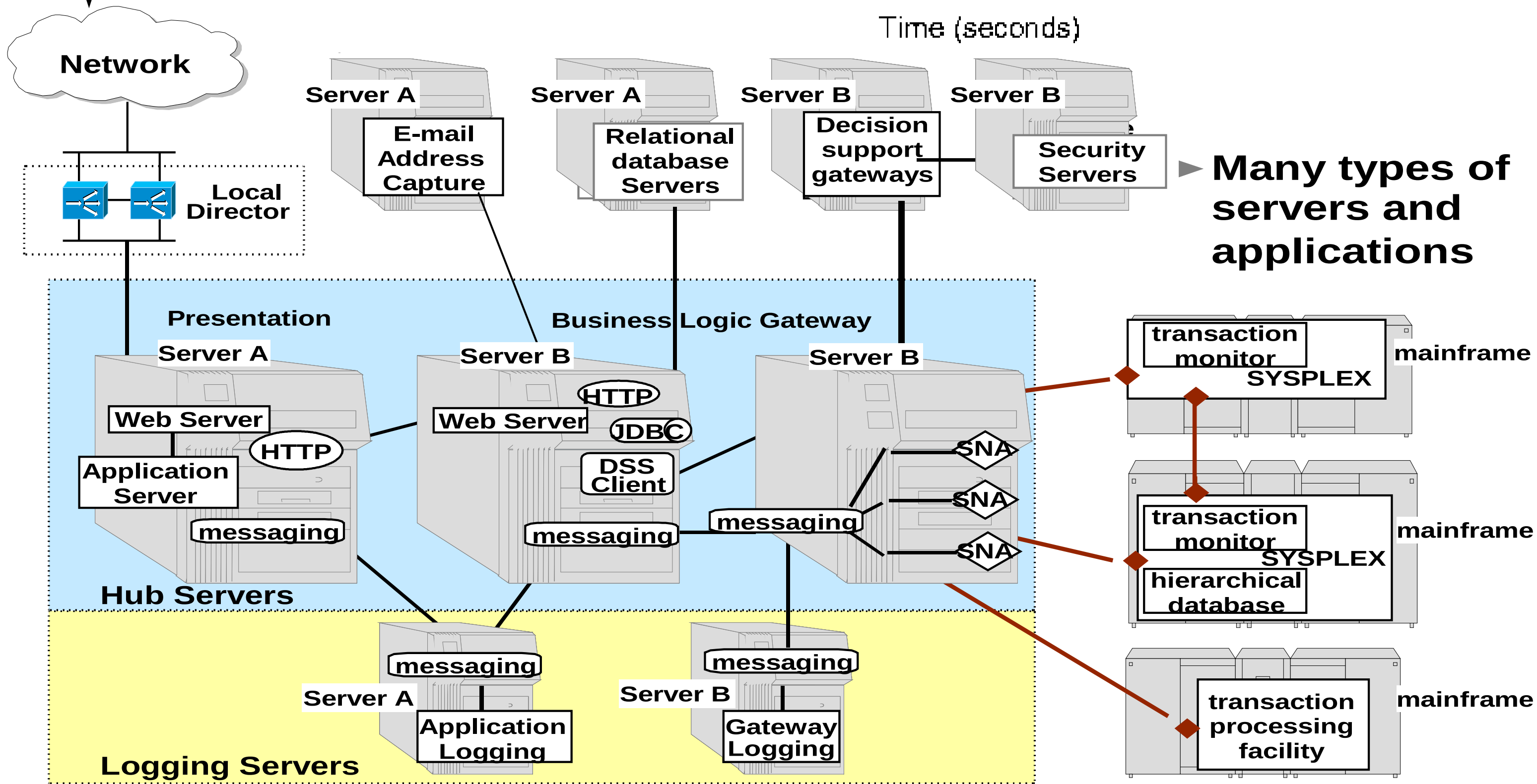
With Controller

Why Control Theory?

Unstable System





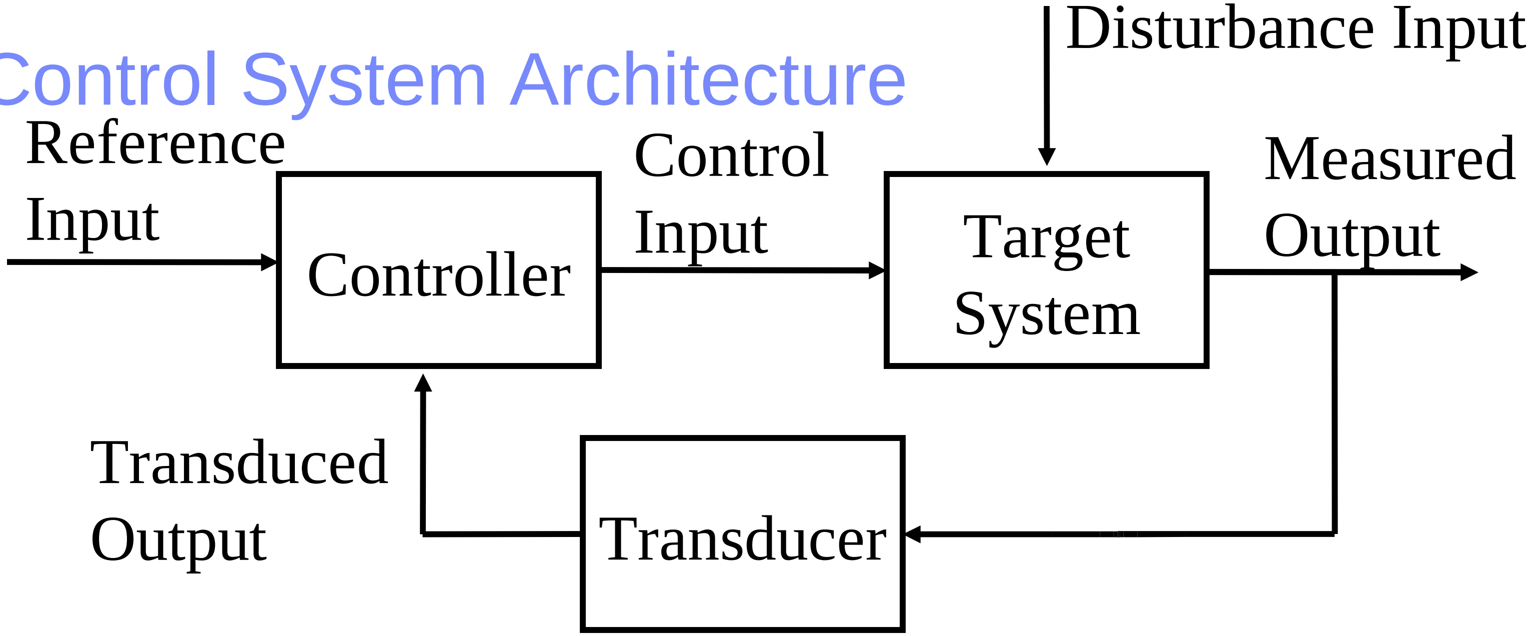


Front end for online customer service

Back-end Systems

•Stabililty •Accuracy •Settling time •Overshoot

Control System Architecture



Components

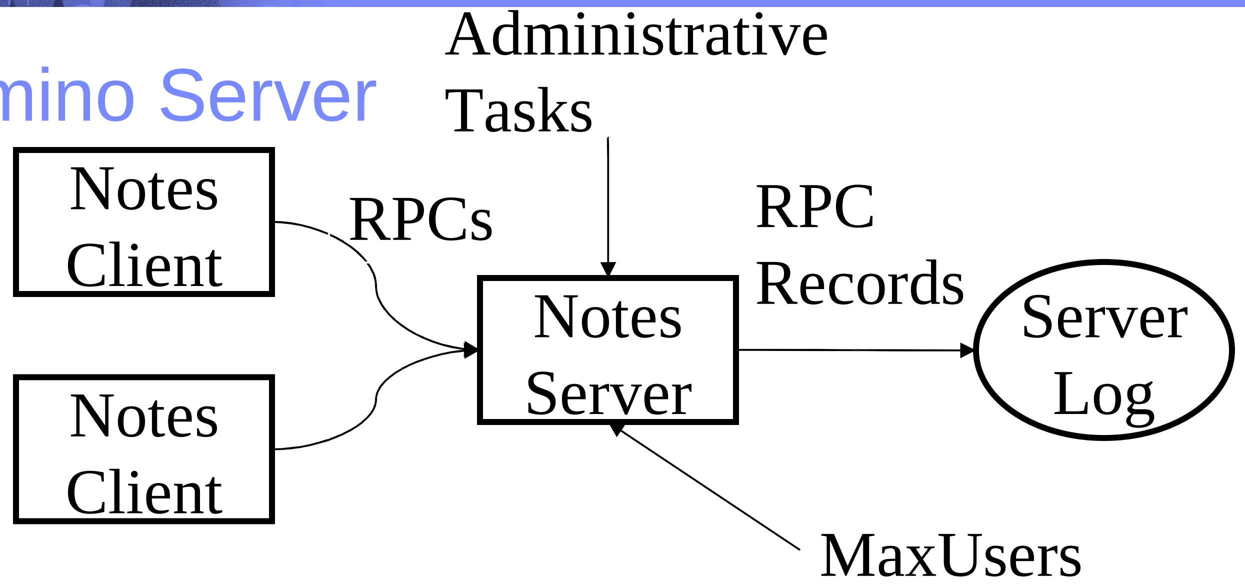
- Target system: what is controlled
- Controller: exercises control
- Transducer: translates measured outputs

Data

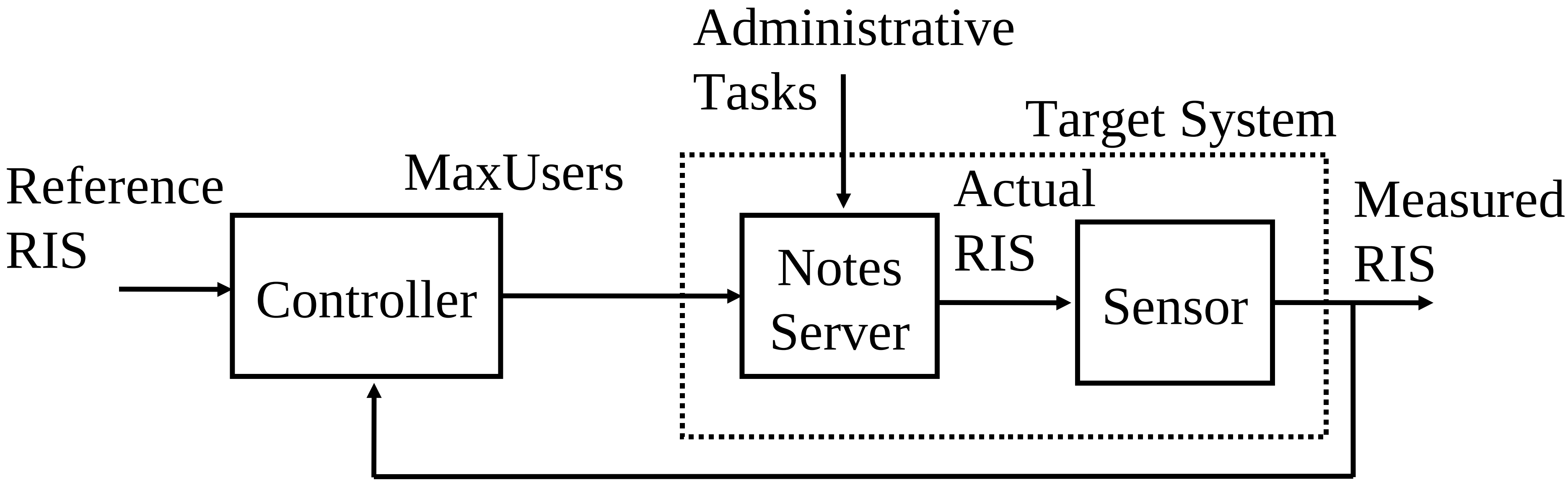
- Reference input: objective
- Control input: manipulated to affect output
- Disturbance input: other factors that affect the target system
- Transduced output: result of manipulation

Given target system, transducer
Control theory finds controller
that adjusts control input
to achieve measured
output in the presence of
disturbances.

IBM Lotus Domino Server



Architecture



Block Diagram

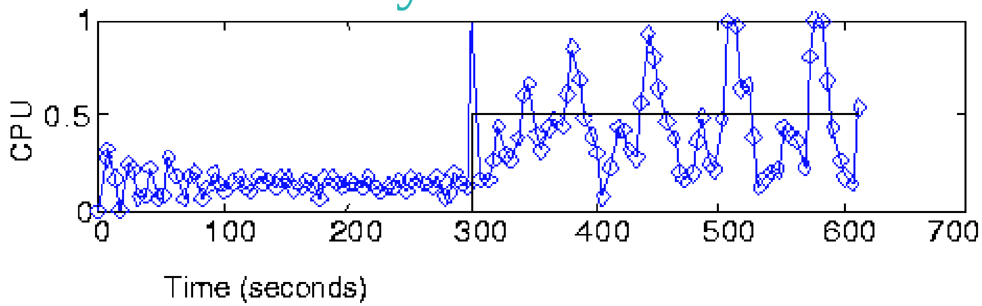
Overview

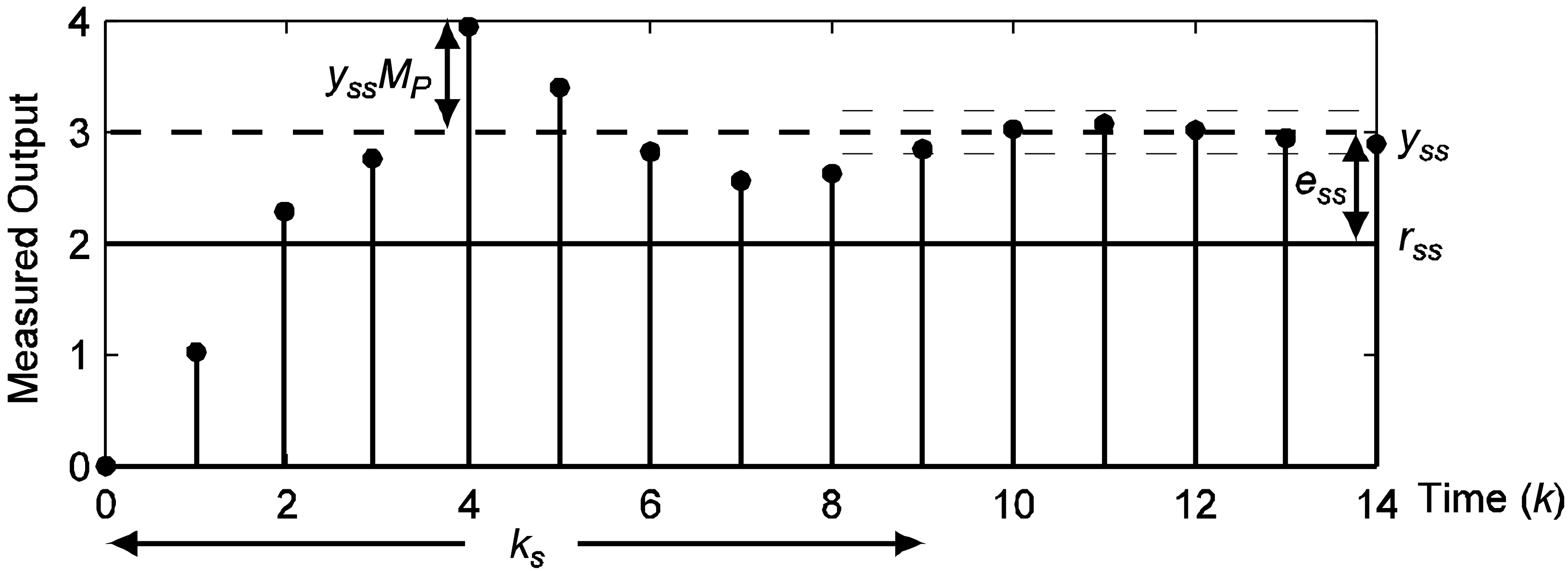
- Open loop, closed loop
- Control system architecture
- Performance specifications (for expressing policies)
- Digital controlers
- z-Transform:
  - concept, signal composition
  - poles and stability
  - transfer function and system composition
  - proportional (P) and integral (I) control, PI control
- Effector example for a software system

Properties of Control Systems

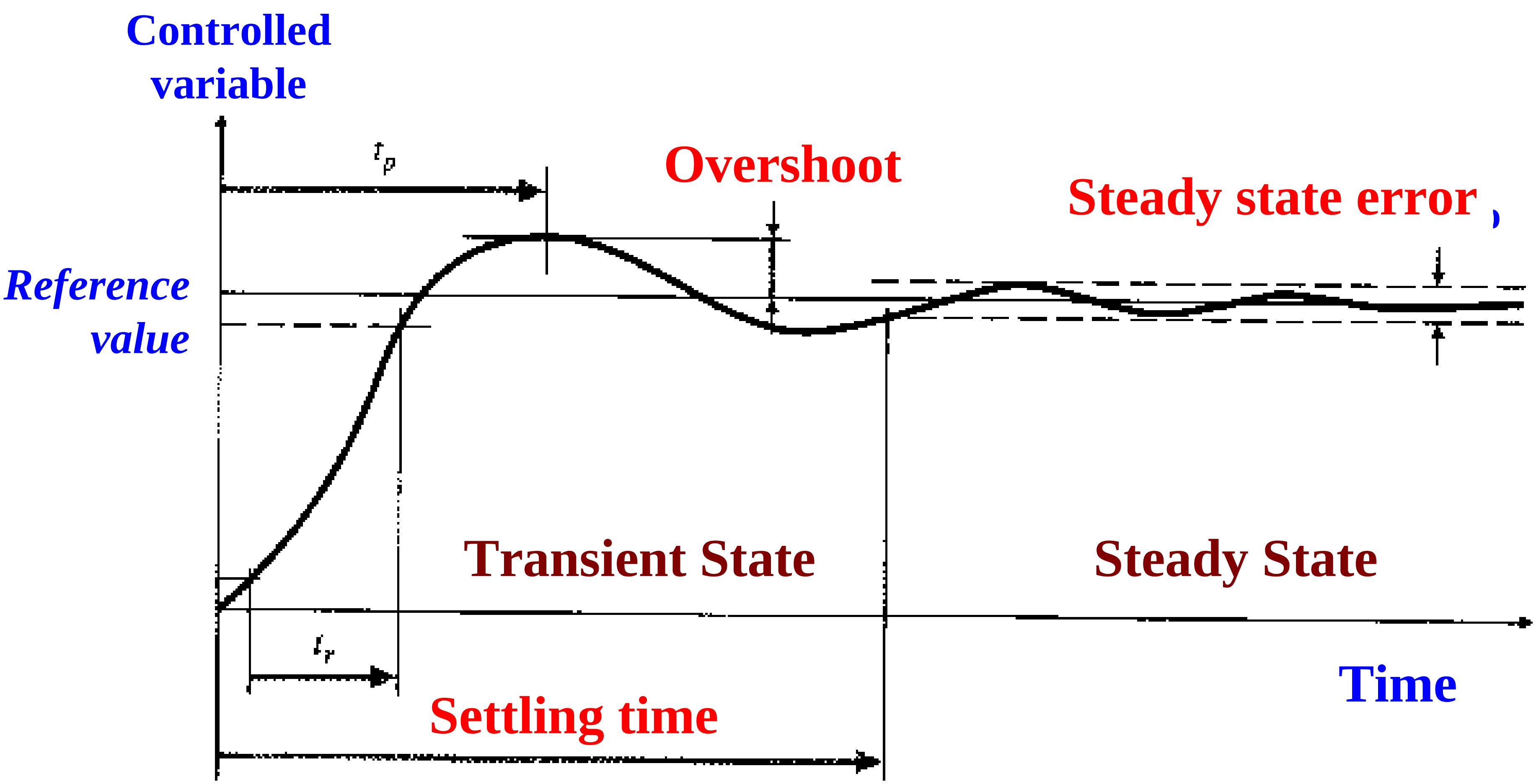
Stability　　Accuracy　　Short settling　　Small overshoot

Unstable System

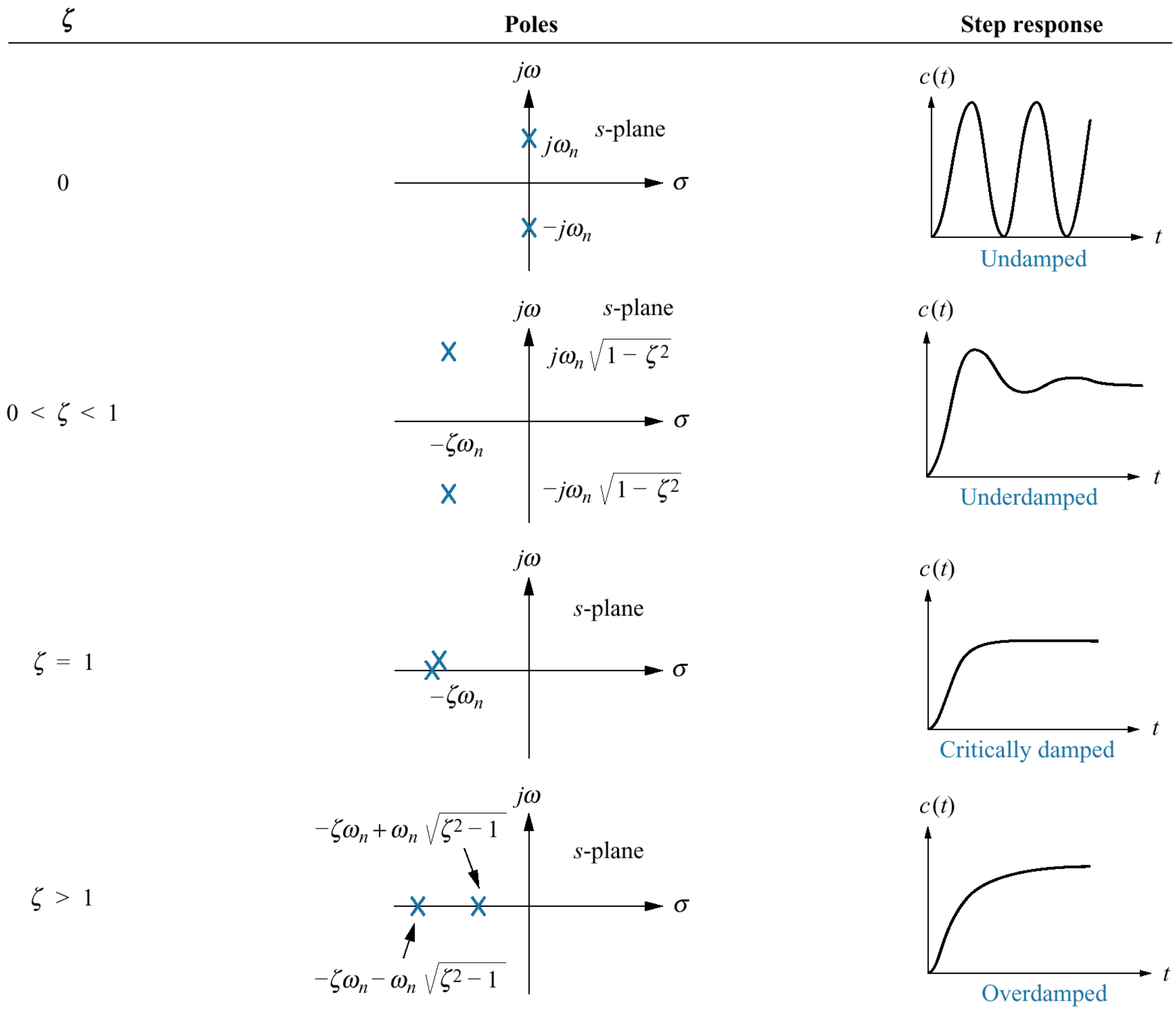




Performance specifications



Different "Responses" (Pure 2nd Order Transfer Functions)



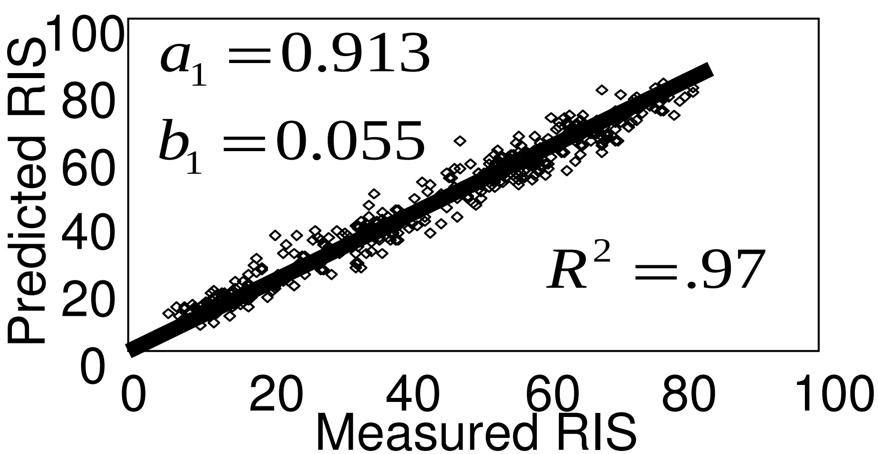
Control Theory in Two Slides: System Identification

MaxUsers $\rightarrow$ **Notes Server** $\rightarrow$ Actual RIS

$$u(k) \qquad\qquad y(k)$$

Model of System Dynamics

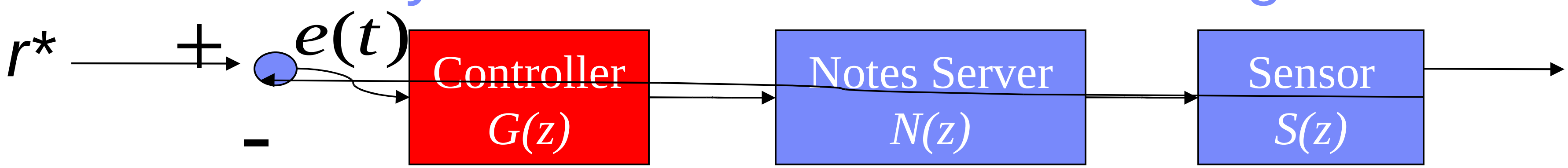
$$y(k) = \boxed{a_1} y(k-1) + \boxed{b_1} u(k-1)$$



Transfer Function

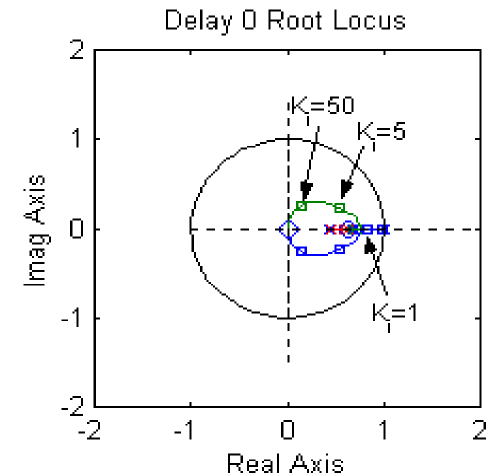
$$N(z) = \frac{b_1}{z - a_1}$$

Control Theory in Two Slides: Control Design



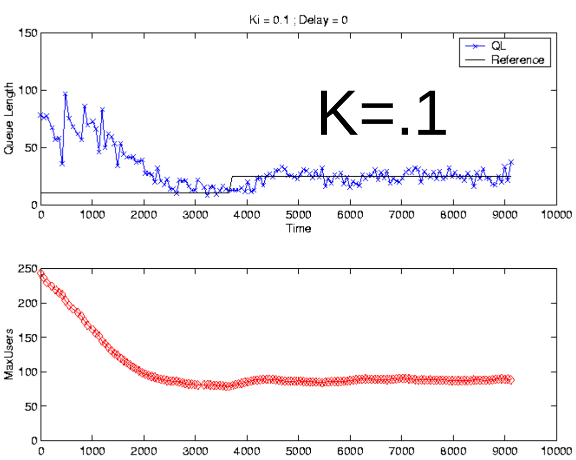
H(z) = Closed Loop Transfer Function

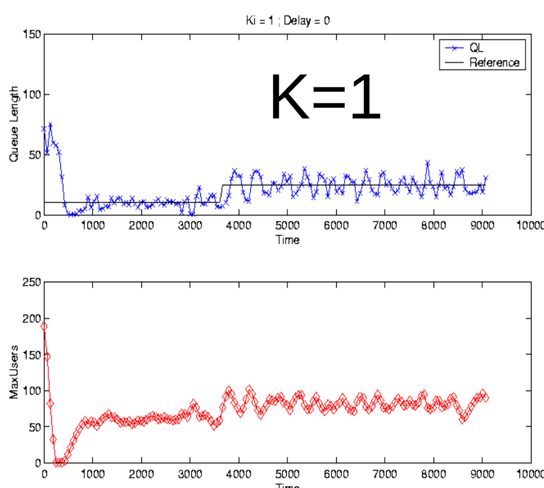
Poles of *H(z)*

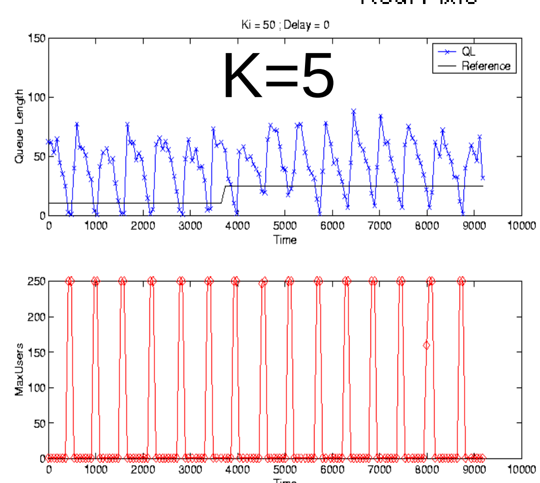


<u>Integral Control Law</u>

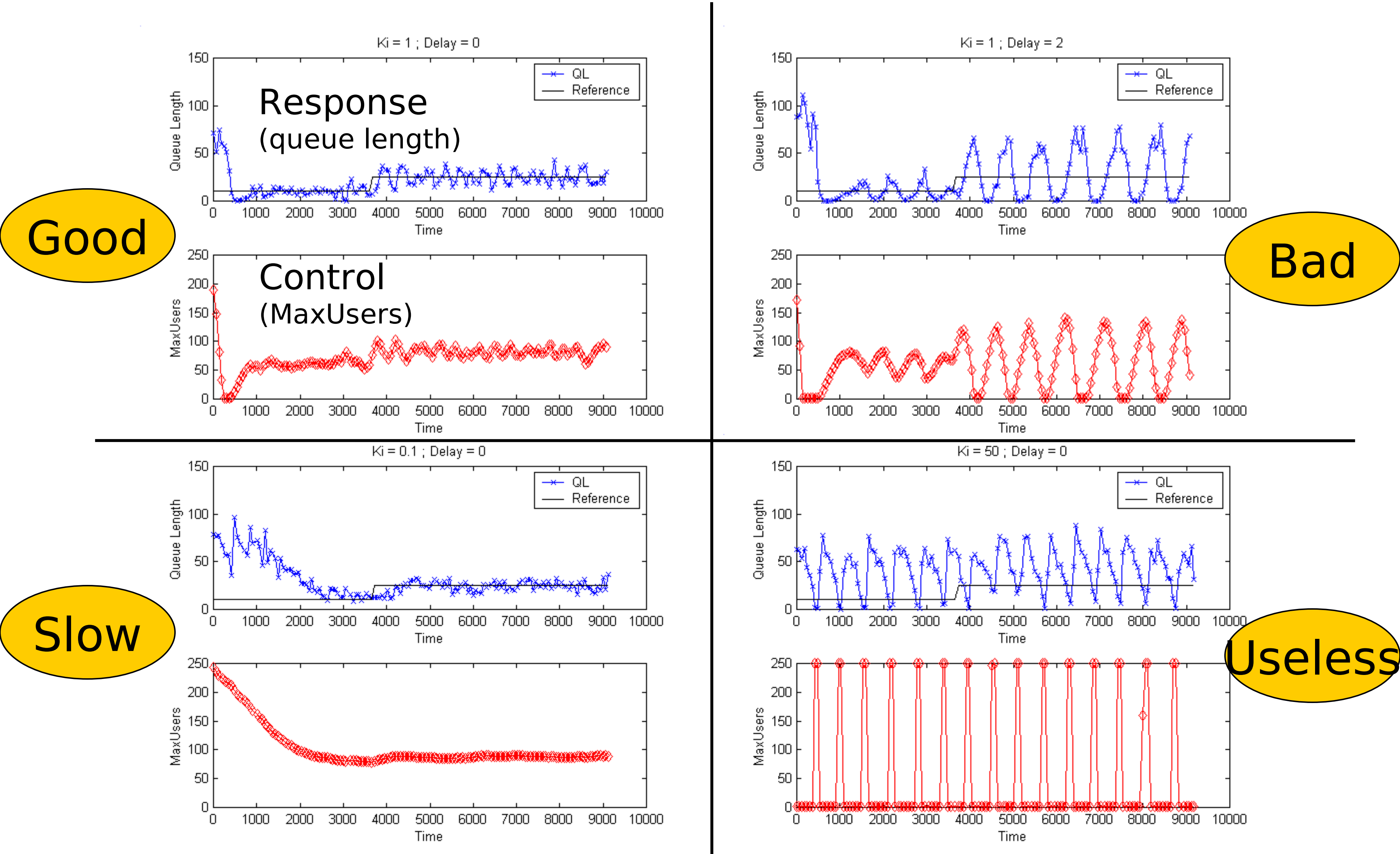
$$u(t) = u(t-1) + Ke(t)$$



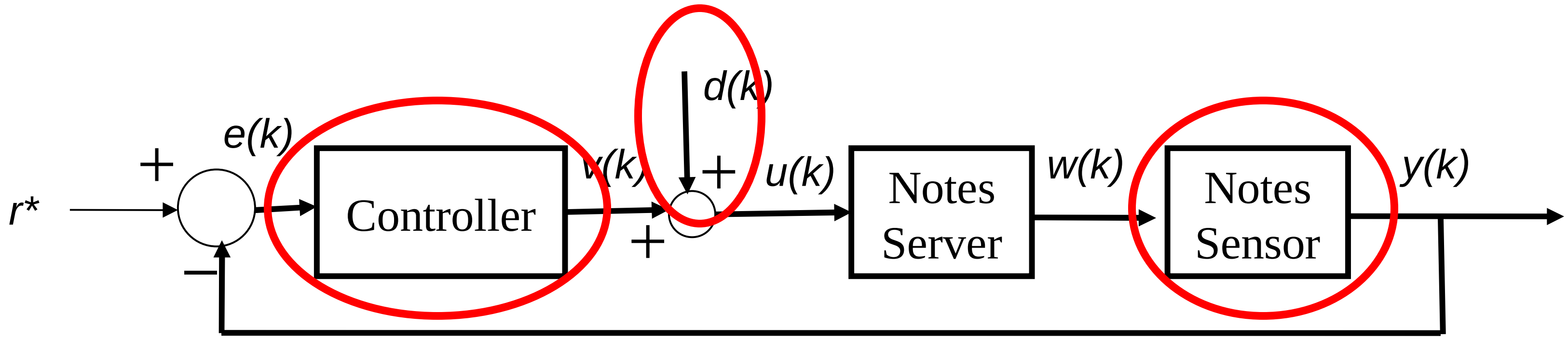




Example: Control & Response in an Email Server



Lab 2: Modifications to the Lotus Notes Simulation



Control error: $e(k)= r^*- y(k)$

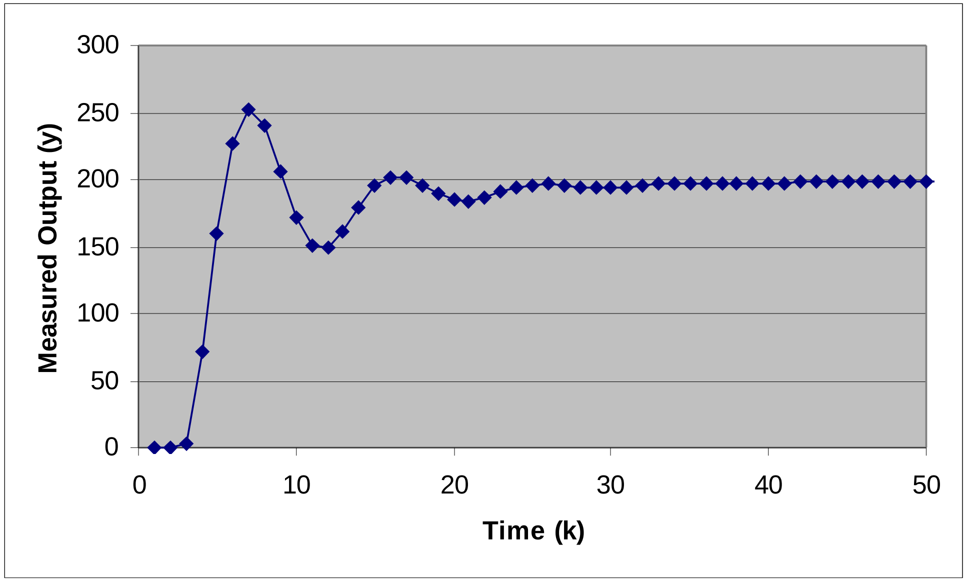
I controller: $v(k)=v(k-1)+Ke(k)$

System model: $w(k)=(0.43)w(k-1)$
$+(0.47)u(k-1)$

Disturbance input: $u(k) = v(k) + d(k)$

Sensor: $y(k)=(0.8)y(k-1)$
$+0.72w(k)-0.66w(k-1)$

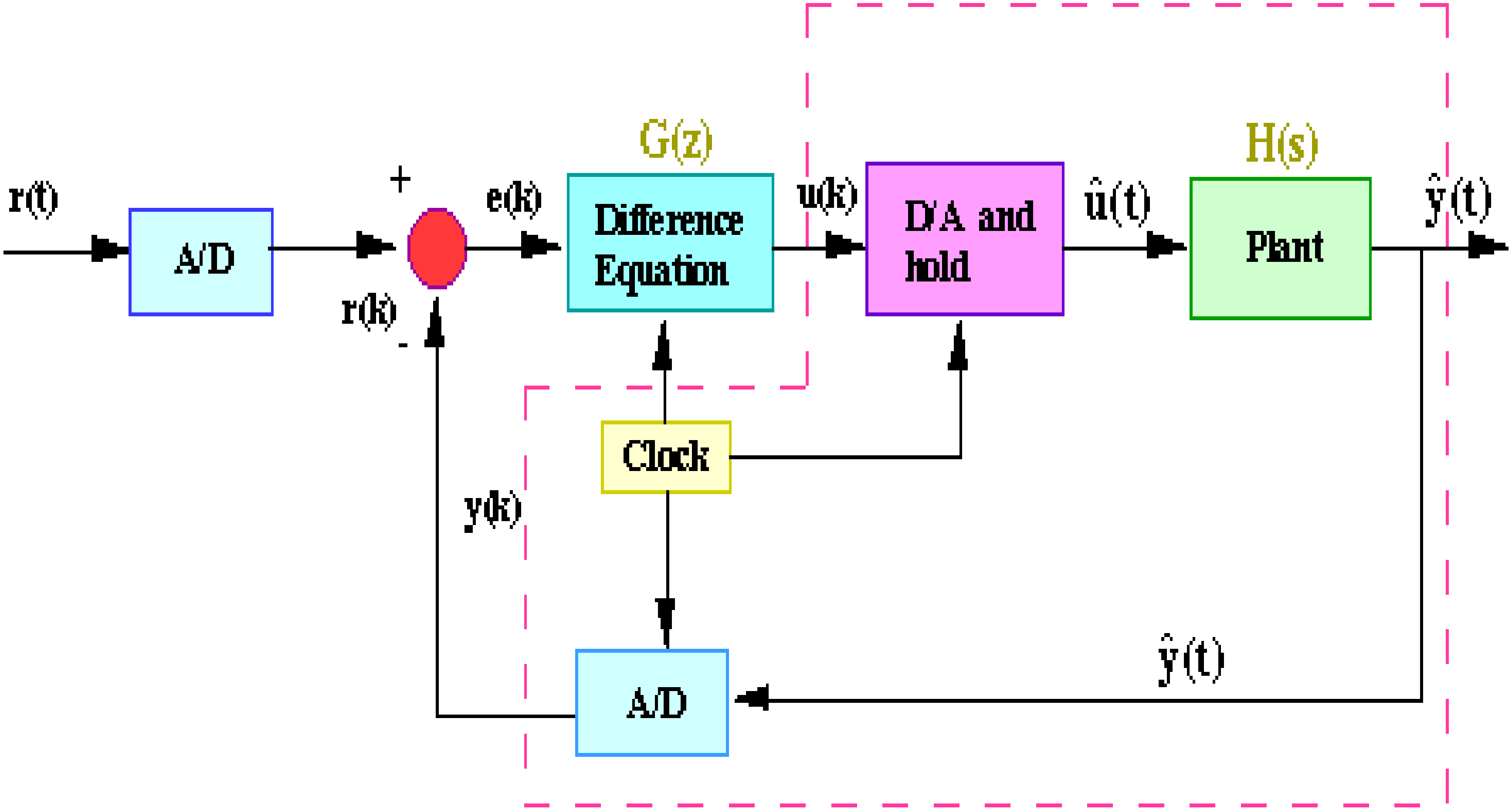
$K=1, r(k)=200, d(k)=10$



Overview

- Open loop, closed loop
- Control system architecture
- Performance specifications (for expressing policies)
- Digital controlers
- z-Transform:
  - concept, signal composition
  - poles and stability
  - transfer function and system composition
  - proportional (P) and integral (I) control, PI control
- Effector example for a software system

Digital Control



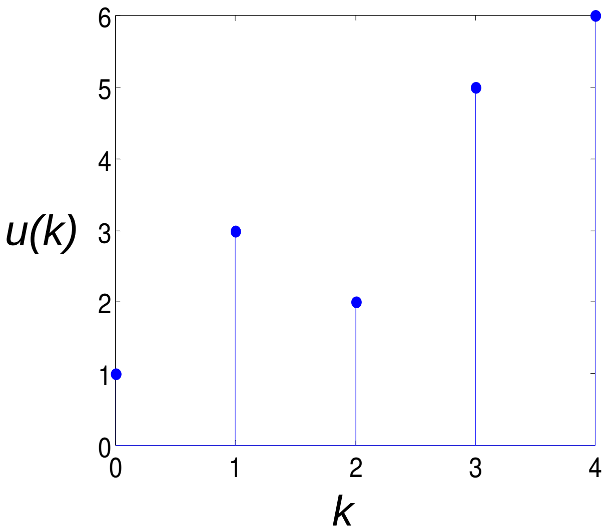
Digital controler

- Discrete time
- Discrete values
- Usually: continuous time, continous values
  → differential equations
  → use of Laplace transform
- With digital controler:
  → difference equations
  → use of z-Transform

Overview

- Open loop, closed loop
- Control system architecture
- Performance specifications (for expressing policies)
- Digital controlers
- z-Transform:
  - concept, signal composition
  - poles and stability
  - transfer function and system composition
  - proportional (P) and integral (I) control, PI control
- Effector example for a software system

Z-Transform of a Signal



<u>**Time domain representation**</u>	<u>***z*** **domain representation**</u>
$u(0)=1$	$1z^{0}$ +
$u(1)=3$	$3z^{-1}$ +
$u(2)=2$	$2z^{-2}$ +
$u(3)=5$	$5z^{-3}$ +
$u(4)=6$	$6z^{-4}$

z is time shift; z^{-1} is time delay ⟹

$$z^{0} = 1\text{: } k = 0 \text{ (current time)}$$

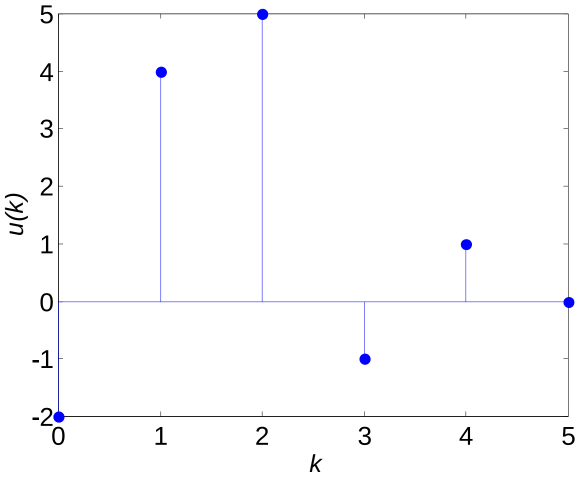
$$z^{-1}\text{: } k = 1 \text{ (one time unit in the future)}$$

$$z^{-2}\text{: } k = 2 \text{ (two time units in the future)}$$

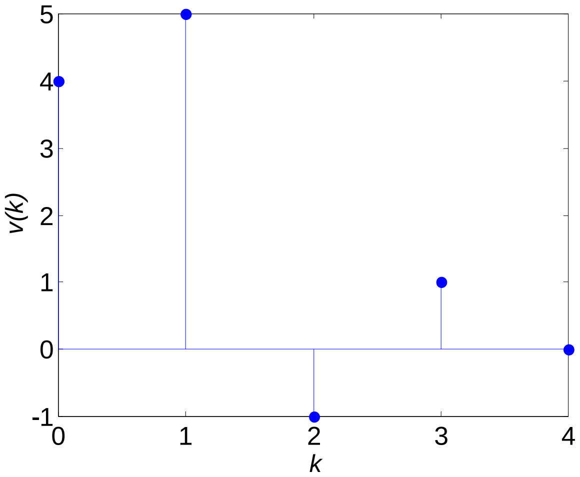
If $\{u(k)\} = u(0), u(1), \ldots$ is a signal, then its z-Transform is

$$U(z) = ¥_{k=0} u(k) z^{-k}$$

Write the Z-Transforms or Plots for the Following Finite Signals



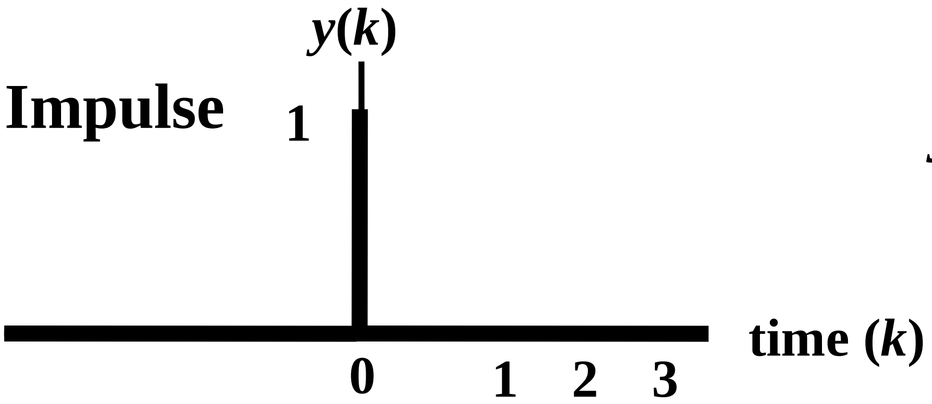
$$U(z) = -2 + 4z^{-1} + 5z^{-2} - z^{-3} + 4z^{-4}$$



$$V(z) = zU(z) = 4 + 5z^{-1} - z^{-2} + z^{-3}$$

(Drop exponents >0.)

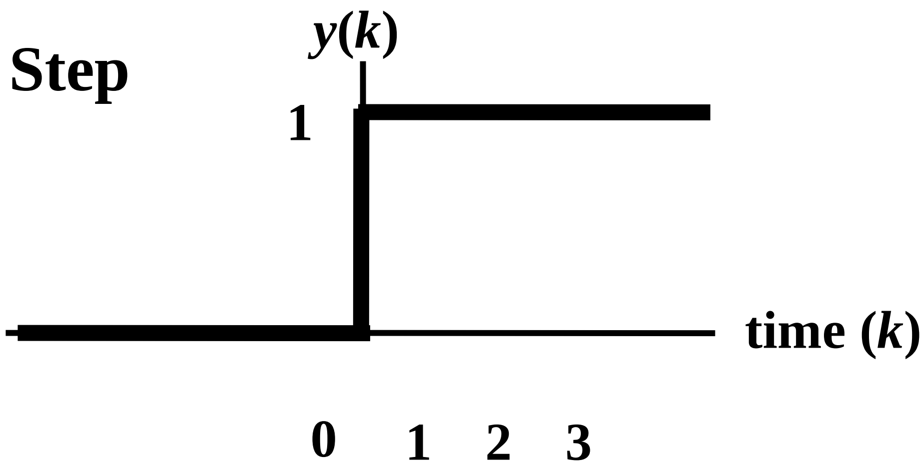
Common Signals: Impulse



$$y(0)=1; y(k)=0, k>0$$
$$Y(z)=1z^{0}+0z^{-1}+0z^{-2}+\ldots$$
$$=1$$

$$\text{Example}: 5z^{-3} \Leftrightarrow u(3)=5$$

Common Signals: Unit Step W(z)



$$y(k) = 1, k \geq 0$$

$$Y(z) = 1z^{0} + 1z^{-1} + 1z^{-2} + \dots$$

$$= \frac{1}{1 - z^{-1}}$$

$$= \frac{z}{z - 1}$$

Properties of z-Transforms of Signals

Signals: $U(z) = u(0)z^0 + u(1)z^{-1} + u(2)z^{-2} + \ldots$

$$V(z) = v(0)z^0 + v(1)z^{-1} + v(2)z^{-2} + \ldots$$

Shift: $zU(z) = u(0)z^1 + u(1)z^0 + u(2)z^{-1} + \ldots$

$$= u(1)z^0 + u(2)z^{-1} + \ldots$$

Delay: $U(z)/z = u(0)z^{-1} + u(1)z^{-2} + u(2)z^{-3} + \ldots$

Scaling: $aU(z) = au(0)z^0 + au(1)z^{-1} + au(2)z^{-2} + \ldots$

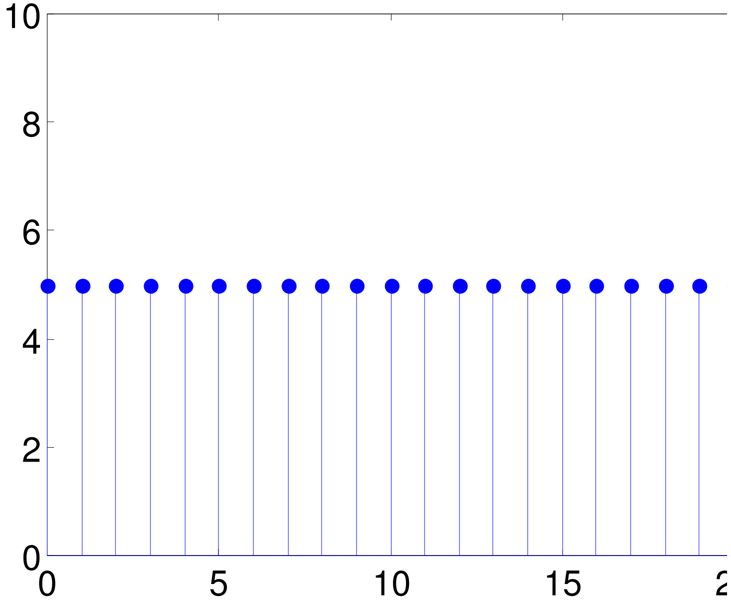
$$= z\text{-Transform of } \{au(k)\}$$

Sum of signals: $u(0)z^0 + u(1)z^{-1} + u(2)z^{-2} + \ldots + v(0)z^0 + v(1)z^{-1} + v(2)z^{-2} + \ldots$

$$= (u(0)+v(0))z^0 + (u(1)+v(1))z^{-1} + (u(2)+v(2))z^{-2} + \ldots$$

$$= U(z) + V(z)$$

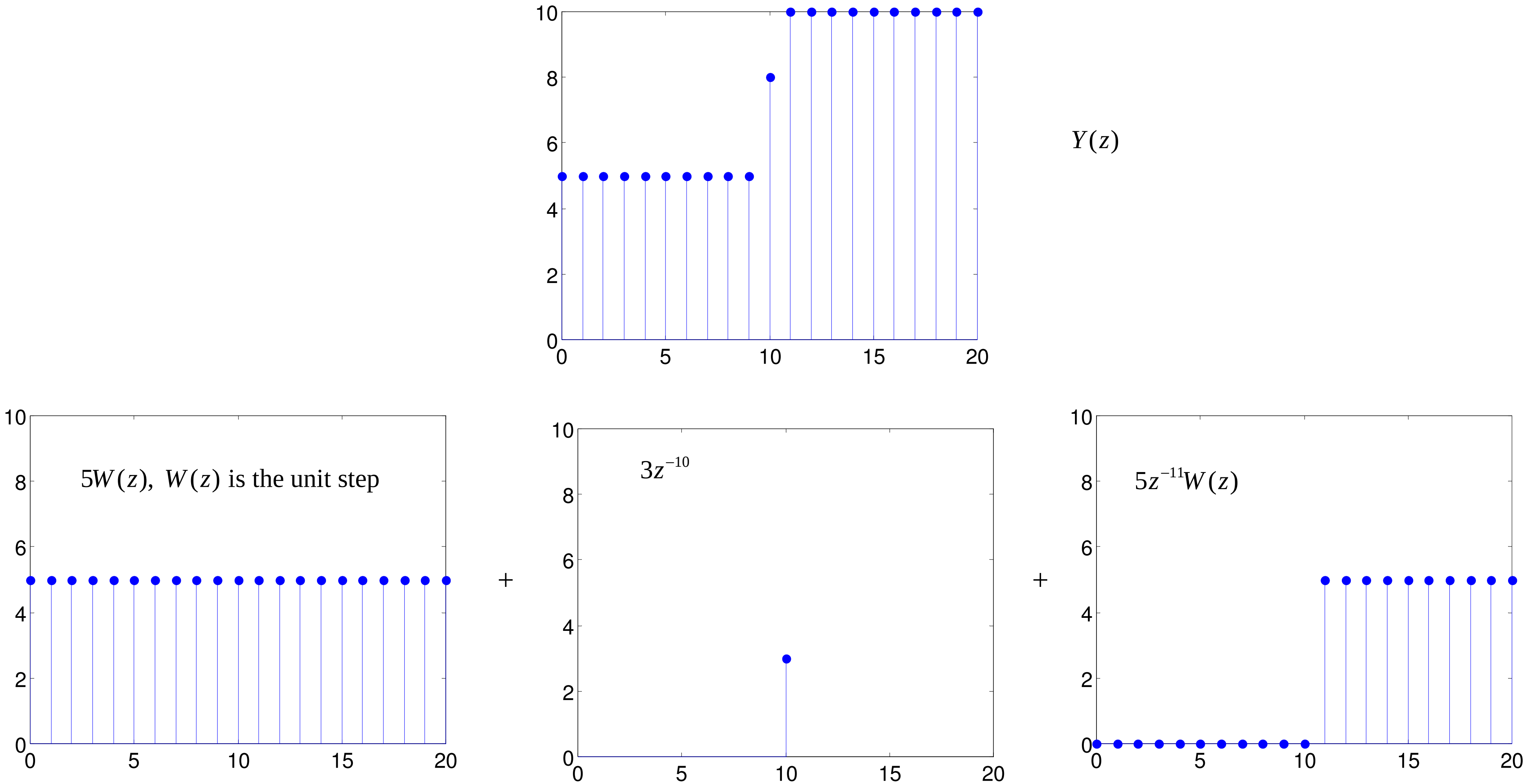
Infinite Length Signals



$$U(z) = 5 + 5z^{-1} + 5z^{-2} + \ldots$$

$$= \frac{5z}{z-1}$$

Construct a Z-Transform for the Following Infinite Length Signal



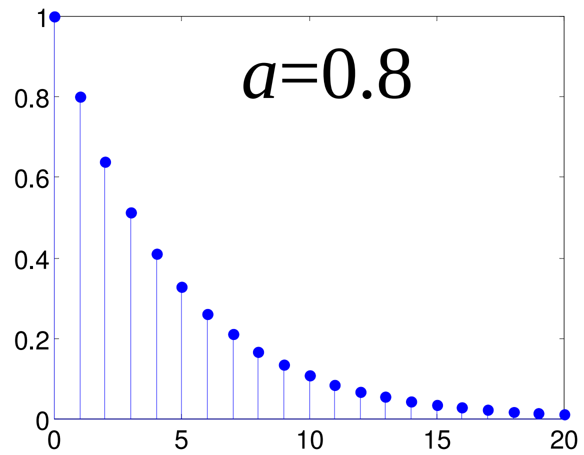
$$Y(z) = 5W(z) + 3z^{-10} + 5z^{-11}W(z)$$

Common Signals: Geometric

$$\text{Geometric}: y(k) = a^k$$

$$Y(z) = 1 + az + a^2 z^2 + \ldots$$

$$= \frac{z}{z-a}$$



$$Y(z) = 1 + 0.8z^{-1} + 0.64z^{-2} + \ldots$$

$$= \frac{z}{z-0.8}$$

Overview

- Open loop, closed loop
- Control system architecture
- Performance specifications (for expressing policies)
- Digital controlers
- z-Transform:
  - concept, signal composition
  - poles and stability
  - transfer function and system composition
  - proportional (P) and integral (I) control, PI control
- Effector example for a software system

Poles of a Z-Transform

Definition: Values of *z* for which the denominator is 0

Easy to find the poles of a geometric: $$V(z) = \frac{z}{z-a}$$ Pole is *a*.

Quick exercises: What are the poles of the following Z-Transforms?

$$Y(z) = \frac{5z}{z-1} + \frac{2z}{z-0.8}$$

Easy if sum of geometrics

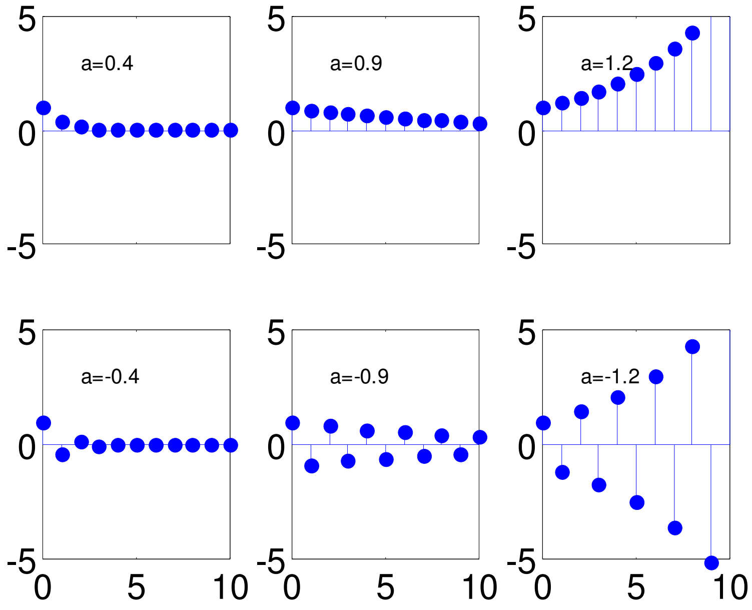
$$V(z) = \frac{3z}{z^2 - 0.5z + 0.06}$$

Harder if expanded polynomial

Poles determine key behaviors of signals

Effect of Pole on the Signal

$$y(k) = a^k \Leftrightarrow \frac{z}{z-a}$$



- What happens when
  - $|a|$ is larger?
  - $|a|>1$?
  - $a<0$?

- Larger $|a|$
  - Slower convergence
- $|a|>1$
  - Does not converge
- $a<0$
  - Oscillates

Why?

$$\frac{z}{z-a} = 1 + az + a^2 z^2 + \ldots \Leftrightarrow (1, a, a^2, \ldots)$$

Final Value Theorem

- Provides an easy way to determine the steady state value of a signal
  - Limit as *k* becomes large

$$v(\infty) = \lim_{z \to 1} (z-1)V(z)$$

(if V(z) has all of its poles inside the unit circle)

$$\lim_{z \longrightarrow 1} (z-1)\frac{z}{z-1} = 1$$

Final value of the unit step is 1.

$$\lim_{z \longrightarrow 1} (z-1)1 = 0$$

Final value of the impulse is 0.

$$\lim_{z \longrightarrow 1} (z-1)\frac{z}{z-a} = 0, \text{if } a \text{ is inside the unit circle}$$

Overview

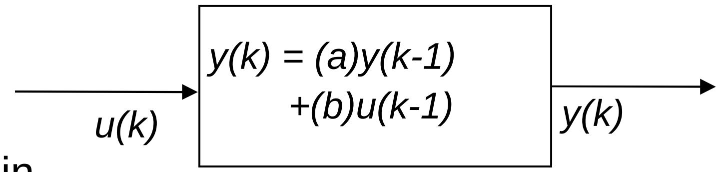
- Open loop, closed loop
- Control system architecture
- Performance specifications (for expressing policies)
- Digital controlers
- z-Transform:
  - concept, signal composition
  - poles and stability
  - transfer function and system composition
  - proportional (P) and integral (I) control, PI control
- Effector example for a software system

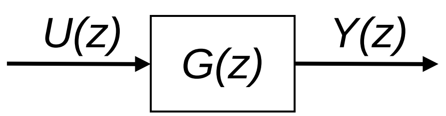
“Transfer Function” - Motivation and Definition

Motivation:

ARX model relates *u(k)* to *y(k)*

Transfer function expresses this relationship in the z domain





$$G(z) = \frac{\text{Output}}{\text{Input}} = \frac{Y(z)}{U(z)}$$

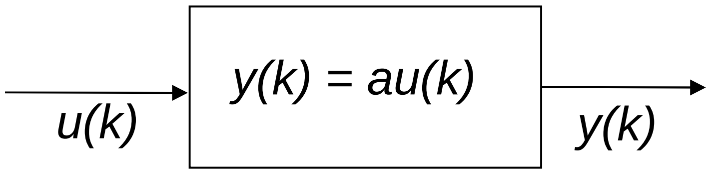
or $Y(z) = G(z)U(z)$

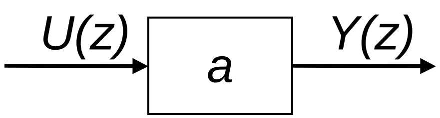
assuming initial conditions are 0.

A transfer function is specified in terms of its input and output.

Intuition for *G(z)*: Effect of “kicking the system”

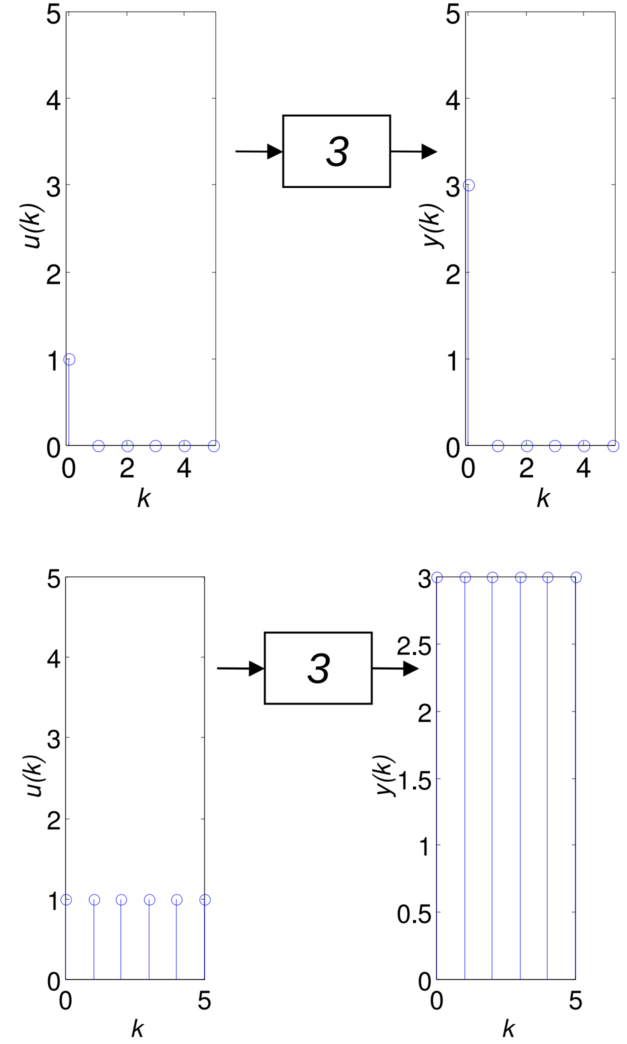
Constant Transfer Function



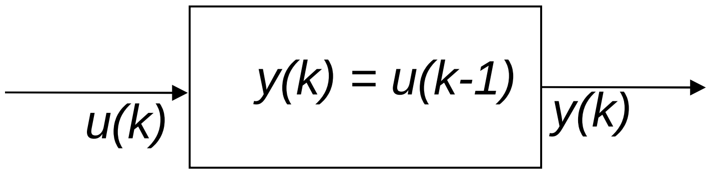


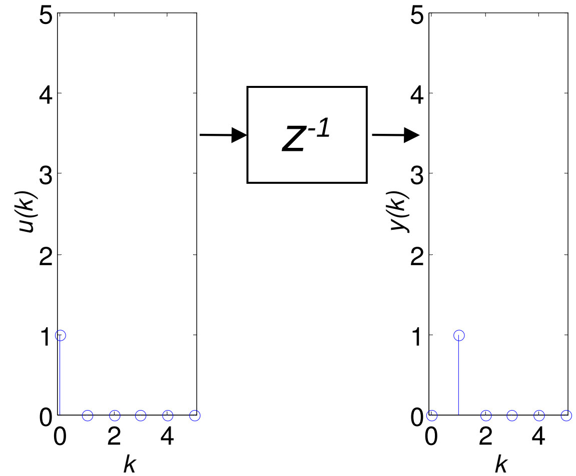
$$Y(z) = aU(z)$$

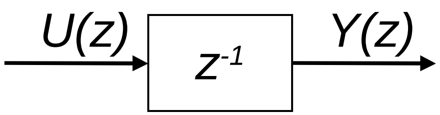
$$G(z) = \frac{Y(z)}{U(z)} = a$$



Time-Delay Transfer Function

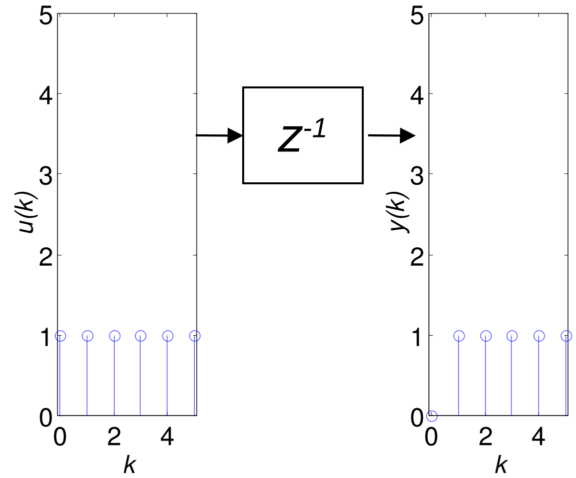




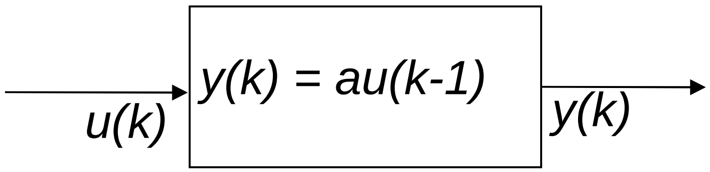


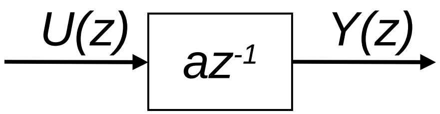
$$Y(z) = z^{-1}U(z)$$

$$G(z) = \frac{Y(z)}{U(z)} = z^{-1}$$



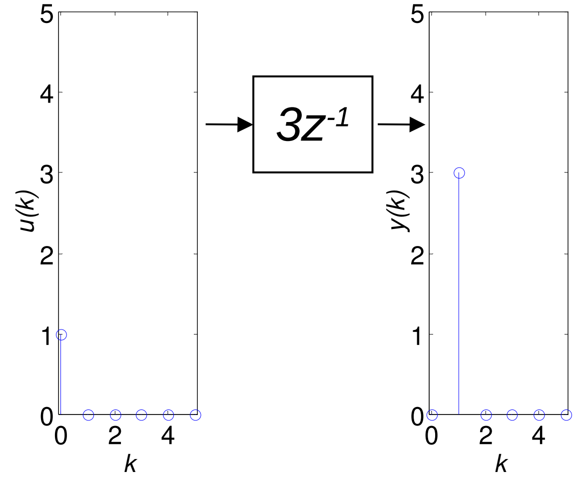
Combining Simple Transfer Functions

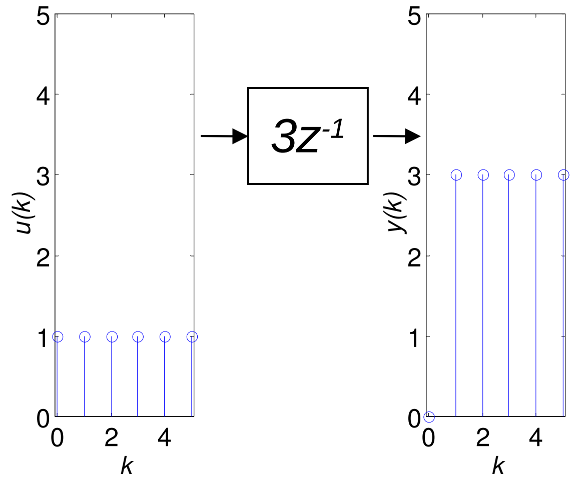




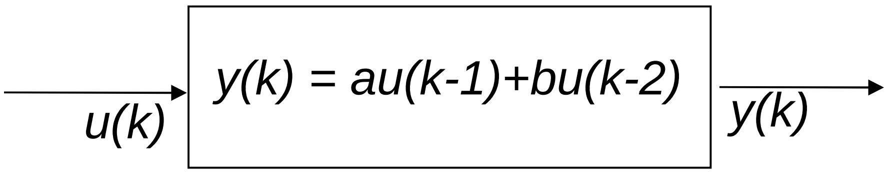
$$Y(z) = az^{-1}U(z)$$

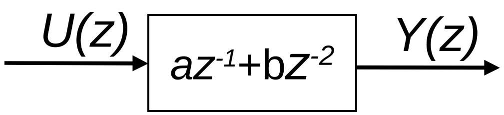
$$G(z) = \frac{Y(z)}{U(z)} = az^{-1}$$





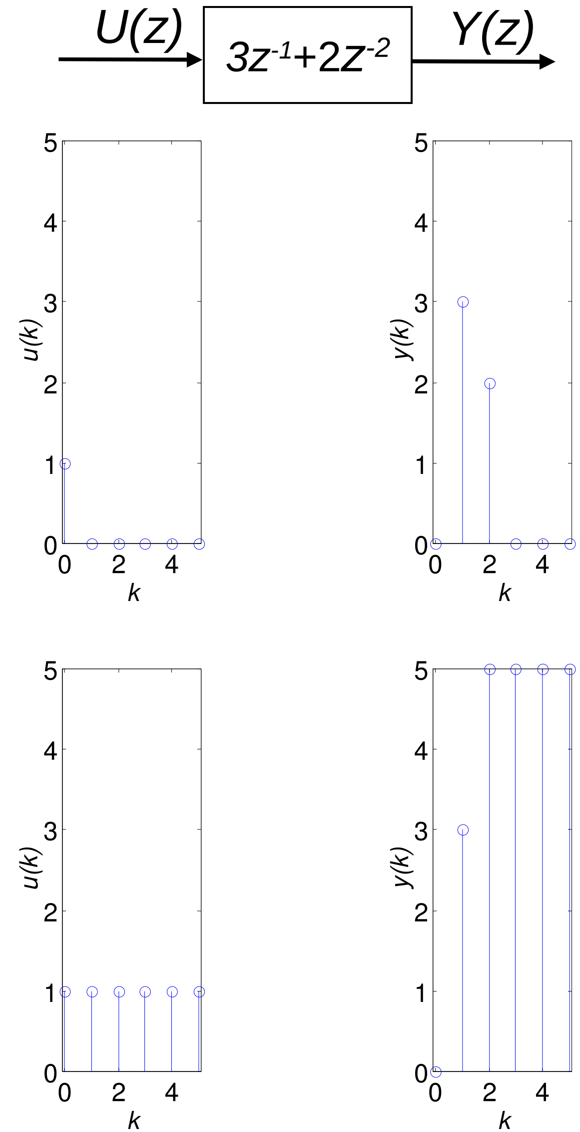
Additional Terms in T.F.



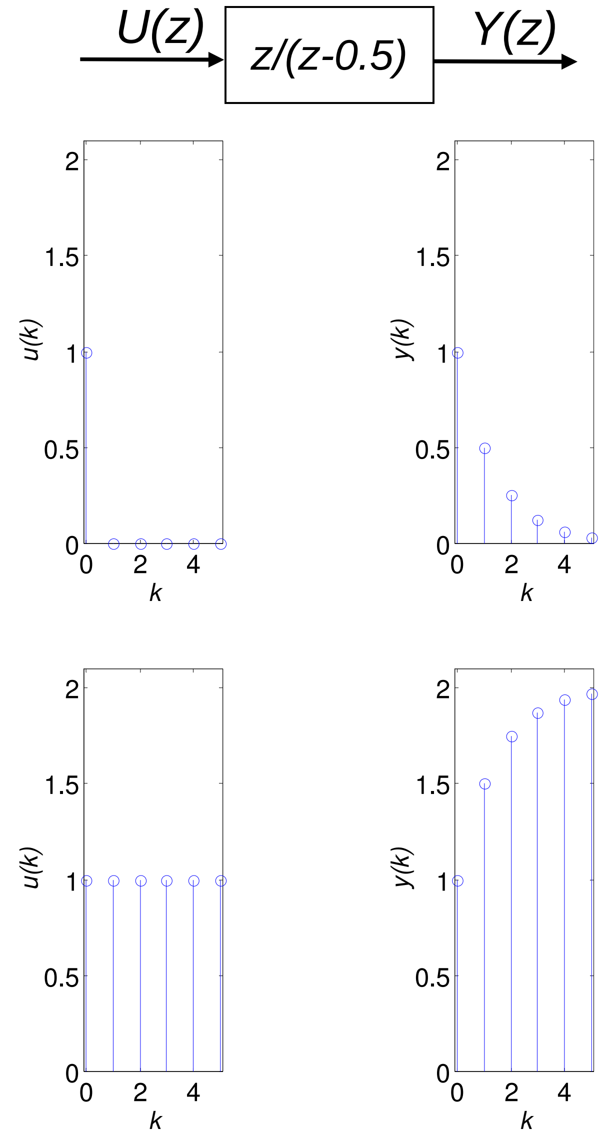


$$Y(z) = (az^{-1} + bz^{-2})U(z)$$

$$G(z) = \frac{Y(z)}{U(z)} = az^{-1} + bz^{-2}$$



Geometric Sum of T.F.



u(k) → $y(k) = u(k)+au(k-1)+a^2u(k-2)+\ldots$ → y(k)

U(z) → $1 + az^{-1} + a^2z^{-2} + \ldots$ → Y(z)

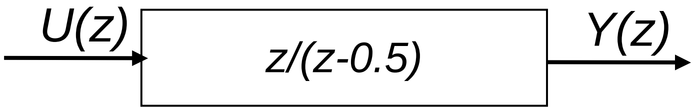
$$Y(z) = (1 + az^{-1} + a^2z^{-2} + \ldots)U(z)$$

$$G(z) = \frac{Y(z)}{U(z)} = 1 + az^{-1} + a^2z^{-2} + \ldots = \frac{z}{z-a}$$

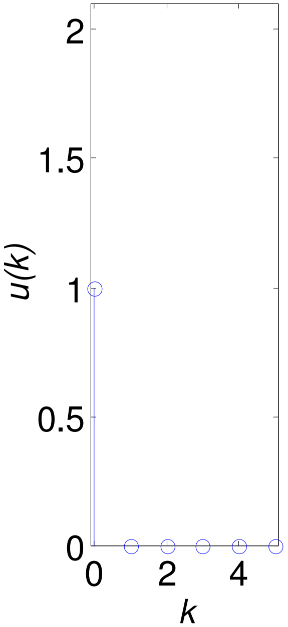
U(z) → z/(z-a) → Y(z)

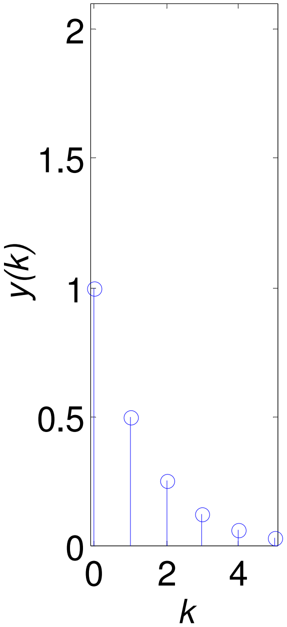
T.F. Interpretation

Signal generated by an impulse input



Example:





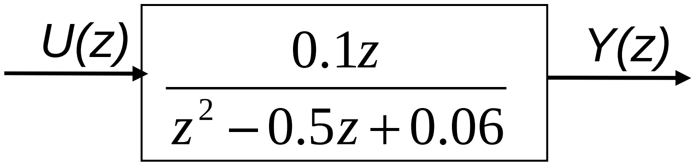
$$G(z) = \frac{Y(z)}{U(z)}$$

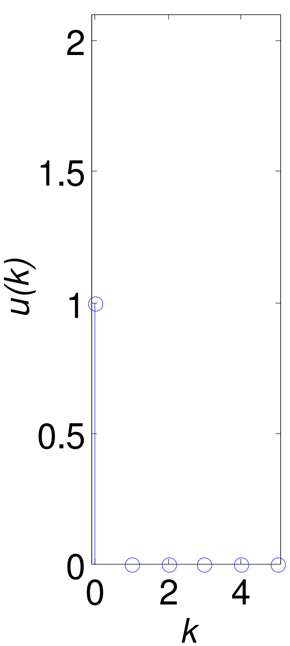
$$Y(z) = G(z)U(z) = G(z)(1)$$

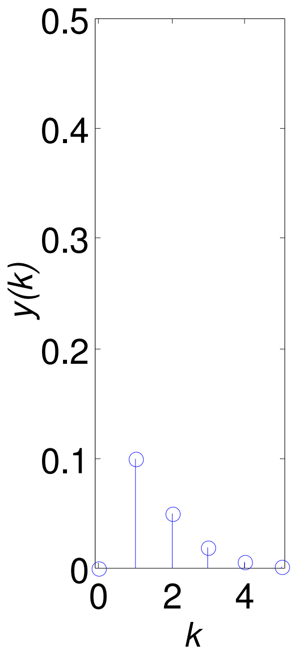
$$G(z) = \frac{z}{z-0.5} = 1 + 0.5z^{-1} + 0.25z^{-2} + 0.125z^{-3} + 0.0625z^{-4} + \ldots$$

Complicated Transfer Functions

Decompose into a sum of geometrics







$$
\begin{aligned}
G(z) &= \frac{0.1z}{z^2 - 0.5z + 0.06} \\
&= \frac{z}{z-0.3} - \frac{z}{z-0.2} \\
&= 1 + 0.3z^{-1} + 0.09z^{-2} + \ldots - 1 - 0.2z^{-1} - 0.04z^{-2} + \ldots \\
&= 0.1z^{-1} + 0.05z^{-2} + \ldots
\end{aligned}
$$

- **Partial fraction expansion** allows rational polynomials to be decomposed into a sum of geometrics
- Poles of the original polynomial are the poles of the geometrics

Converting ARX Models into T.F.

Approach

- Replace $u(k+n)$ by $z^n U(z)$ *and* replace $y(k+n)$ by $z^n Y(z)$
- Express $Y(z)$ as a function of $U(z)$
- Divide both sides by $U(z)$

Example:

$$\text{T.S. Model}: y(k+1) = 0.43y(k) + 0.47u(k)$$

$$1:\ zY(z) = 0.43Y(z) + 0.47U(z)$$

$$2: (z-0.43)Y(z) = 0.47U(z);\ Y(z) = \frac{0.47}{z-0.43}U(z)$$

$$3: \frac{Y(z)}{U(z)} = \frac{0.47}{z-0.43}$$

Converting T.F. into ARX Model

Approach

- Express transfer function as *Y(z)=G(z)U(z)*
- Multiply both sides by the denominator of *G(z)*
- Convert *Y(z)* and *U(z)* into the time domain

Example 1:

$$G(z) = \frac{z}{z-0.3}$$

$$Y(z) = \frac{z}{z-0.3}U(z)$$

$$Y(z)(z-0.3) = zU(z)$$

$$zY(z) \Leftrightarrow y(k+1)$$

$$0.3Y(z) \Leftrightarrow 0.3y(k)$$

$$zU(z) \Leftrightarrow u(k+1)$$

$$y(k+1) - 0.3y(k) = u(k+1)$$

$$y(k) = 0.3y(k-1) + u(k)$$

Example 2:

$$G(z) = \frac{0.1z}{z^2 - 0.5z + 0.06}$$

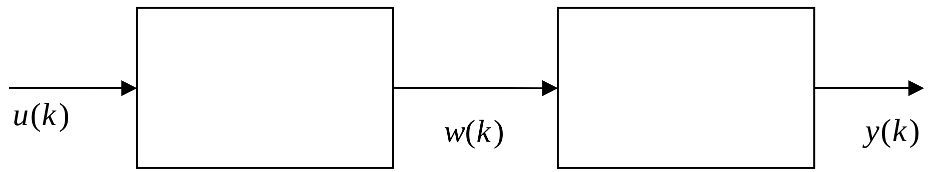
$$Y(z) = \frac{0.1z}{z^2 - 0.5z + 0.06}U(z)$$

$$Y(z)(z^2 - 0.5z + 0.06) = 0.1zU(z)$$

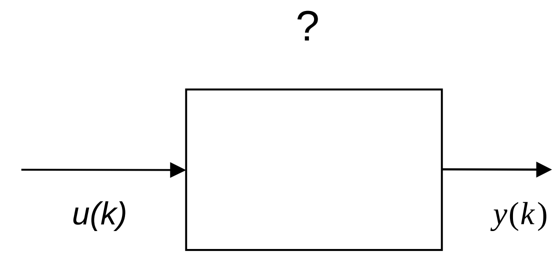
$$y(k) = 0.5y(k-1) + 0.06y(k-2) + 0.1u(k-1)$$

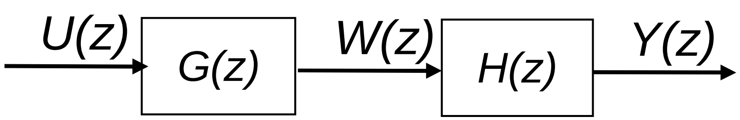
Transfer Functions In Series

$w(k+1) = (0.43)w(k) + (0.47)u(k)$ $\quad$ $y(k+1) = 0.8y(k) + 0.72w(k) - 0.66w(k-1)$

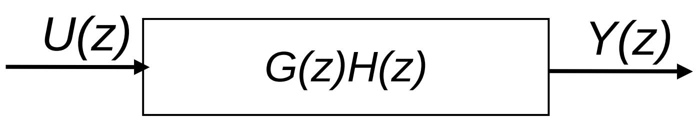


$\Leftrightarrow$





$\Leftrightarrow$



$$\frac{Y(z)}{U(z)} = \frac{W(z)}{U(z)}\frac{Y(z)}{W(z)} = G(z)H(z)$$

T.F. provide an easy way to analyze the behavior of complex structures.

Poles of a Transfer Function

<u>Poles: Values of *z* for which the denominator is 0.</u>

Example:

$$H(z) = \frac{0.1z}{z^2 - 0.5z + 0.06} = \frac{z}{z-0.3} - \frac{z}{z-0.2}$$

Poles: 0.3, 0.2

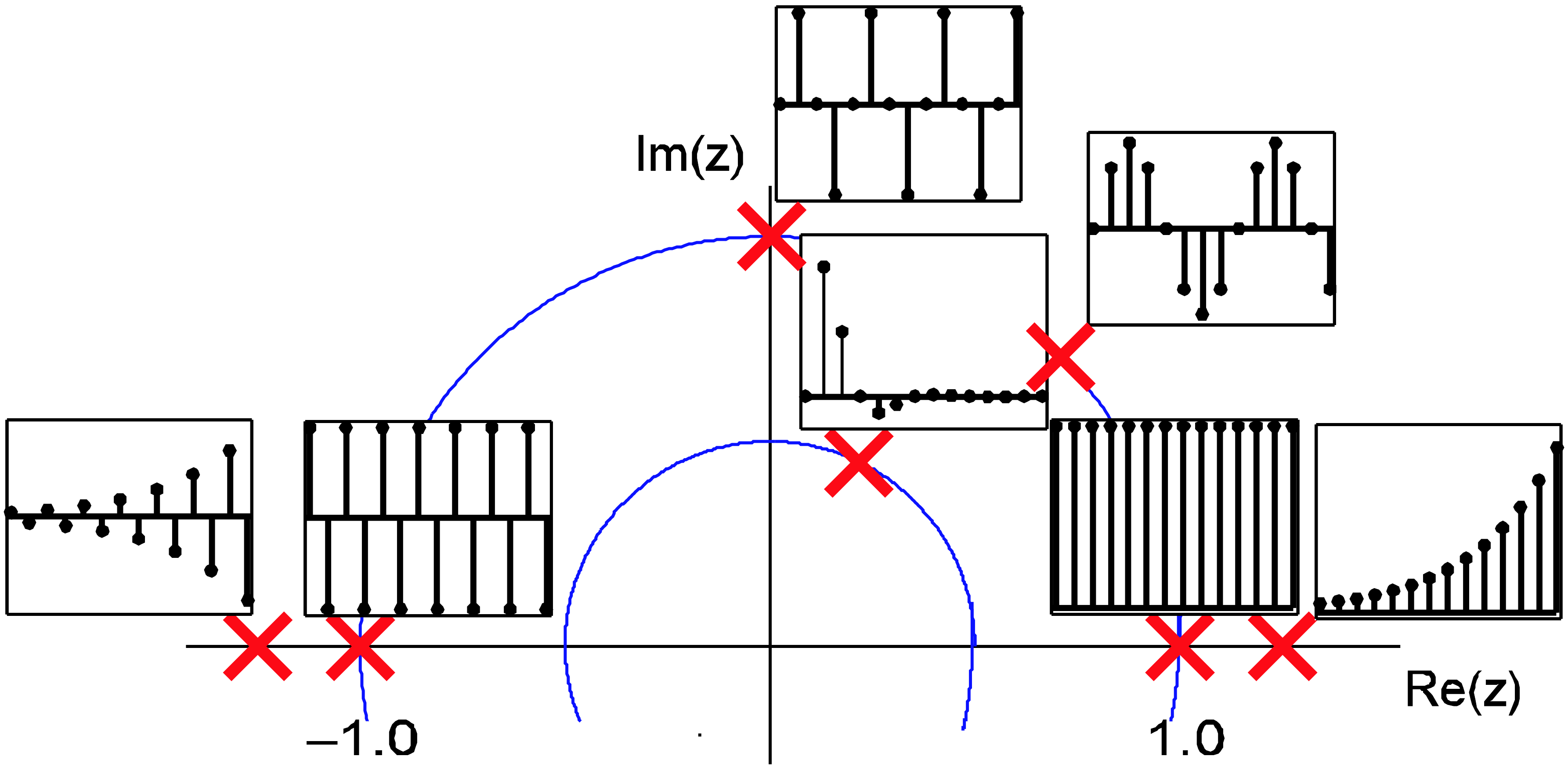
Poles

- Determine stability
- Major effect on settling time, overshoot

$$G(z) = \frac{z}{z-a} = 1 + az^{-1} + a^2 z^{-2} + a^3 z^{-3} + \ldots$$

- Larger $|a|$
  - Slower convergence
- $|a|>1$
  - Does not converge
- $a<0$
  - Oscillates

Almost All You'll Ever Need to Know About Poles



$$G(z) = \frac{z}{z-a} = 1 + az^{-1} + a^2 z^{-2} + a^3 z^{-3} + \ldots$$

Steady State Gain (ssg) of a Transfer Function

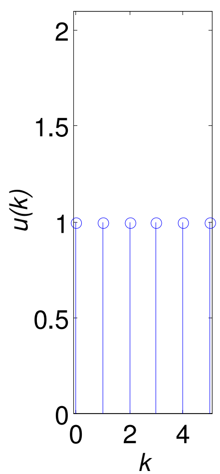
Definition and result: Steady state divided by steady state input.

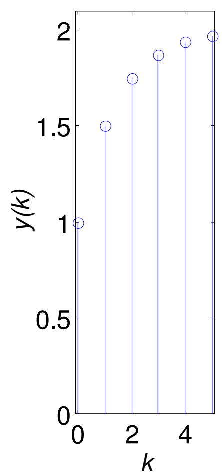
from final value thm

ssg of *G(z)* is $\dfrac{y(\infty)}{u(\infty)} = G(1)$

<u>Example:</u>

$$u(\infty) = 1 \qquad G(z) = \frac{z}{z - 0.5} \qquad y(\infty) = 2$$



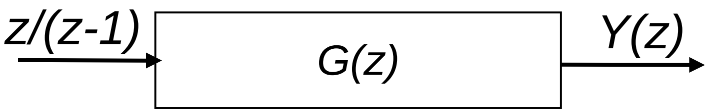


$$\frac{y(\infty)}{u(\infty)} = \frac{2}{1} = 2 = G(1)$$

Settling Time (k_s) of a System

Definition and result: Time until an input signal is within 2% of its steady state

from geometric

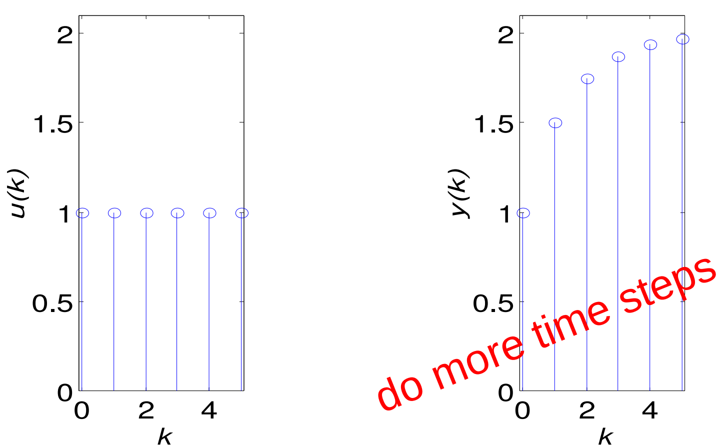


$$k_S \approx \frac{-4}{\log|a|}, \text{ where } |a| \text{ is the largest pole of } G(z)$$

Examples:

$$G(z) = \frac{z}{z - 0.5}$$

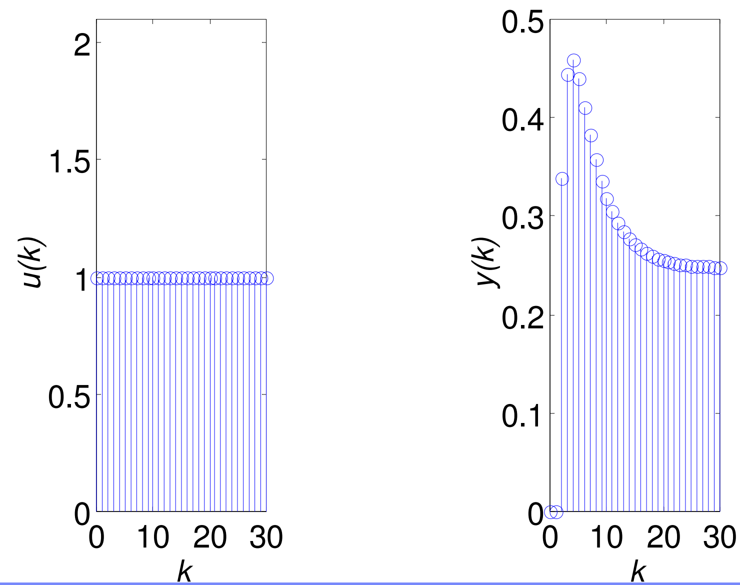
$$k_S \approx \frac{-4}{\log 0.5} \approx 6$$



$$G(z) = \frac{0.34z - 0.31}{z^3 - 1.23z^2 + 0.34z},$$

poles : 0, 0.43, 0.8

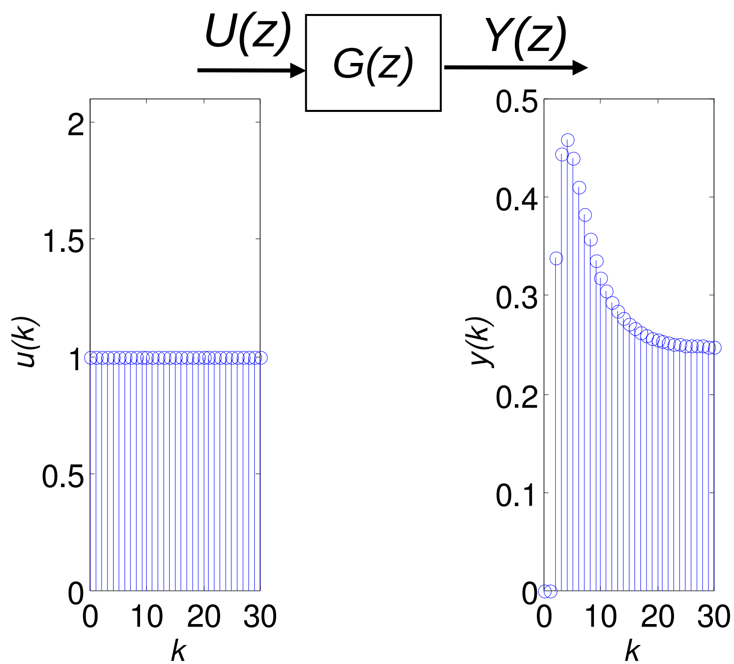
$$k_S \approx \frac{-4}{\log 0.8} \approx 18$$



Summary of Key Results

- Interpretation of transfer functions
  - Signal created by an impulse (impulse response)
  - Sum of time delayed effects
  - Encoding of a ARX model
- Transfer functions in series
  - End-to-end T.F. is the product of the individual T.F.
- Steady state gain
  - Magnitude of effect on the output for a unit change in input
- Settling time
  - Time until steady state is reached after applying a unit input

Key Results for LTI Systems (LTI = Linear Time Invariant)



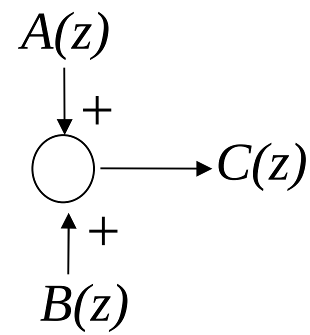
Stable if $|a|<1$, where a is the largest pole of $G(z)$

$$k_s \approx \frac{-4}{\log|a|}, \text{ where } |a| \text{ is the largest pole of } G(z)$$

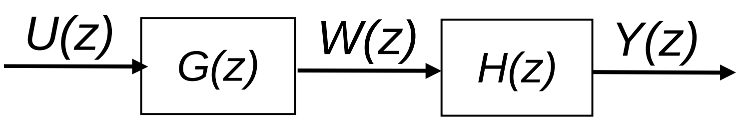
ssg of $G(z)$ is $\dfrac{y(\infty)}{u(\infty)} = G(1)$

Adding signals:

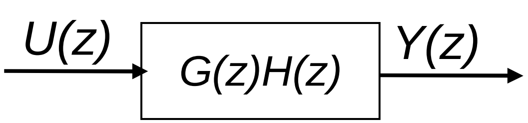
If $\{a(k)\}$ and $\{b(k)\}$ are signals, then $\{c(k)=a(k)+b(k)\}$ has Z-Transform $A(z)+B(z)$.



Transfer functions in series



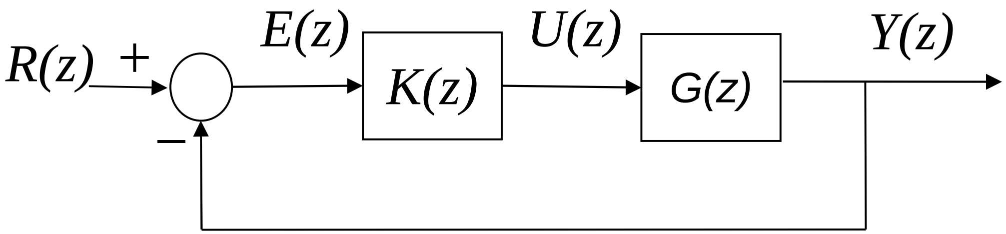
is equivalent to



Overview

- Open loop, closed loop
- Control system architecture
- Performance specifications (for expressing policies)
- Digital controlers
- z-Transform:
  - concept, signal composition
  - poles and stability
  - transfer function and system composition
  - proportional (P) and integral (I) control, PI control
- Effector example for a software system

Basic Controllers



Proportional (P) Control

$$u(k) = K_P e(k)$$

$$K(z) = \frac{U(z)}{E(z)} = K_P$$

Integral (I) Control

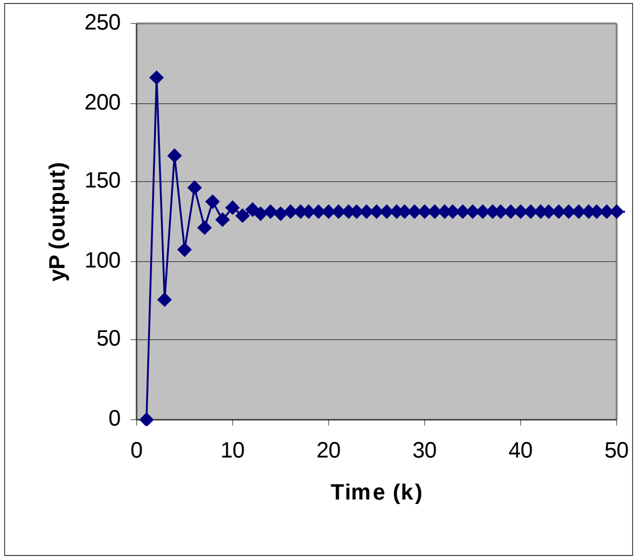
$$u(k+1) = K_I u(k) + e(k+1)$$

$$K(z) = \frac{U(z)}{E(z)} = \frac{K_I z}{z-1}$$

K_P and K_I are called ***control gains***.

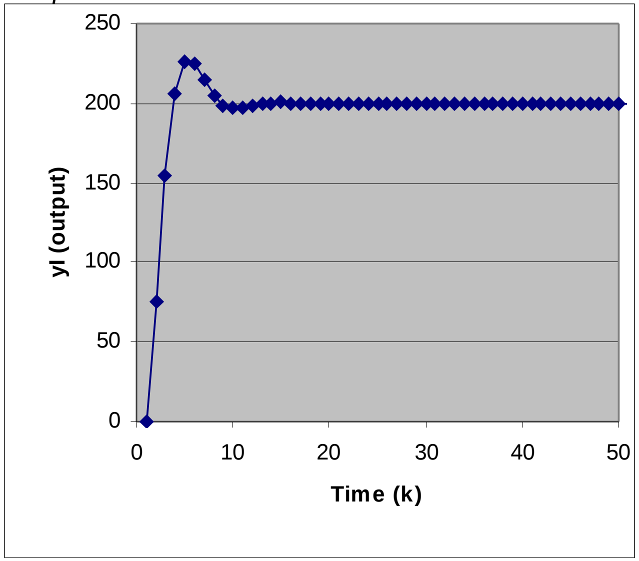
Conclusions from P vs. I Comparison

K_P=2.3



$r(k)$=200

K_I=0.8



Conclusions:

P is fast

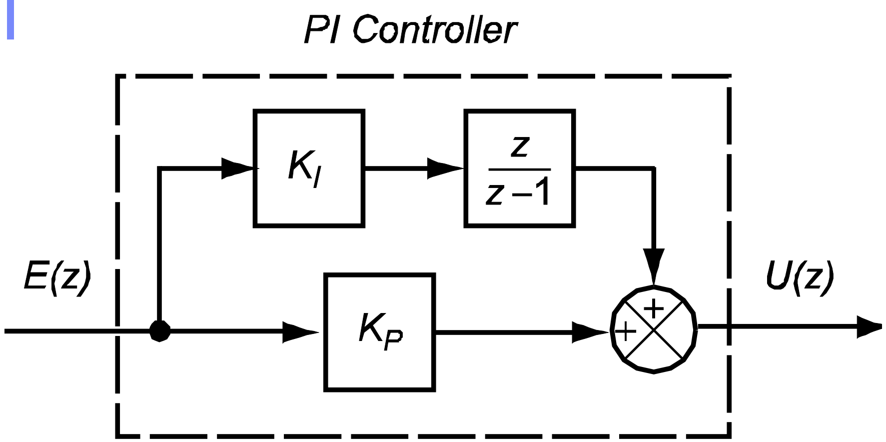
I is accurate and has less overshoot.

Design challenge:

Make P accurate.

Reduce P's overshoot.

PI Control



$$u(k) = u_P(k) + u_I(k)$$
$$= u(k-1) + (K_P + K_I)e(k) - K_P e(k-1)$$

$$K(z) = \frac{E(z)}{U(z)} = K_P + \frac{K_I z}{z-1} = \frac{(K_P + K_I)z - K_P}{z-1}$$

Overview

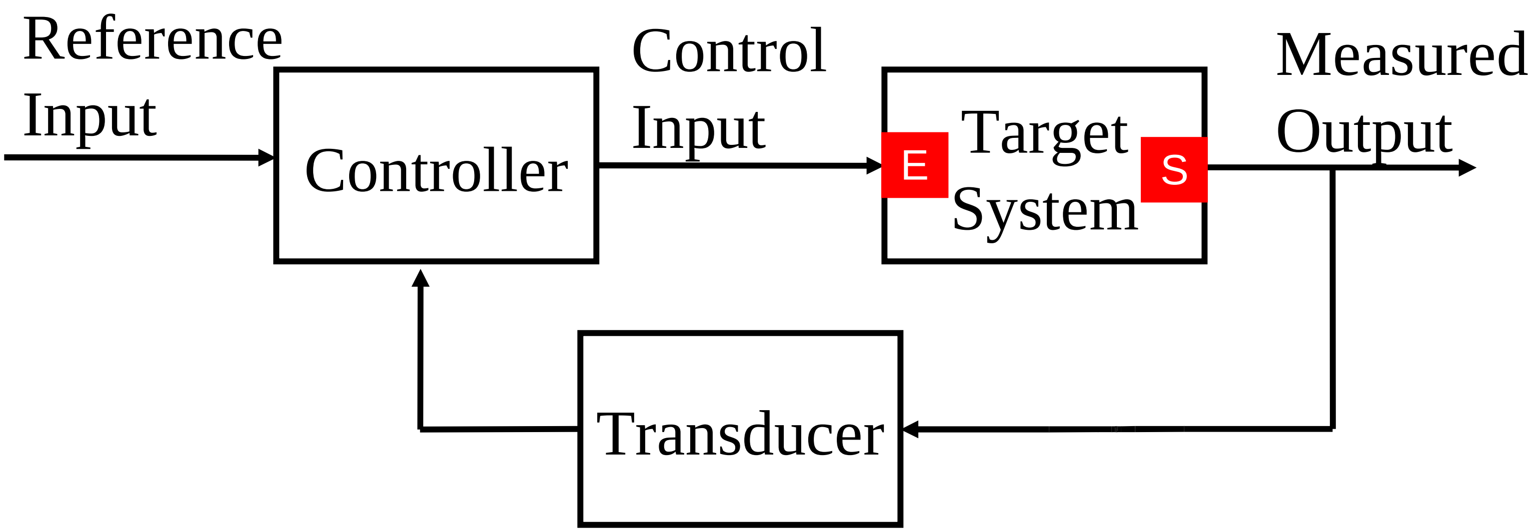
- Open loop, closed loop
- Control system architecture
- Performance specifications (for expressing policies)
- Digital controlers
- z-Transform:
  - concept, signal composition
  - poles and stability
  - transfer function and system composition
  - proportional (P) and integral (I) control, PI control
- Effector example for a software system

M7 Topics

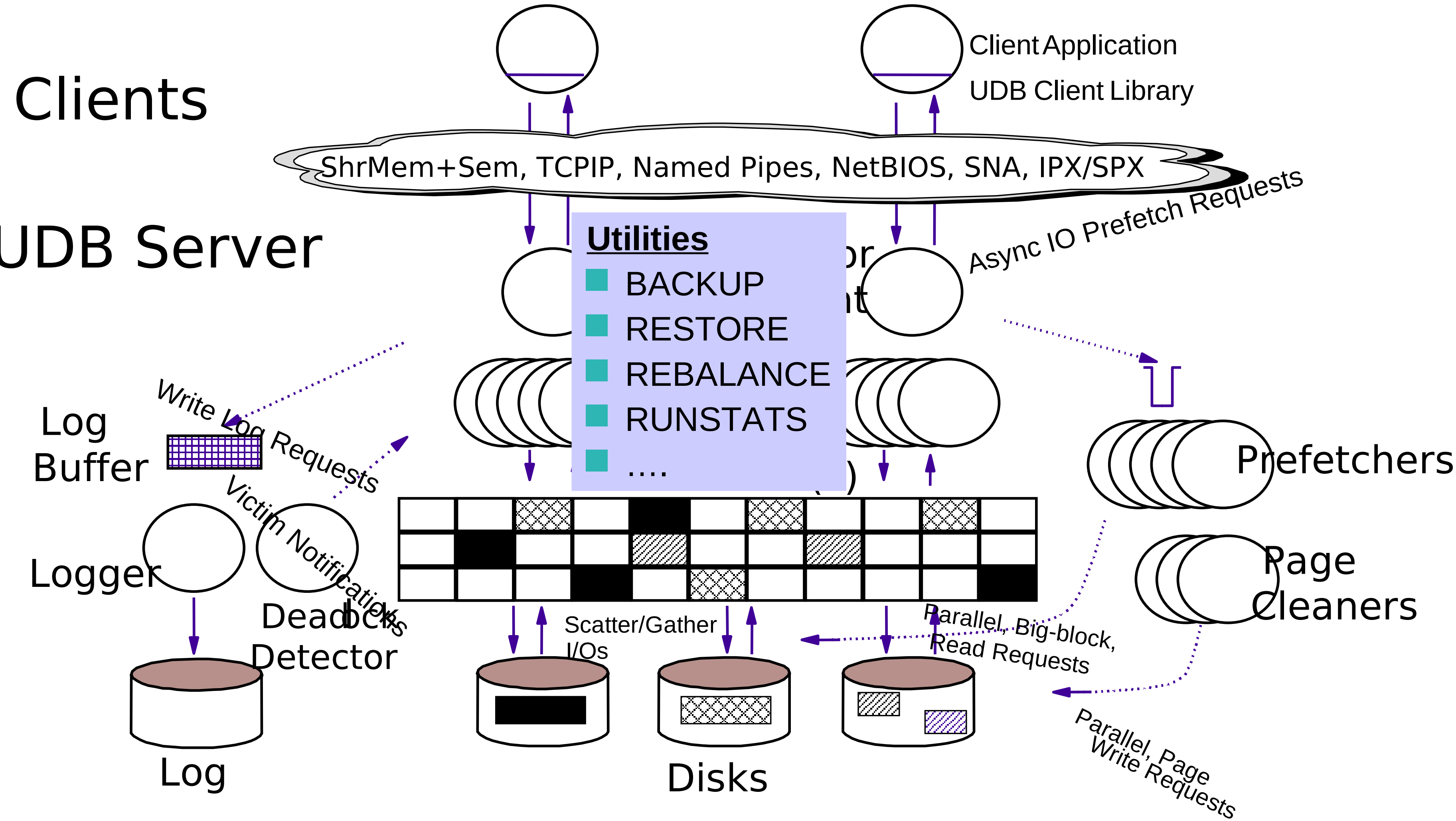
- DB2 Utilities Throttling (DB2 v8.1)
  - http://www.databasejournal.com/features/db2/article.php/3339041
  - "Throttling Utilities in the IBM DB2 Universal Database Server," Sujay Parekh, Kevin Rose, Yixin Diao, Victor Chang, Joseph L. Hellerstein, Sam Lightstone, Matthew Huras. American Control Conference, 2004.
- Self-tuning memory management
  - "Using MIMO Linear Control for Load Balancing in Computing Systems," Yixin Diao, Joseph L. Hellerstein, Adam Storm, Maheswaran Surendra, Sam Lightstone, Sujay Parekh, and Christian Garcia-Arellano. American Control Conference, 2004.
  - "Incorporating Cost of Control Into the Design of a Load Balancing Controller," Yixin Diao, Joseph L. Hellerstein, Adam Storm, Maheswaran Surendra, Sam Lightstone, Sujay Parekh, and Christian Garcia-Arellano. Real-Time and Embedded Technology and Application Systems Symposium, 2004.

DB2 Utilities Throttling

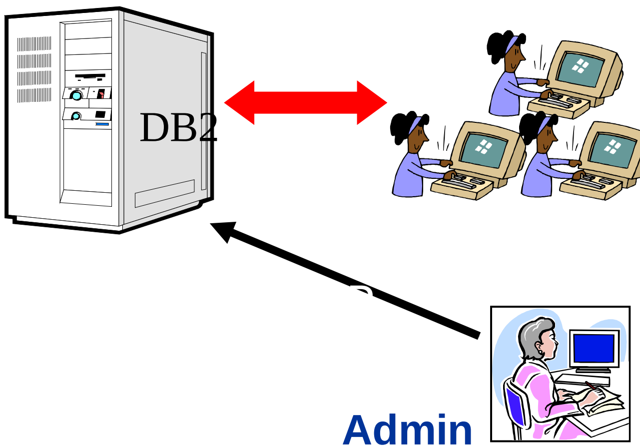
- System description
- Control problem
- Design of sensors (S) and effectors (E)
- Control system design
- Evaluation



System Description: DB/2 UDB

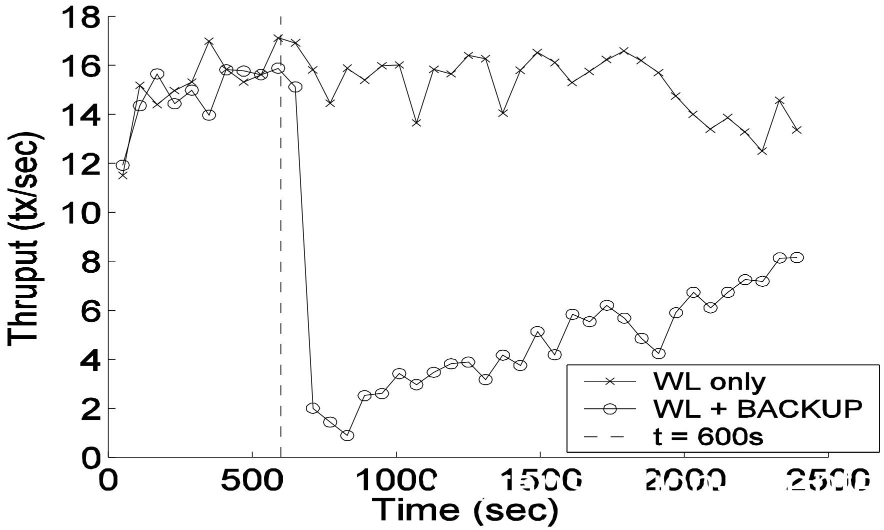


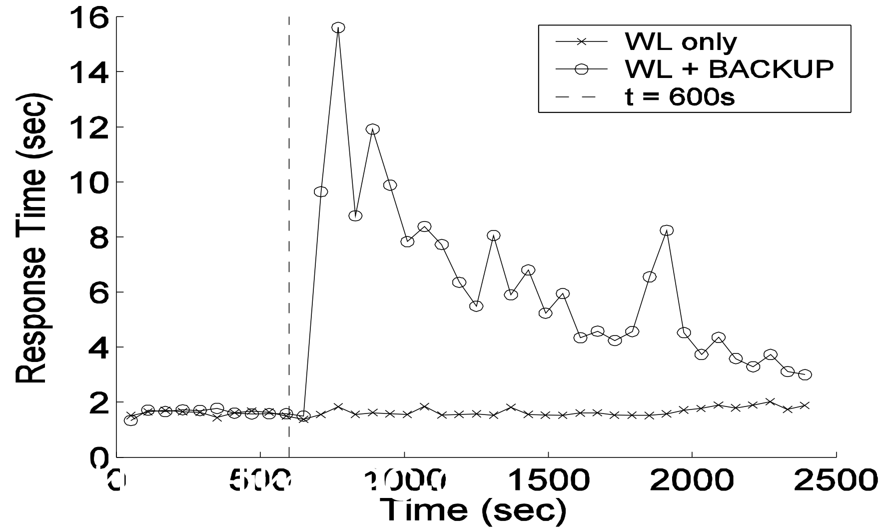
Control Problem: Utility Impact



Administrative policy

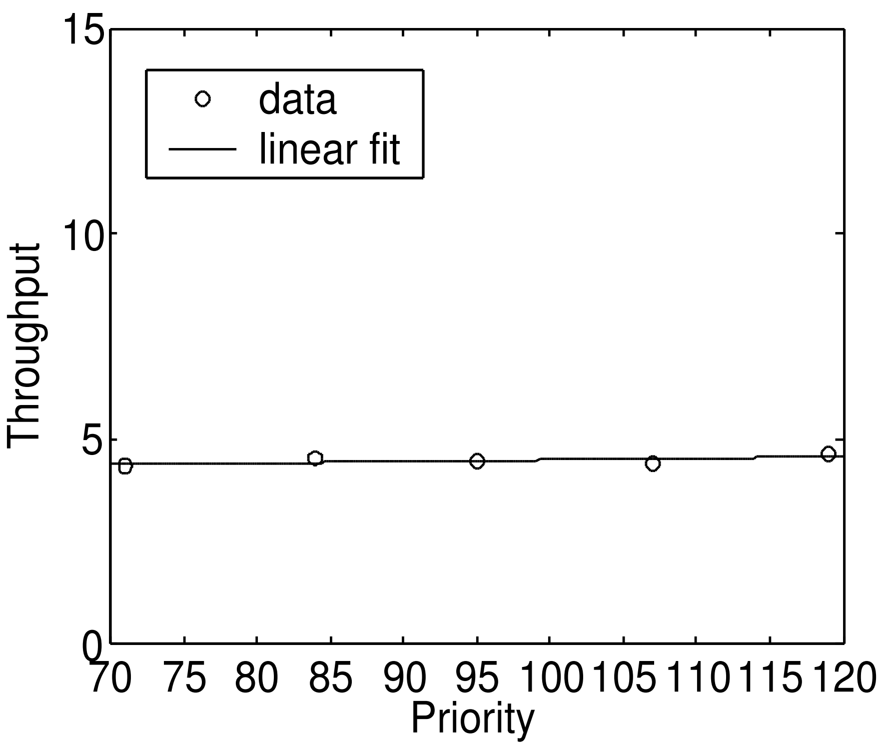
There should be no more than an x% performance degradation of production work as a result of executing administrative utilities





Utilities have a huge impact on production performance.

Effector Design: Why Not Priorities



- Assign fixed priority to Utility processes
  - Unix => higher # is lower priority
- Measure avg Workload throughput over a run
- NO EFFECT!
- Why?
  - BACKUP (and others) are I/O bound
  - OS priority == CPU priority
    - No effect on other resources
- Even if I/O priorities are present, may not solve the problem
  - Don't want to starve the utilities

Effector Design: Self-Induced sleep (SIS)

Augmenting the Utility

```
FUNCTION Utility()
BEGIN
        WHILE (NOT done)
        BEGIN
                ... do some work ...
                SleepIfNeeded()
        END
END
```

"Library" implementation of Throttling

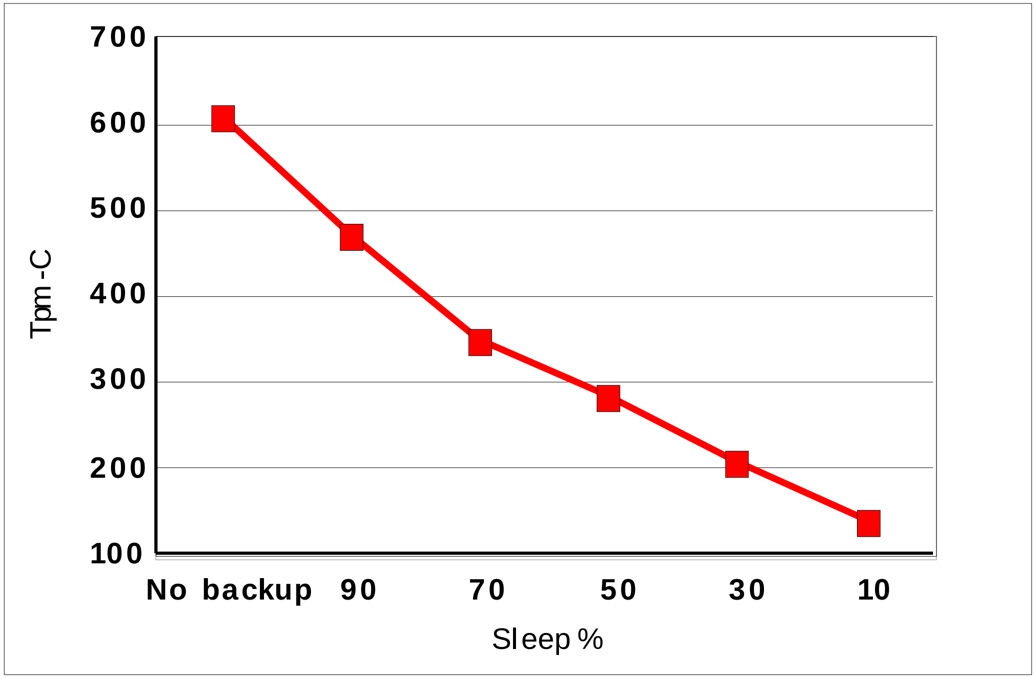
```
FUNCTION SleepIfNeeded()
BEGIN
        (workTime, sleepTime) = GetThrottlingLevel()
;
        timeWorked = Now() - workStart ;
        IF (timeWorked > workTime)
                SLEEP( sleepTime ) ;
                workStart = Now() ;
        ENDIF
END
```

Appeal: Handle any resource bottleneck

Shortcoming: Requires change to application code.

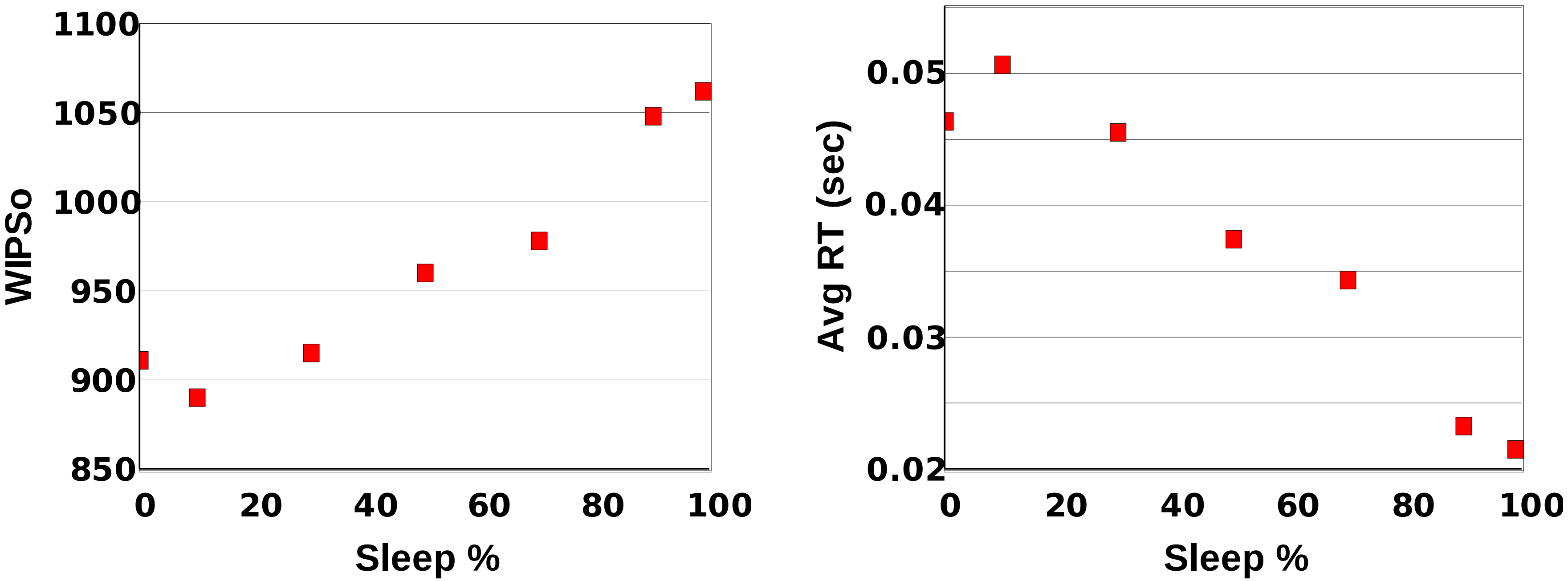
But change is modest and does not affect the customer

Effector Design: Assessment of SIS for TPM-C



Wide range of "control volume"
Linear effect

Effector Design: Assessment of SIS for TPC-W



Wide range of "control volume"
Linear effect

Summary

- Control theory
  - a model and a tool
  - starts with system identification, „black box“ view
  - finding good effectors
  - „designing a controller“ for given policy
- Predictable outcome
  - stability analysis, controler bandwidth etc
- Is it „autonomic“?
  - adaptive, on the reactive side of the adaptivity spectrum
  - important example of the „sens/analyze/plan/exec“ cycle