System Dynamics Group
Sloan School of Management
Massachusetts Institute of Technology

System Dynamics for Business Policy, 15.874
Professors John Sterman, Brad Morrison

Assignment 6
Structure and Behavior of Delays

Assigned: Thursday 13 October 2003; Due: Tuesday 4 November 2003

This assignment should be done in a team totaling three people.
This assignment will be graded pass-fail.

Delays are a critical source of dynamics in nearly all systems. Thus far in our modeling, however, we have represented delays in causal diagrams qualitatively. In this assignment you explore the structure and behavior of delays and test their responses to a range of inputs. The assignment helps you understand the dynamics of delays so that you can use them appropriately in more complex models. The assignment also develops your skills in model formulation and analysis. To do this assignment it is essential that you read chapter 11 in the text.

A. Material Delays

❑ A1. Do the challenge on page 425 of the text (Response of Material Delays to Steps, Ramps, and Cycles). Note that you are asked to:

❑ Sketch your intuitive estimate of the response of a first order material delay to the four test inputs shown in Figure 11-9.

☛ To assist you, Figure 11-9 is reproduced on the following pages so you can sketch your estimates directly on the graphs.

☛ Your grade will not be affected by your answer to this part of the question. Please ***be honest*** and sketch your responses ***before*** simulating the model; this is important for building your intuition about delays.

❑ A2. After sketching your intuitive estimates, build and simulate the model for the first-order material delay.

☛ The structure and equations for a first-order delay are described in the text (section 11.2.4). Build the delay explicitly as a stock and flow structure shown in Figure 11-4 (that is, do not use Vensim's built-in function DELAY1).

☛ Set the initial value of the stock of letters in transit so that the delay is always initialized in equilibrium, independent of the initial value of the input. In equilibrium, the stock in transit is unchanging, so the inflow and outflow to the

Originally prepared by John Sterman; revised by Brad Morrison, October 2003.

❑ denotes a question for which you must hand in an answer, a model, or a plot.
☛ denotes a tip to help you build the model or answer the question.

delay must be equal. Solve this equation for the equilibrium value for the stock in transit, and enter this expression as the initial condition for the stock in your model. Run your model with a constant input to confirm that it begins and remains in equilibrium. You do not need to hand in the equilibrium run.

☛ We have created a “test input generator” that will make it easy for you to generate the step, ramp, sine wave, and other test inputs you need to run the tests. Download the model, TestGen.mdl, from the course website and use it to generate the input patterns you need. The test input generator is fully documented and dimensionally consistent (see the appendix).

☛ To learn more about the parameters for the built-in functions used in the Test Input Generator (specified in the Appendix) and in the following exercises, use the pull-down “Help” menu in Vensim. Click on “Keyword Search...” and type in the function name (e.g., STEP), and hit Enter.

☛ Note that the test generator has an initial value of one, so the exponential growth rate you need to replicate the test input in the graphs is 0.05/day. The ramp slope needed is 0.05/day.

☛ Set the initial time to –5 days, the final time to 25 days, and the Time Step (dt) to 0.125 days. Be sure to save output every time step (in the **Time Bounds** tab under **Settings…**, in the **Model** menu).

☛ To show the relationship between the input and output, define a custom graph that shows the input and output rate on the same graph, and with the same scale. Also use the strip graph tools to examine the behavior of the stock in transit.

☛ It may be helpful to envision the delay as the post office, with the input representing the mailing rate, the output representing the delivery rate, and the stock in transit representing the letters in the post office system.

❑ After you run the model for each of the test inputs, compare the results to your intuitive estimates and comment briefly. Were your mental simulations correct? Why? / Why not? In particular,

1. Consider the linear ramp. The response of any system to a shock consists of a *transient*, during which the relationship of input and output is changing, and a *steady state* in which the relationship between input and output is no longer changing (even though the input and output might both be growing, for example). Does the output of the delay (e.g., the delivery rate of letters) equal the input (e.g., the mailing rate) in the steady state for the linear ramp? Explain.

2. How does the response to the sine wave depend on the delay time relative to the period of the cycle? In particular, as the delay time increases relative to the period of the cycle, what happens to the amplitude and timing of the output?

A3. ❑ Next, repeat the steps above (*both* mental and formal simulation) for a third order material delay with the same average delay time of 5 days.

☛ Build the delay explicitly as shown in section 11.2.5. Do not use the Vensim built-in function DELAY3. Figure 11-6 shows a second order material delay; the third order delay is analogous but contains three intermediate stocks of material in transit (see equation 11-4).

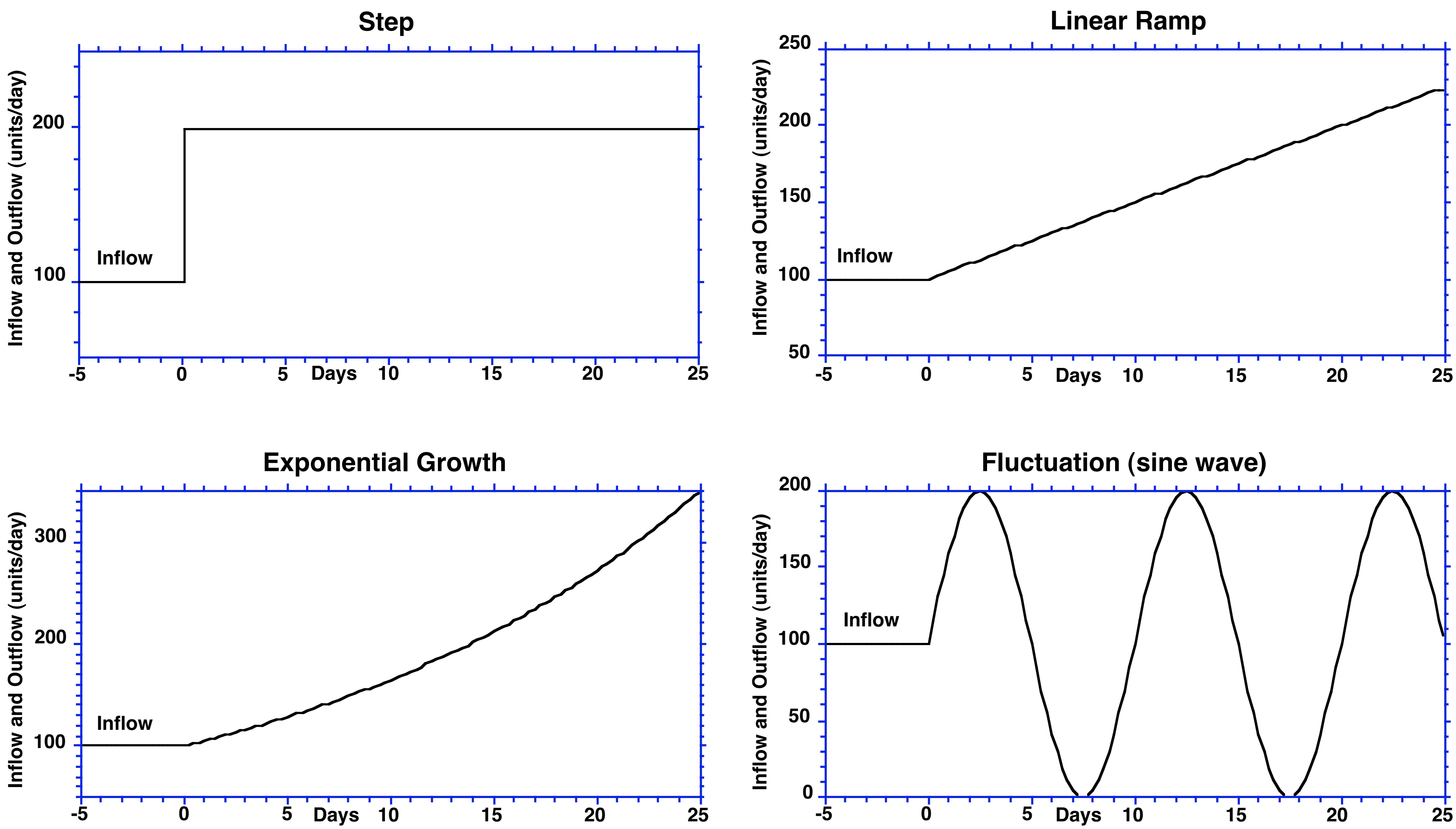
☛ Build the third order delay in the same model as your first-order delay and use the same input test generator. This allows you to compare the behavior of the third order delay to the first order delay.

A4. ❑ Next, repeat the steps above (again *both* mental and formal simulation) for a pipeline delay with the same average delay time of 5 days.

☛ Build the pipeline delay using the DELAYFIXED function in Vensim. The pipeline delay is described in section 11.2.3. To access the DELAYFIXED function from the variable definition dialog in VensimPLE, click on the tab labeled Functions; you will then see an alphabetical list of all the built in functions available in PLE. The syntax for the DELAYFIXED function is:

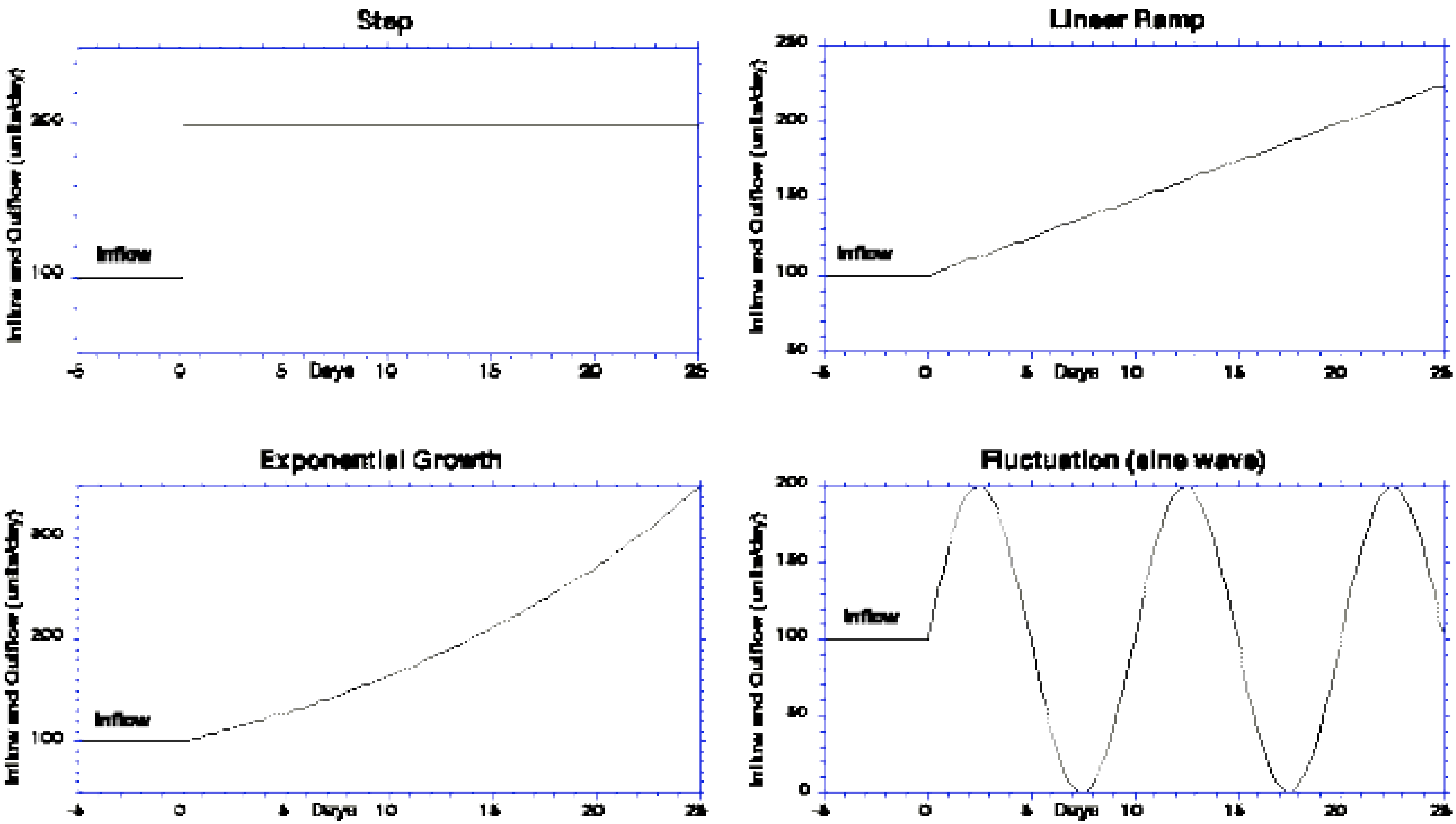
OUTPUT = DELAYFIXED(INPUT, DELAY TIME, INITIAL OUTPUT)

☛ Build the pipeline delay in the same model as your other delays so you can easily compare the output of the three delay types.



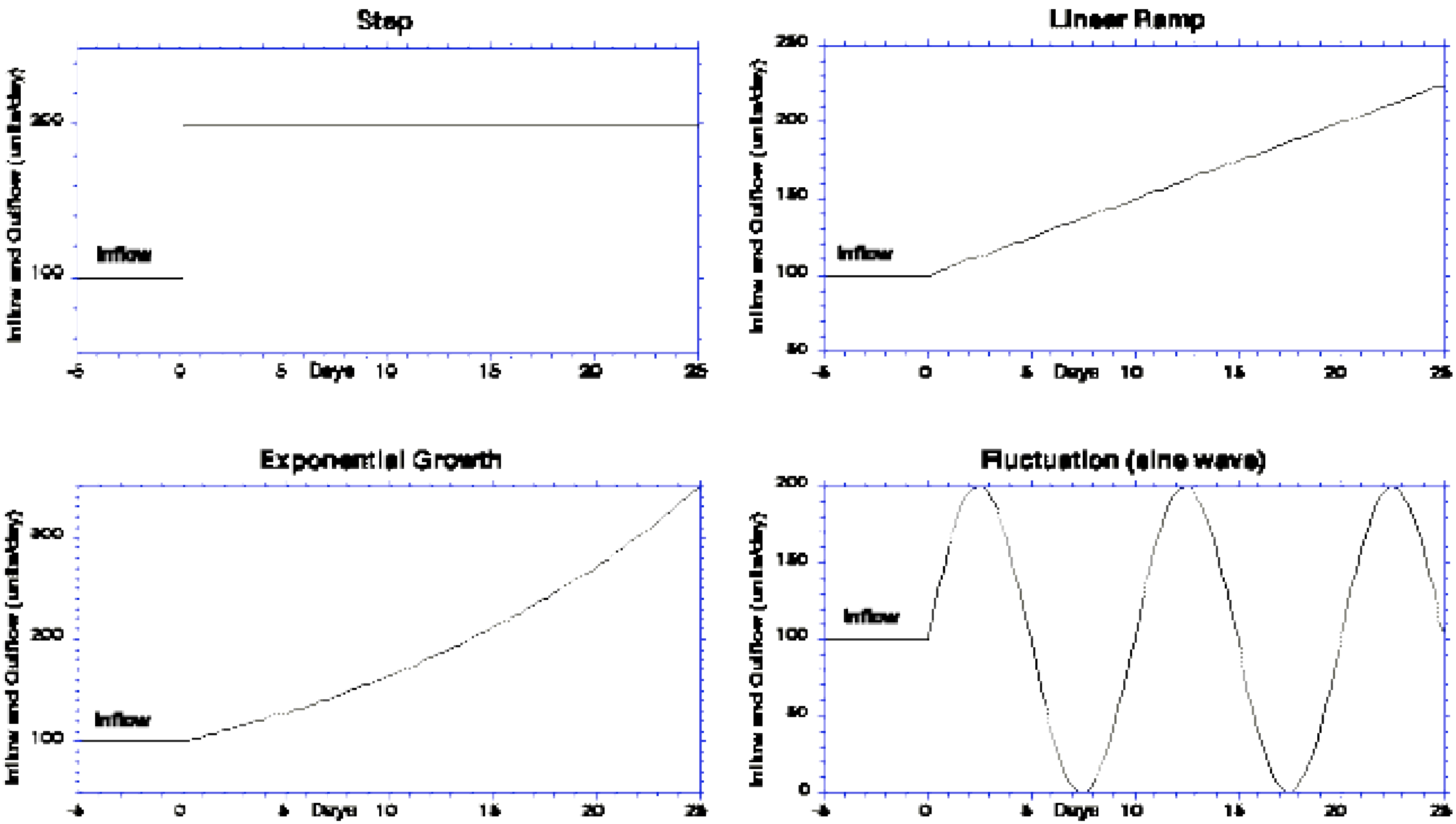
Four test inputs to a delay

Before simulating the model, sketch the response of a ***first order material delay*** to each of these inputs, assuming a 5 day average delay time.



Four test inputs to a delay

Before simulating the model, sketch the response of a ***third order material delay*** to each of these inputs, assuming a 5 day average delay time.

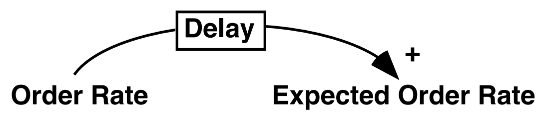


Four test inputs to a delay

Before simulating the model, sketch the response of a ***Pipeline delay*** to each of these inputs, assuming a 5 day average delay time.

B. Information Delays

In the previous sections we considered material delays, where the input to the delay is a physical inflow of items to a stock of units in transit, and the outflow is the physical flow of items exiting the stock. Many delays, however, exist in channels of information feedback—for example, in the measurement or perception of a variable, such as the order rate for a product.

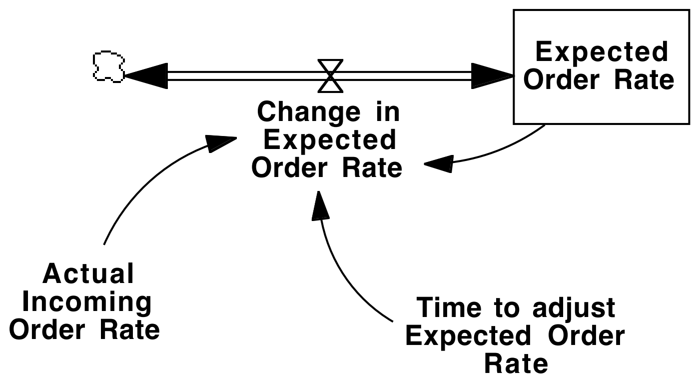


In this example, the expected order rate is management's belief about the value of the order rate. The expected order rate lags behind the actual order rate due to delays in measuring and reporting orders, and due to the time it takes managers to update their beliefs about the order rate. The physical order rate does not flow into the delay; rather information about the order rate enters the delay. For reasons you will discover below, a different structure is needed to represent these information delays.

As an example of an information delay, consider the way a firm forecasts demand for its products. Why does forecasting inevitably involve a delay? Firms must forecast demand because it takes time to adjust production to changes in demand, and because it is costly to make large changes in production. They don't want to respond to temporary changes in demand but only to sustained new trends. A good forecasting procedure should filter out random changes in incoming orders to avoid costly and unnecessary changes in output (setups, changeovers, hiring and firing, overtime, etc.) while still responding quickly to changes in trends to avoid costly stockouts and lost business. To do so, firms constantly revise their forecasts as conditions change.

Consider the stream of successive forecasts rather than any particular forecast. Even though the firm is trying to anticipate the future order rate, the only information available upon which to base a forecast is information about the current or past behavior of the system. Since it takes time to gather the information required to forecast, and since it takes time to decide whether a change in the current order rate heralds a new trend or represents a random change that will rapidly reverse, changes in the forecast will lag behind changes in actual conditions. That is, all forecasts create delays in decision making. The challenge is to respond to changing rates without overreacting to noise, to tell which change in demand is the beginning of a new trend and which a mere random blip.

One of the most widely used forecasting techniques is called "exponential smoothing" or "adaptive expectations." As discussed in section 11.3.1, "Adaptive expectations" means the forecast adjusts (adapts) gradually to the actual stream of orders for the product. If the forecast is persistently wrong, it will gradually adjust until the error is eliminated. The structure of an exponential smoothing delay is shown in Figure 11-10. Specifying this structure for the case of a demand forecast leads to the following:



The equation for the rate at which expectations adapt—the rate at which the forecast is revised—is:

d(EOR)/dt = (IOR - EOR)/TEOR

where

EOR = Expected Order Rate (widgets/month)
IOR = Actual Incoming Order Rate (widgets/month)
TEOR = Time to adjust the Expected Order Rate (months)

The "time constant" TEOR represents the time required, on average, for expectations to respond to a change in actual conditions. The forecast, EOR, is a *stock*, since the forecast is a state of the system, in this case representing a state of mind of the managers regarding what the future order rate will be. This stock remains at its current value until there is some reason to change it. In adaptive expectations, the stock (the managers' belief about the future order rate) changes when there is a difference between the current order rate and their belief about the expected order rate. This structure is known as a first-order information delay, or 'first-order exponential smoothing'.

Build a model to represent this forecasting procedure. Your model should include the equations above exactly. Do not add any additional structure.

- Assume TEOR is 6 months.
- Assume the order rate IOR is exogenous. Use the Test Input Generator in the Appendix to determine the incoming order rate. You will have to change the units for time in the test generator from days to months.
- Set the initial value of the forecast to 100 widgets/month.
- Use a simulation time step of 0.25 months and simulate the model for 60 months.
- Create a custom graph that traces the behavior of both IOR and EOR.

❑ B1. What kind of feedback loop does the forecasting procedure represent?

B2.

❑ a. What are the units for EOR?

❑ b. What are the units for the flow into EOR?

B3. Run your model with a random input to demand. Set the Noise Standard Deviation to 0.50, and the Noise Start Time to 5 months. That is, the noise should have a standard deviation of 50% of the initial input.

❑ a. About how much of the random noise is filtered out by exponential smoothing?

❑ b. How does the amount of noise filtered out depend on the value of the adjustment time TEOR?

❑ c. To filter out random variation in incoming orders, should the adjustment time be long or short?

B4. Consider the response to a step input. Set the Step Height to 0.50 and the Step Time to 5 months.

❑ a. About how long does it take for adaptive expectations to adjust about two-thirds of the way?

❑ b. About how long does it take for adaptive expectations to adjust 85% of the way? 95% of the way?

❑ c. What is the relationship between these adjustment times and TEOR?

❑ d. Does the size of the step play a role in the adjustment time?

☛ Review section 8.3 (especially 8.3.1) for information about the adjustment time and its function in a negative feedback system.

B5. Investigate what happens when the order rate is growing linearly. ***Before you simulate the model,*** make a sketch of the forecast EOR over time for the case where the actual order rate, from an initial equilibrium, suddenly starts rising linearly (following a ramp input). Make your sketch ***before*** you run the model. *Your grade will not be affected by your answer to this question*. Sketching the response you anticipate before running the model helps make your mental model and intuitive understanding explicit and *speeds your learning*. After you've drawn the behavior you expect, run the model with a ramp input with Ramp Slope = 0.10 and Ramp Start Time = 5 months.

❑ a. Explain the behavior and compare it to your expectations (include your graph of expectations, even though your grade will not depend on its accuracy).

❑ b. Is the forecast accurate in the steady state? Why or why not? A persistent difference between the forecast and the actual order rate in the steady state is known as "steady state error."

❑ c. Use the equation for the change in the expected order rate to derive a simple expression for the steady state expected order rate as a function of the actual order rate, the slope of the ramp, and the adjustment time. In steady state, what is the rate of change of the forecast?

B6. Now test the response of the exponential smoothing structure to a fluctuating input. Set the amplitude of the sine wave to 0.50 and the period to 12 months, to approximate a seasonal cycle in demand.

❑ a. What is the steady state behavior of the forecast?

❑ b. How does the amplitude and timing of the forecast compare to that of the input?

❑ c. How the amplitude and timing of the forecast depend on the forecast adjustment time?

B7. Based on your experiments, consider exponential smoothing as a forecasting technique.

❑ a. Under what circumstances (that is, what demand characteristics) would exponential smoothing be a useful forecasting technique?

❑ b. Under what circumstances would it be ineffective or even damaging?

❑ c. Under what circumstances would you want to have a long forecast adjustment time? A short adjustment time?

C. Response to Changing Delay Times

Until now all our tests have considered the response of different delays to changing inputs. We have assumed the delay time has remained constant. Now we consider what happens when the delay time changes.

The delay times for both material and information delays can change. For example, raising the speed limit on the interstate highways from 55 to 65 reduced the delay in the transport of raw materials from supplier to customer (assuming any truckers were actually obeying the 55 mph speed limit in the first place). Similarly, replacing a paper-based invoicing system with a globally integrated, real-time client-server network can reduce the delay in the perception by senior management of the order rate for their firm's products.

❑ C1. Do the Challenge "Response of Delays to Changing Delay Times" on p. 435 of the text.

☛ For part 1 and part 2 of the challenge, do not use the computer. Be sure to sketch the behavior you expect for each case before you build and simulate the models.

❑ Hand in the graphs showing your estimate of the behavior for each case. *Your grade does not depend on the accuracy of your answer to this question.* Explain your graphs briefly.

❑ C2. For part 3 of the challenge you will simulate the effect of the changes in delay times described in parts 1 and 2. To do so you need to modify the equation for the delay time in your models so that it can change.

To implement the test, modify the equation for the delay time to:

Delay Time = IF_THEN_ELSE(Time < Time to Change Delay Time,
Base Delay Time, New Delay Time)

where the Time to Change Delay Time is the time at which the value of the delay time will switch from the Base Delay Time to the New Delay Time.

☛ The challenge does not require you to experiment with changing delay times for the pipeline delay.

❑ Hand in plots of the behavior of the first- and third-order material delay when the delay time changes from 5 to 10 days, and from 5 to 2.5 days (all changes occur on day 5).

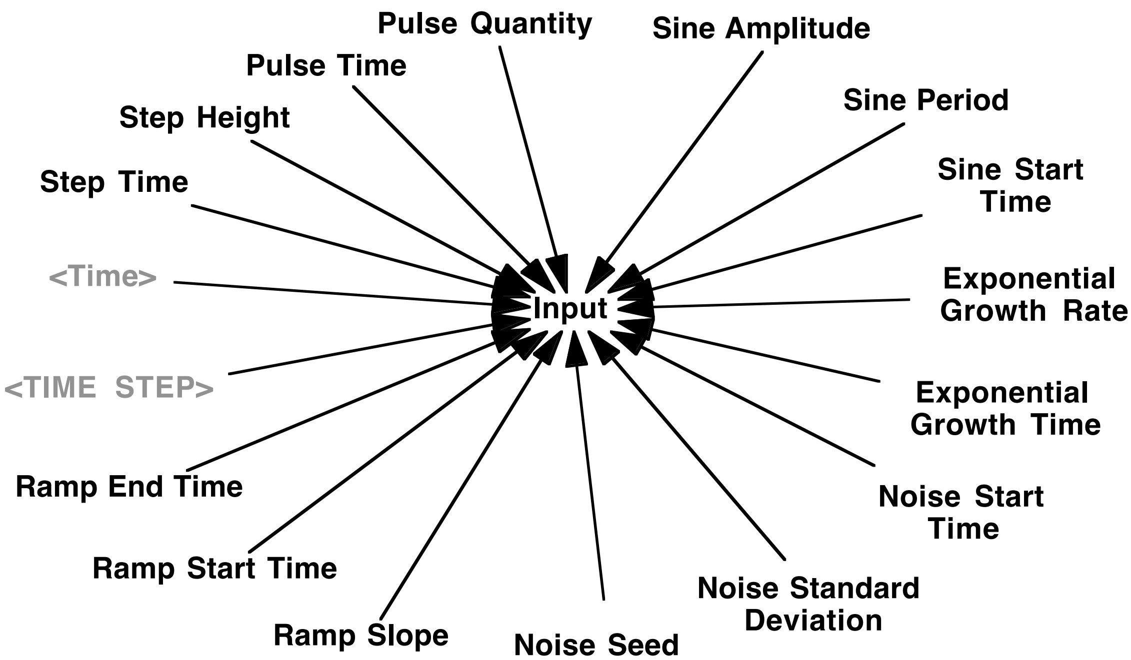
Was your intuition correct? Explain the dynamics and the equilibrium state (the magnitude of the stock in transit). If there are differences between the response of the material and information delays, explain why.

❑ C3. *0 points--unless you don't document your model, in which case the grader may deduct an arbitrary number of points*. Hand in the diagrams and documented equation listings for your models (including the first order, third order, and pipeline material delays, and for the information smoothing delay).

Appendix: Test Inputs

System dynamics modeling software provides functions for the most commonly used test inputs: step, pulse, ramp, sine wave, exponential, and uncorrelated (or “white”) noise, etc. These inputs are not necessarily intended to correspond to anything that really happened in the past or that will happen in the future. Rather, the inputs are designed to help you easily explore the dynamics of a model. With a very simple input, it is easy to see the dynamics generated by the model. More complicated input patterns, such as actual historical data, make it difficult to isolate the behavior generated by the model’s structure from the input pattern. Once the dynamics of the structure are understood, it is usually possible to grasp how the structure will behave with more complicated inputs such as the actual historical input data.

A useful “Test Input Generator” is provided by the following structure. Note: When using this in other models, be sure to check the units for time and modify if necessary, along with the default parameters.



Equations:

```
Input=
       1+STEP(Step Height,Step Time)+
       (Pulse Quantity/TIME STEP)*PULSE(Pulse Time,TIME STEP)+
       RAMP(Ramp Slope,Ramp Start Time,Ramp End Time)+
       STEP(1,Exponential Growth Time)*(EXP(Exponential Growth Rate*Time)-1)+
       STEP(1,Sine Start Time)*Sine Amplitude*SIN(2*3.14159*Time/Sine Period)+
       STEP(1,Noise Start Time)*RANDOM NORMAL(-4, 4, 0, Noise Standard
         Deviation, Noise Seed)

       ~ Dimensionless
       ~ The test input can be configured to generate a step, pulse, linear
ramp, exponential growth, sine wave, and random variation.  The initial value
of the input is 1 and each test input begins at a particular start time.  The
magnitudes are expressed as fractions of the initial value.
```

```
Step Height=0
       ~ Dimensionless
       ~ The height of the step increase in the input.

Step Time=0
       ~ Day
       ~ The time at which the step increase in the input occurs.

Pulse Quantity=0
       ~ Dimensionless*Day
       ~ The quantity added to the input at the pulse time.

Pulse Time=0
       ~ Day
       ~ The time at which the pulse increase in the input occurs.

Ramp Slope=0
       ~ 1/Day
       ~ The slope of the linear ramp in the input.

Ramp Start Time=0
       ~ Day
       ~ The time at which the ramp in the input begins.

Ramp End Time=1e+009
       ~ Day
       ~ The end time for the ramp input.

Exponential Growth Rate=0
       ~ 1/Day
       ~ The exponential growth rate in the input.

Exponential Growth Time=0
       ~ Day
       ~ The time at which the exponential growth in the input begins.

Sine Start Time=0
       ~ Day
       ~ The time at which the sine wave fluctuation in the input begins.

Sine Amplitude=0
       ~ Dimensionless
       ~ The amplitude of the sine wave in the input.

Sine Period=10
       ~ Day
       ~ The period of the sine wave in the input.

Noise Seed=1000
       ~ Dimensionless
       ~ Varying the random number seed changes the sequence of realizations
            for the random variable.

Noise Standard Deviation=0
       ~ Dimensionless
       ~ The standard deviation in the random noise.  The random fluctuation
            is drawn from a normal distribution with min and max values of +/-
            4.  The user can also specify the random number seed to replicate
```

```
            simulations.  To generate a different random number sequence,
            change the random number seed.

Noise Start Time=0
	~	Day
	~	The time at which the random noise in the input begins.

********************************************************
	.Control
********************************************************~
		Simulation Control Parameters
	|

FINAL TIME  = 25
	~	Day
	~	The final time for the simulation.

INITIAL TIME  = -5
	~	Day
	~	The initial time for the simulation.

SAVEPER  = 
        TIME STEP
	~	Day
	~	The frequency with which output is stored.

TIME STEP  = 0.125
	~	Day
	~	The time step for the simulation.
```

A Note on Random Noise

The random input is useful to simulate unpredictable shocks. The RANDOM_NORMAL function in Vensim samples from a normal distribution with parameters the user specifies. The function has the following syntax:

RANDOM_NORMAL (min, max, mean, std dev, seed)

Vensim uses a default random number "seed." You can specify a different seed by defining a constant called "Noise Seed" in your model and setting it equal to some value (e.g. Noise Seed = 1000). Vensim generates a single random sequence for any given seed. Let's say the sequence is: 0.500, 0.213, 0.678, 0.932, 0.340, 0.015. If there is a single random number function in the model it will simply yield the random sequence. If there are two or more random functions, the functions will take turns accessing the sequence. For example, if you have two functions, the first will yield 0.5, 0.678, 0.34; and the second will yield 0.213, 0.932, 0.015. If you run two simulations with the same seed, you will get exactly the same sequence of random numbers. This is important so that you can compare two runs with different policies and be sure the differences in behavior are due only to the policies and not to different realizations of the random number generator. When you do want to examine runs with different realizations of the random process, you need to change the value of the random number seed.

Note also that the use of a function such as RANDOM_NORMAL means a new random number is selected every time step. Cutting the time step in half would then double the number of random shocks to which the model is subjected, and increase the highest frequency represented in the random signal. This is generally not good modeling practice. In realistic models, one must not only select the standard deviation of any random processes, but also specify its frequency spectrum (or, equivalently, the autocorrelation function). Failure to do so can lead to spurious results and make your model overly sensitive to the time step. These issues are discussed in Appendix B.

A Note on the Pulse Function

The pulse function is used to simulate the effect of instantaneously adding a fixed quantity Q to a variable. To ensure the entire quantity is added all at once (within a single time step, or DT [delta time]), the duration of the pulse is set to the smallest interval of time in the model, that is, to the time step DT. The height of the pulse is then the quantity to be added divided by the time step in the model, Q/DT. The inflow increases by the height of the pulse and remains at the higher level for one time step, so that the total quantity added to the accumulation is (Q/DT)*DT = Q units.