



University of Southern California
Center for Software Engineering

CS599 Software Process Modeling

Week 7

Ray Madachy

October 12, 1999



Outline

- Quiz #2
- Modeling reminders
- Signups for student project briefings
- Calibration example using process
concurrency homework problem
- General system behaviors
- Rayleigh model



Modeling Reminders

- For all homeworks and projects
 - always show equations
 - liberally include comments in equations
 - provide rationale for numeric values and calibrations
 - show outputs for different simulation cases
 - discuss results



Term Project Briefings

- Plan for about 20 minutes presentation
- Describe problem and background
 - what will result from your study
- Show system boundary
- How will a user interact?
 - inputs and outputs
 - a prototype of I/O would be very useful to show
- Reference behaviors and other methods of verification
- Other material as appropriate
- Signups

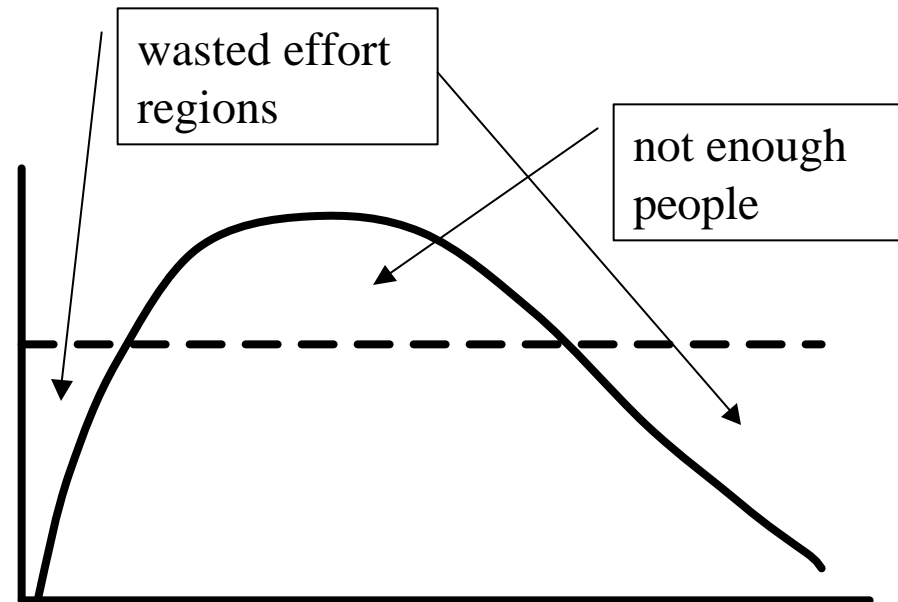


Calibration Example

- Use COCOMO II model to match the given effort for the software development portion
 - derive schedule, personnel count and productivity rates for the model from COCOMO estimate
 - derive the nominal specification productivity from problem statement
 - Example
 - nominal COCOMO estimate for 16 KSLOC is 6 person-years (72 person-months) and 11.3 calendar months
 - nominal software development productivity = $16 \text{ KSLOC} / 72 \text{ person-months} = .222 \text{ KSLOC/person-month}$
 - average personnel = $72 \text{ person-months} / 11.3 \text{ months} = 6.4 \text{ people}$
 - nominal specification productivity = $16 \text{ KSLOC} / 24 \text{ person-months} = .666 \text{ KSLOC/person-month}$
- For homework, experiment with different staffing patterns and process concurrence relationships

Staffing Waste

- Rayleigh-like ideal staffing pattern vs. constant level-of-effort





General System Behaviors

- Behaviors are representative of many known types of systems.
- Knowing how systems respond to given inputs is valuable intuition for the modeler
- Can be used during model assessment
 - use test inputs to stimulate the system behavioral modes



System Order

- The *order* of a system refers to the number of levels contained.
- A single level system cannot oscillate, but a system with at least two levels can oscillate because one part of the system can be in disequilibrium.

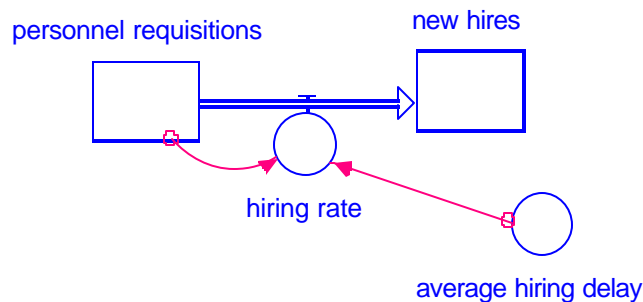


Example System Behaviors

- Delays
- Goal-seeking Negative Feedback
 - *First-order Negative Feedback*
 - *Second-order Negative Feedback*
- Positive Feedback Growth or Decline
- S-curves

Delays

- Time delays are ubiquitous in processes
- They are important structural components of feedback systems.
- Example: hiring delays in software development.
 - the average hiring delay represents the time that a personnel requisition remains open before a new hire comes on board

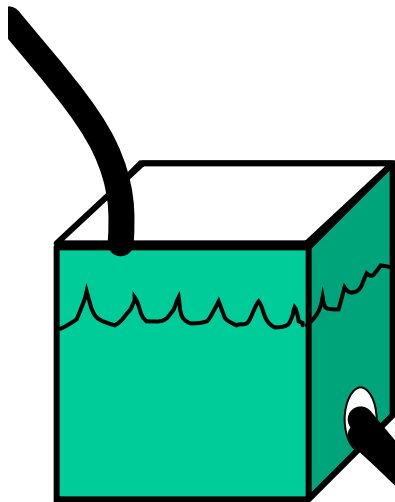


```

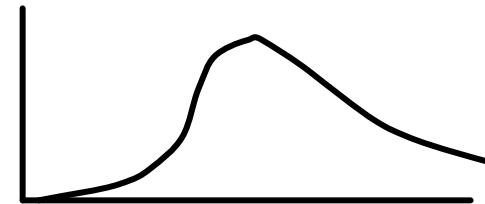
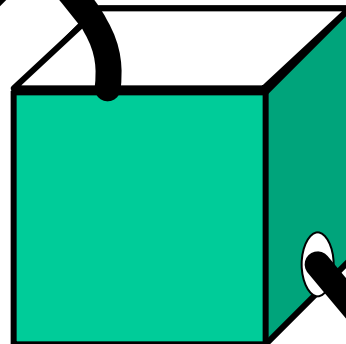
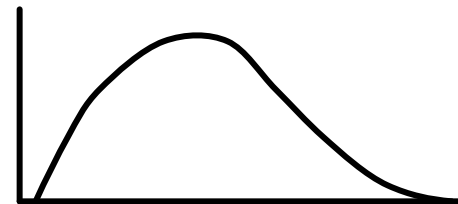
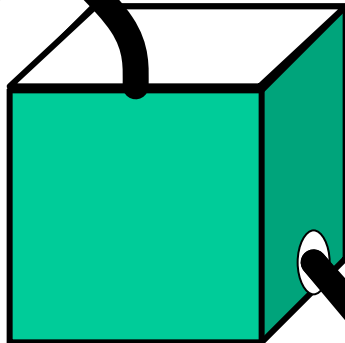
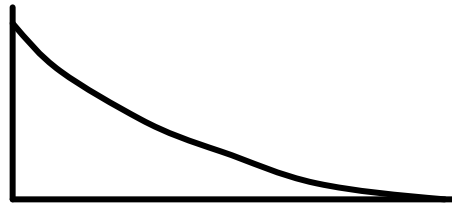
□ new_hires(t) = new_hires(t - dt) + (hiring_rate) * dt
INIT new_hires = 0
INFLOWS:
  ⚙ hiring_rate = personnel_requisitions/average_hiring_delay
□ personnel_requisitions(t) = personnel_requisitions(t - dt) + (- hiring_rate) * dt
INIT personnel_requisitions = 100
OUTFLOWS:
  ⚙ hiring_rate = personnel_requisitions/average_hiring_delay
○ average_hiring_delay = 10
  
```

Third Order Delay

- A series of 1st order delays
- Graphs show water levels over time in each tank

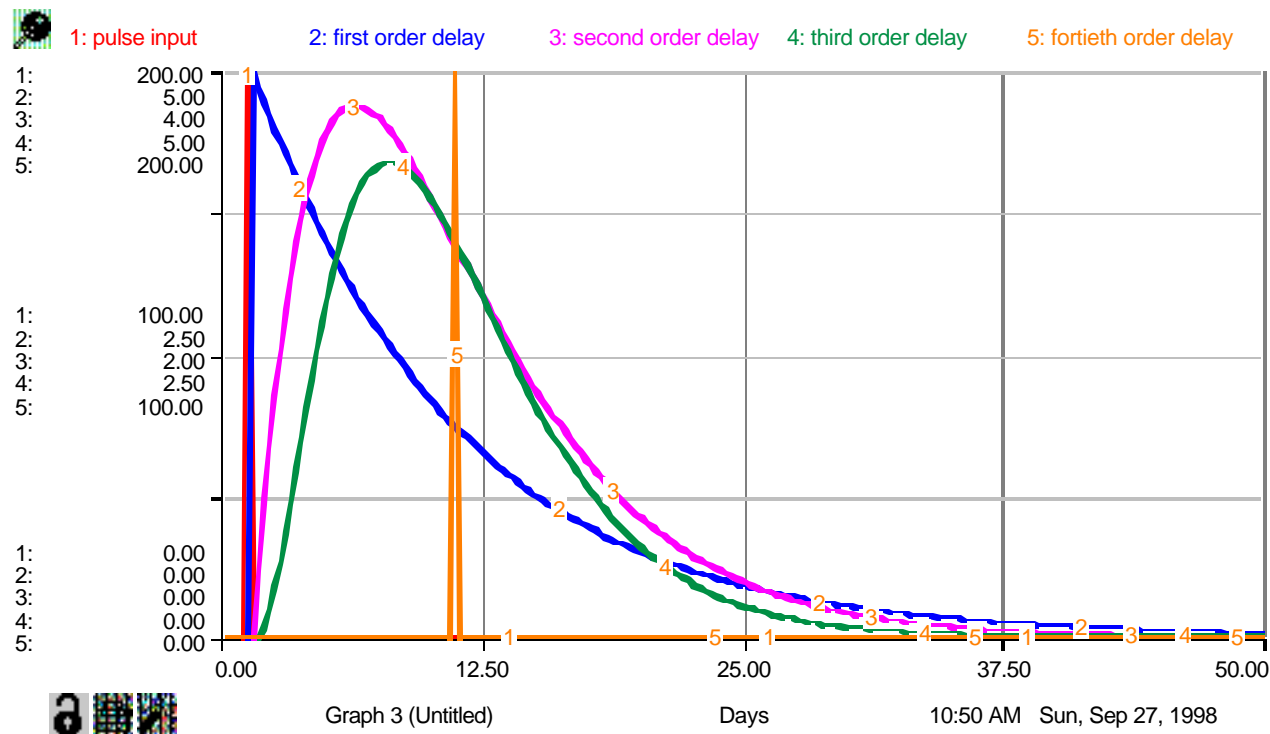


tank 1 starts full



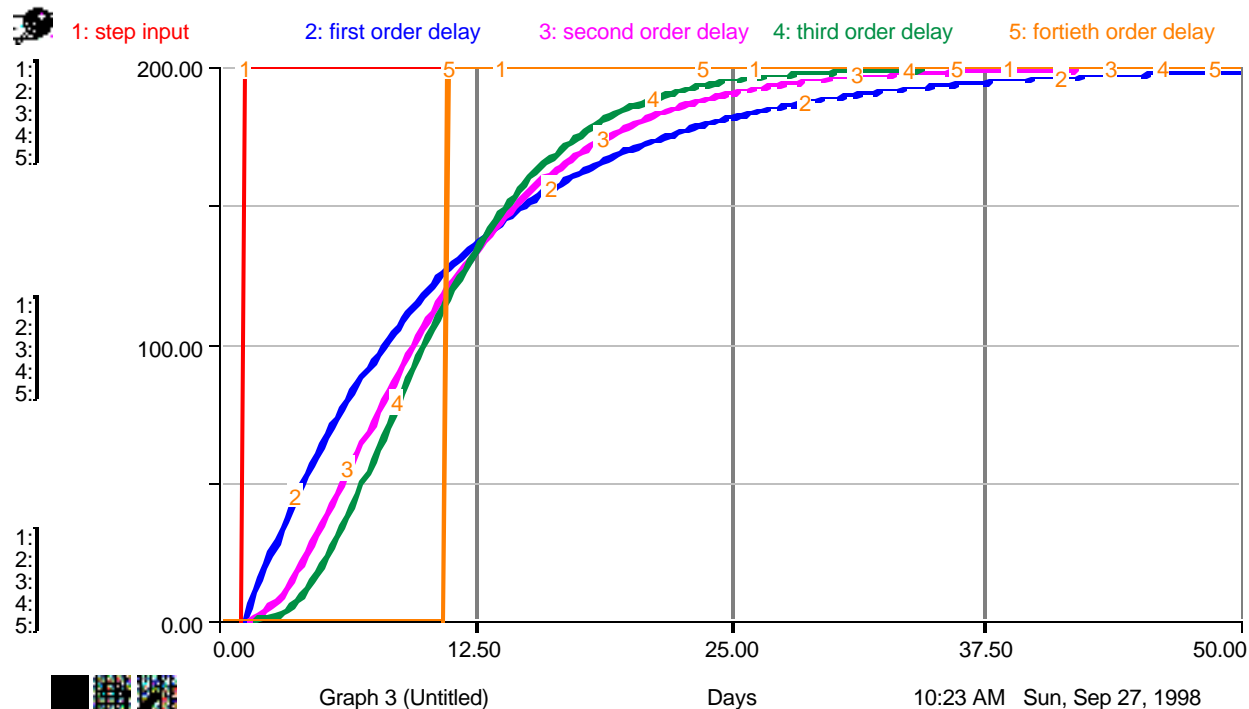
Delay Responses to Pulse Input

- First, second and third order delay responses to a pulse input

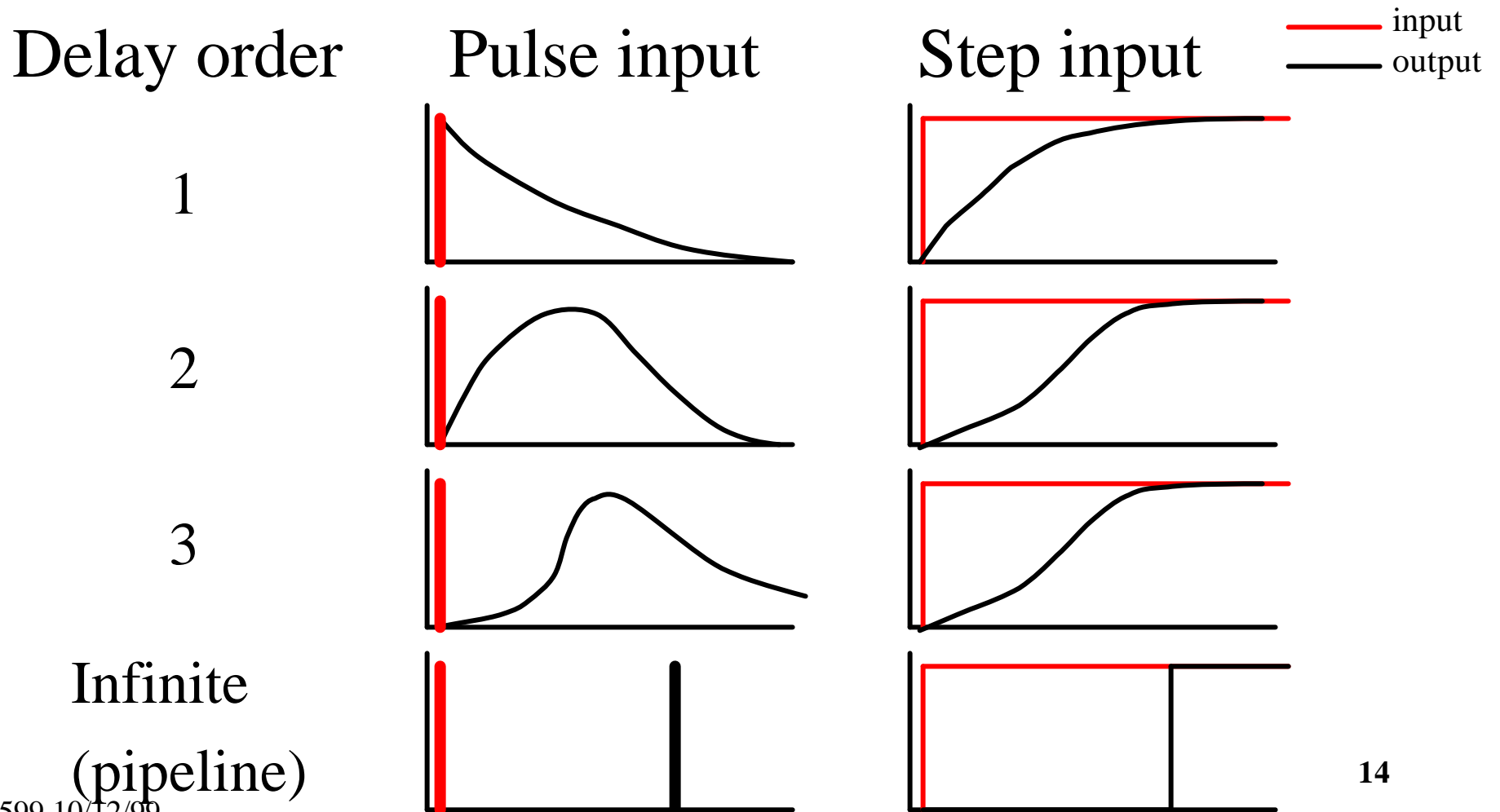


Delay Responses to Step Input

- First, second and third order delay responses to a step input



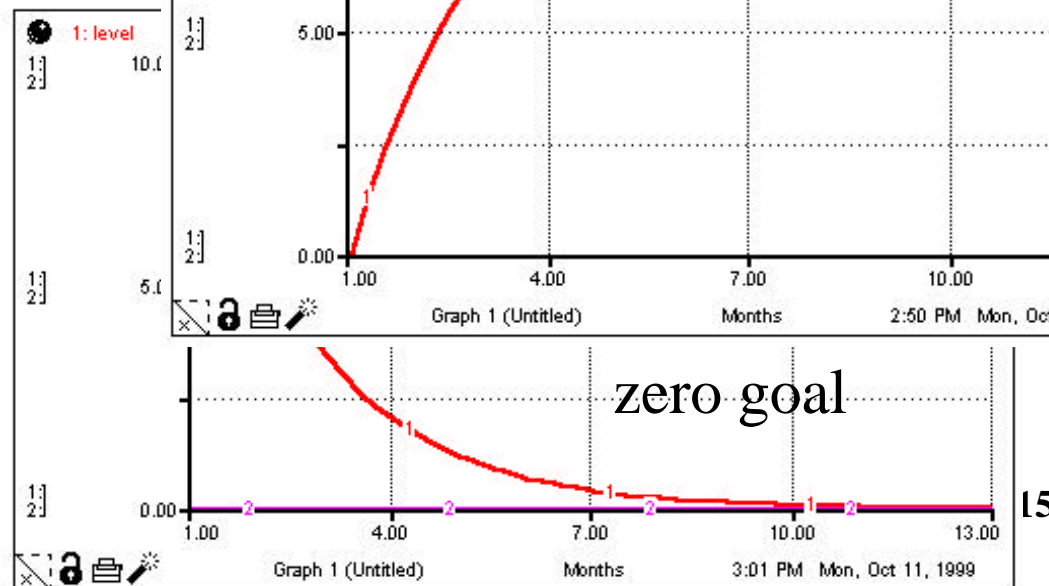
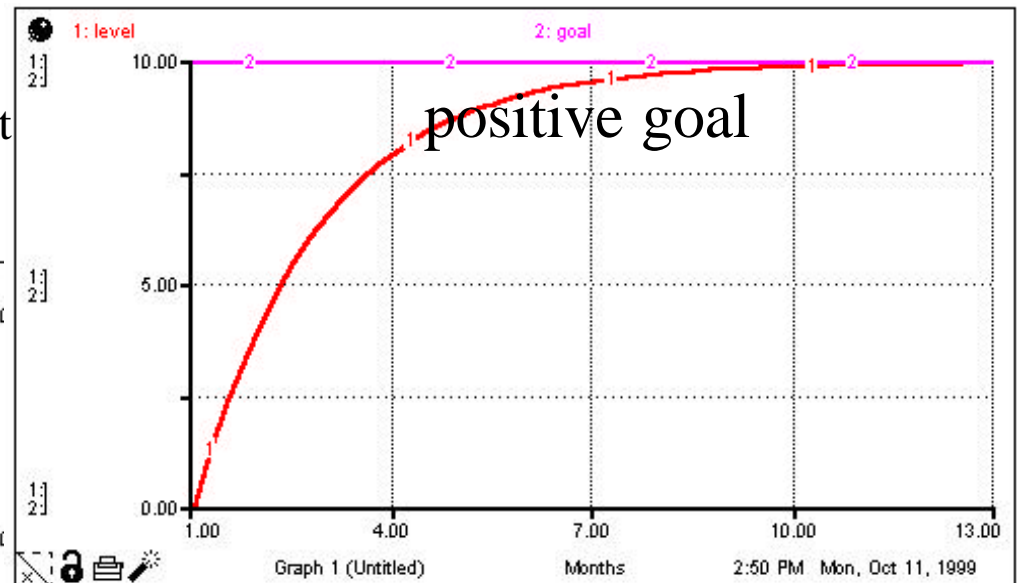
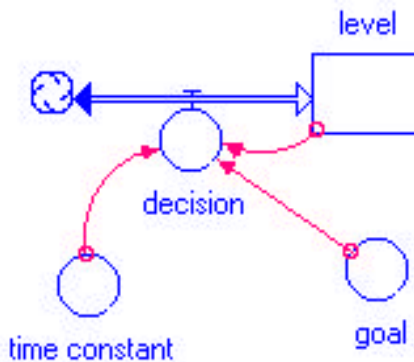
Delay Summary



Negative Feedback

- Negative feedback exhibits goal seeking behavior, or sometimes instability
- May represent hiring increase towards a staffing goal. The change is more rapid at first and slows down as the discrepancy between desired and perceived decreases. Also a good trend for residual defect levels.

- $\text{rate} = (\text{goal} - \text{present level}) / \text{time constant}$

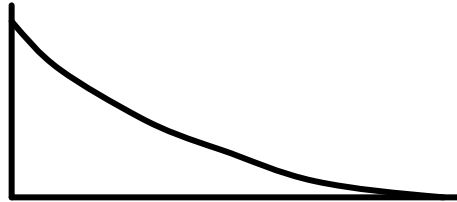


Analytically:

$$\text{Level} = \text{Goal} + (\text{Level}_0 - \text{Goal})e^{-t/tc}$$

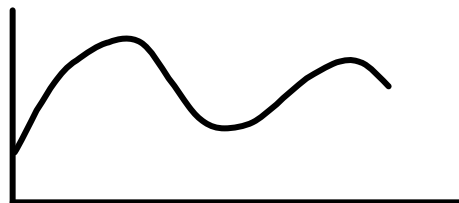
Orders of Negative Feedback

- *First-order Negative Feedback*



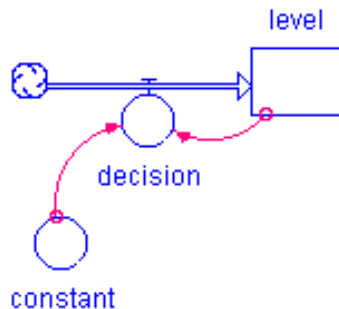
- *Second-order Negative Feedback*

- Oscillating behavior may start out with exponential growth and level out. It could represent the early sales growth of a software product that stagnates due to satisfied market demand, competition or declining product quality.



Positive Feedback

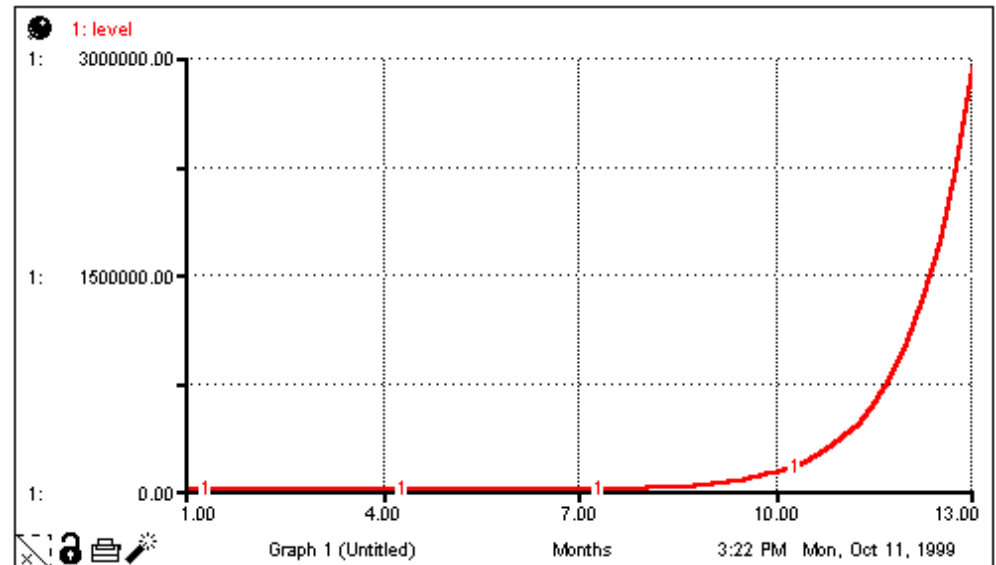
- Positive feedback produces a growth process
- Exponential growth may represent sales growth (up to a point), Internet traffic, defect fixing costs over time
- $\text{rate} = \text{present level} * \text{constant}$



Analytically:

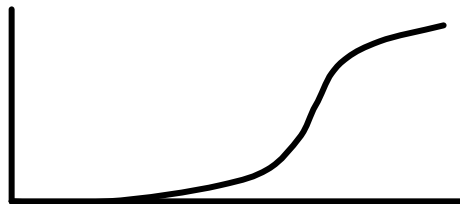
exponential growth: $\text{Level} = \text{Level}_0 e^{at}$

exponential decay: $\text{Level} = \text{Level}_0 e^{-t/TC}$



S-Curves

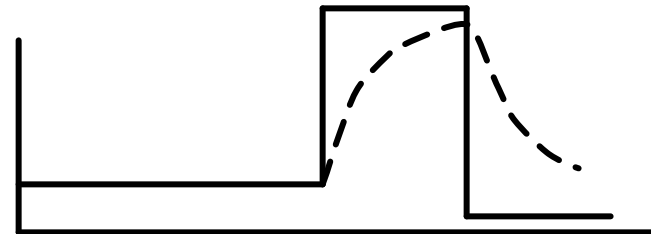
- S-curve: graphic display of a quantity like progress or cumulative effort plotted against time that exhibits an s-shaped curve. It is flatter at the beginning and end, and steeper in the middle. It is produced on a project that starts slowly, accelerates and then tails off as work tapers off
- S-curves are also observed in the ROI curve of technology adoption, either time-based return or in production functions that relate ROI to investment.





Information Smoothing

- Smoothed variables exponentially seek the input signal





Rayleigh Manpower Distribution

- Rayleigh curve is a popular model of personnel loading
- Assumptions:
 - Only a small number of people are needed at the beginning of a project to carry out planning and specification. As the project progresses and more detailed work is required, the number of staff builds up to a peak. After implementation and unit testing is complete, the number of staff required starts to fall until the product is delivered.
 - The number of people working on a project is approximately proportional to the number of problems ready for solution at that time



Rayleigh Formula

- A Rayleigh curve describes the rate of change of manpower effort per the following first order differential equation:

$$\frac{dC(t)}{dt} = p(t)[K - C(t)]$$

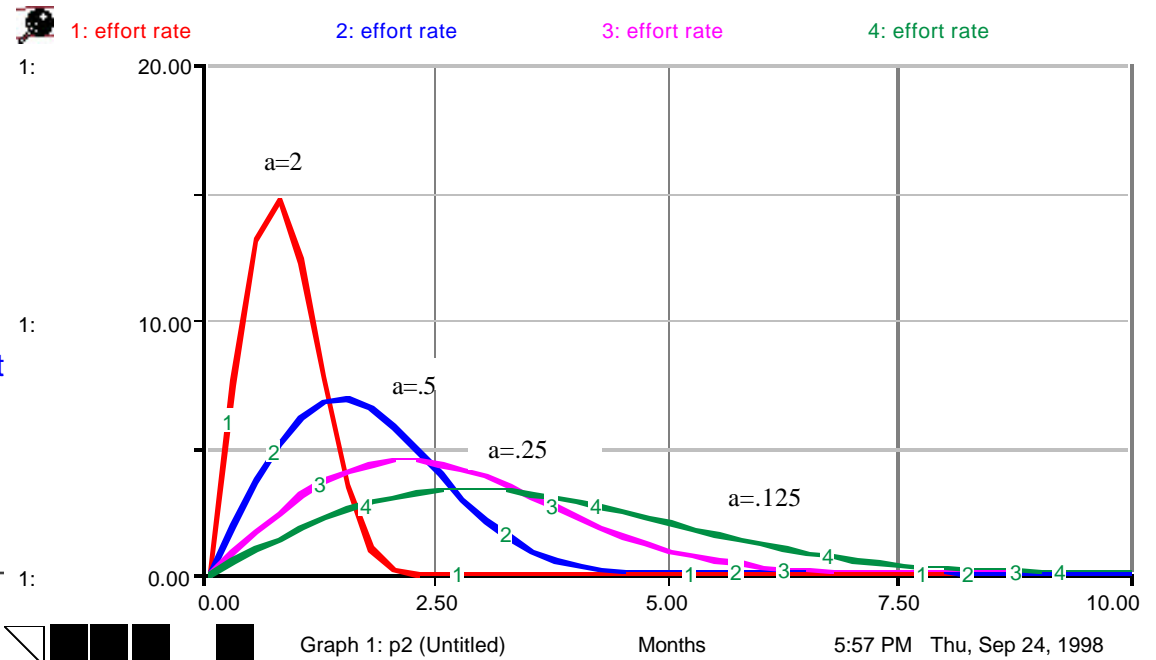
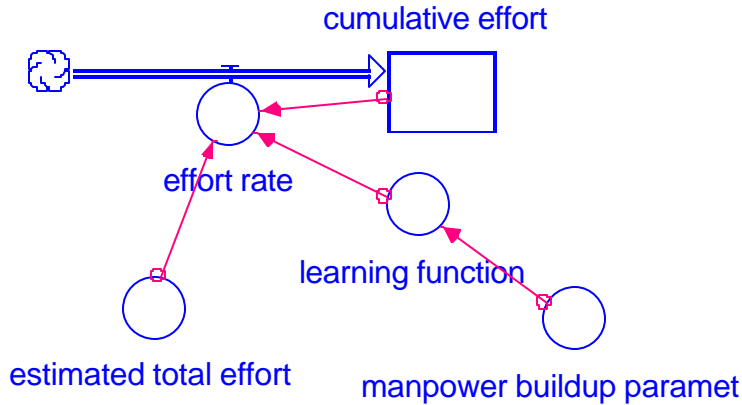
where $C(t)$ is the cumulative effort at time t , K is the total effort, and $p(t)$ is a learning function. The learning function is linear and can be represented by

$$p(t) = 2at$$

where a is a positive number.

- The manpower rate of change represents the number of people involved in development at any time (staffing profile).
- The a parameter is an important determinant of the peak personnel loading called the manpower buildup parameter.

Rayleigh Model



$\text{cumulative_effort}(t) = \text{cumulative_effort}(t - dt) + (\text{effort_rate}) * dt$
 INIT cumulative_effort = 0
 INFLOWS:
⚙ effort_rate = learning_function*(estimated_total_effort-cumulative_effort)

estimated_total_effort = 15
 DOCUMENT: Estimated total effort for the project.

learning_function = manpower_buildup_parameter*time
 DOCUMENT: The Norden learning function.

manpower_buildup_parameter = .5
 DOCUMENT: The manpower buildup parameter determines the steepness of buildup and the peak personnel loading.



Interactive Rayleigh Model Demo

- Vary manpower buildup parameter
- Show effect of midstream added requirements
- Demonstrate S-curve



Homework

- Prepare introduction to term project briefing
- Complete external process concurrence homework