Guided Study Program in System Dynamics
System Dynamics in Education Project
System Dynamics Group
MIT Sloan School of Management[1]

Solutions to Assignment #30
Saturday, September 4, 1999

Reading Assignment:

Please download and read the following paper from http://sysdyn.mit.edu/gsp98/ :

- *Properties of Damped Oscillations Systems, by Helen Zhu (D-4767)*

Exercises:

1. Properties of Damped Oscillations Systems

Please read "Properties of Damped Oscillations Systems" (D-4767). This paper dampens the first model in "Oscillating Systems II: Sustained Oscillations" (D-4602). In this exercise, you study a damped version of the second model in "Oscillating Systems II: Sustained Oscillations" (D-4602).

After reading "Properties of Damped Oscillations Systems" (D-4767) please refer back to "Oscillating Systems II: Sustained Oscillations" (D-4602), and re-read the description of the Cleanliness of a College Dorm Room model.

Kevin's roommate is not the only one who grows sick of seeing all of Kevin's laundry on the floor. Eventually Kevin realizes that he himself is unable to find the papers that he needs for class or the pizza that he had ordered the night before. Kevin can really only tolerate a maximum of eight articles of clothing on the floor before his messiness begins to impede his lifestyle. When Kevin changes his clothes, he either drops them on the floor or, if he feels that his floor is too cluttered, goes to the trouble of putting them away.

A. Change the Cleanliness of a College Dorm Room model to reflect Kevin's reaction to excessive laundry buildup. In your assignment solutions document, include the model diagram, documented equations, and graphs of the model behavior. Explain the

---

behavior of the model in one or two paragraphs, contrasting the behavior of this model and the behavior of the model in the paper.

Model diagram:
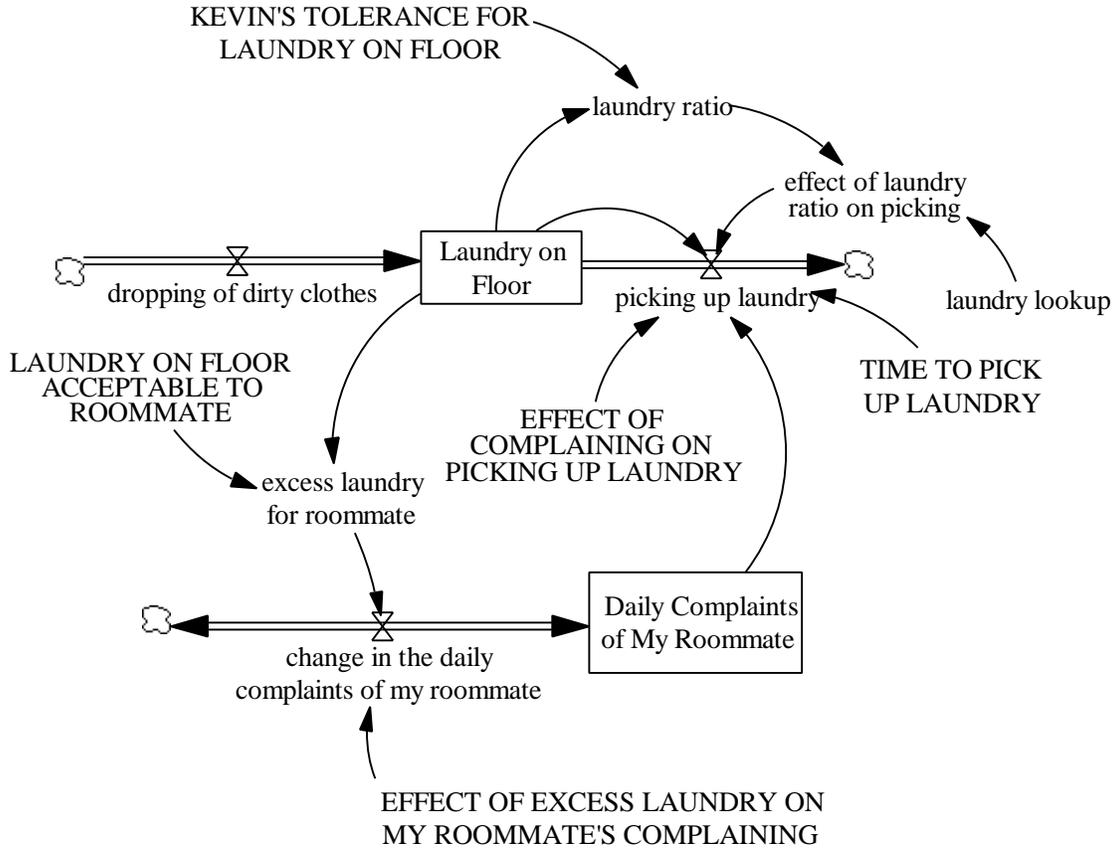
Model equations:

change in the daily complaints of my roommate = EFFECT OF EXCESS LAUNDRY ON MY ROOMMATE'S COMPLAINING * excess laundry for roommate
Units: (Complaints / day) / day
This flow represents how my roommate's amount of complaining changes over time. It is a function of the amount of excess laundry on the floor.

Daily Complaints of My Roommate = INTEG (change in the daily complaints of my roommate, 3)
Units: Complaints/ day
The number of complaints my roommate registers with me each day about the cleanliness of the room.

dropping of dirty clothes = 5
Units: Clothes / day

The number of dirty clothes I drop on my floor every day. The model assumes that my roommate's complaining or my own tolerance for laundry do not stop me from dropping all my clothes on the floor, they only change how many I pick up.

EFFECT OF COMPLAINING ON PICKING UP LAUNDRY = 1
> Units: Clothes / Complaints
> This variable is the number of extra clothes I will pick up each day if my roommate increases his complaining by one complaint per day.

EFFECT OF EXCESS LAUNDRY ON MY ROOMMATE'S COMPLAINING = 1
> Units: ((Complaints / day) / day) / Clothes
> This constant reflects how my roommate increases his complaining based on the addition of one more article of clothing to the floor.

effect of laundry ratio on picking = laundry lookup (laundry ratio)
> Units: dmnl
> The number of clothes picked up as a function of the laundry ratio.

excess laundry for roommate = Laundry on Floor - LAUNDRY ON FLOOR
> ACCEPTABLE TO ROOMMATE
> Units: Clothes
> This variable is the difference between the number of articles of clothing on my floor and the number of articles acceptable to my roommate.

KEVIN'S TOLERANCE FOR LAUNDRY ON FLOOR = 8
> Units: Clothes
> The maximum number of clothes I can tolerate on my floor.

laundry lookup ([(0,0) - (2,5)], (0,0), (0.2,0.05), (0.4,0.2), (0.6,0.4), (0.8,0.65), (1,1), (1.2,1.4), (1.4,2), (1.6,2.8), (1.8,3.8), (2,5))
> Units: dmnl
> The lookup function for the effect of laundry ratio on picking up laundry.

Laundry on Floor = INTEG (dropping of dirty clothes - picking up laundry, 6)
> Units: Clothes
> The number of articles of clothing on my dormitory floor.

LAUNDRY ON FLOOR ACCEPTABLE TO ROOMMATE = 3
> Units: Clothes
> This is the number of clothes on the floor my roommate finds acceptable (because they don't spill over onto his side of the room).

laundry ratio = Laundry on Floor / KEVIN'S TOLERANCE FOR LAUNDRY ON
> FLOOR
> Units: dmnl

The ratio of the laundry on the floor to the amount of laundry that Kevin can tolerate.

picking up laundry = (EFFECT OF COMPLAINING ON PICKING UP LAUNDRY *
    Daily Complaints of My Roommate) + (effect of laundry ratio on picking *
    Laundry on Floor / TIME TO PICK UP LAUNDRY)
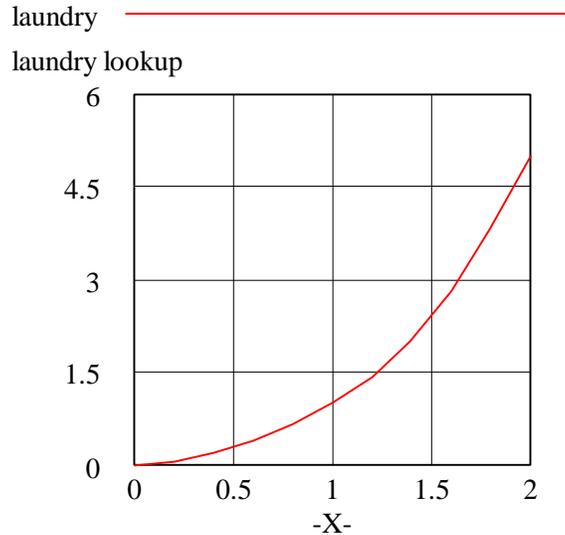   Units: Clothes / day
    The number of clothes I pick up each day. It is a function of how many
    complaints my roommate registers with me. It is also a function of the ratio of the
    number of clothes that lie on the floor to the number of clothes I can tolerate.

TIME TO PICK UP LAUNDRY = 1
    Units: day
    Kevin picks up his clothes once a day.

laundry ─────────────────────────

laundry lookup

Kevin's reaction to excessive laundry buildup is modeled through the "effect of laundry ratio on picking" multiplier. Kevin takes the ratio of the amount of "Laundry on Floor" to the amount he can tolerate, and the "laundry ratio" determines the fraction of "Laundry on Floor" that he picks up. The "laundry lookup" table function shows that as the "laundry ratio" increases, Kevin picks up an increasingly higher fraction of the "Laundry on Floor." When "Laundry on Floor" is small compared to Kevin's tolerance, the "effect of laundry ratio on picking" is small. When "Laundry on Floor" equals "KEVIN'S TOLERANCE FOR LAUNDRY ON FLOOR," he picks up all his laundry over a "TIME TO PICK UP LAUNDRY" equal to 1 day. As "Laundry on Floor" exceeds his tolerance, Kevin becomes increasingly unhappy about the mess in his room, so he picks up more and more laundry.

Model behavior:

The graph below shows the behavior of the system when Kevin does not care how many clothes pile up on the floor. As expected, the system demonstrates sustained oscillations.

This situation can be set up by multiplying the second part of the "picking up laundry" rate equation by 0, which basically implies that no matter how many clothes accumulate on the floor, Kevin does not pick them up. Thus, picking clothes up becomes a function of roommate complaints only:

picking up laundry = (EFFECT OF COMPLAINING ON PICKING UP LAUNDRY *
      Daily Complaints of My Roommate) + (effect of laundry ratio on picking *
   Laundry on Floor / TIME TO PICK UP LAUNDRY) * 0

Complaints and Laundry - Kevin indifferent

8	Complaints/day
6	Clothes
4	Complaints/day
3	Clothes
0	Complaints/day
0	Clothes

Day: 0 5 10 15 20
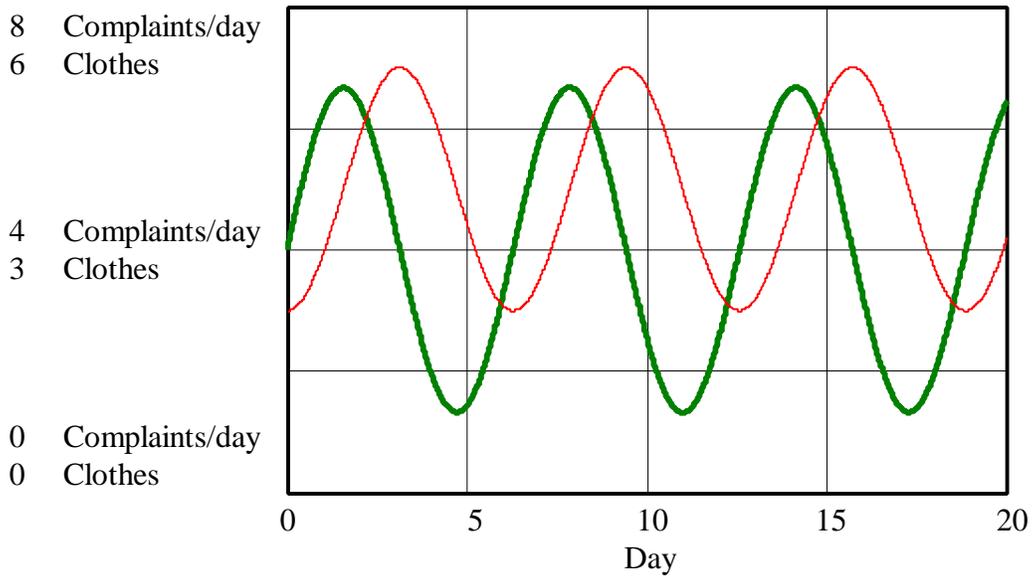
Daily Complaints of My Roommate : Kevin indifferent ——— Complaints/day
Laundry on Floor : Kevin indifferent ━━━ Clothes

When Kevin is bothered by the amount of clothes on his floor, we see the behavior below (here, the rate equation for "picking up laundry" is as shown in the documentation):

Complaints and Laundry - Kevin cares

6	Complaints/day
4	Clothes
4	Complaints/day
3	Clothes
2	Complaints/day
2	Clothes

Day
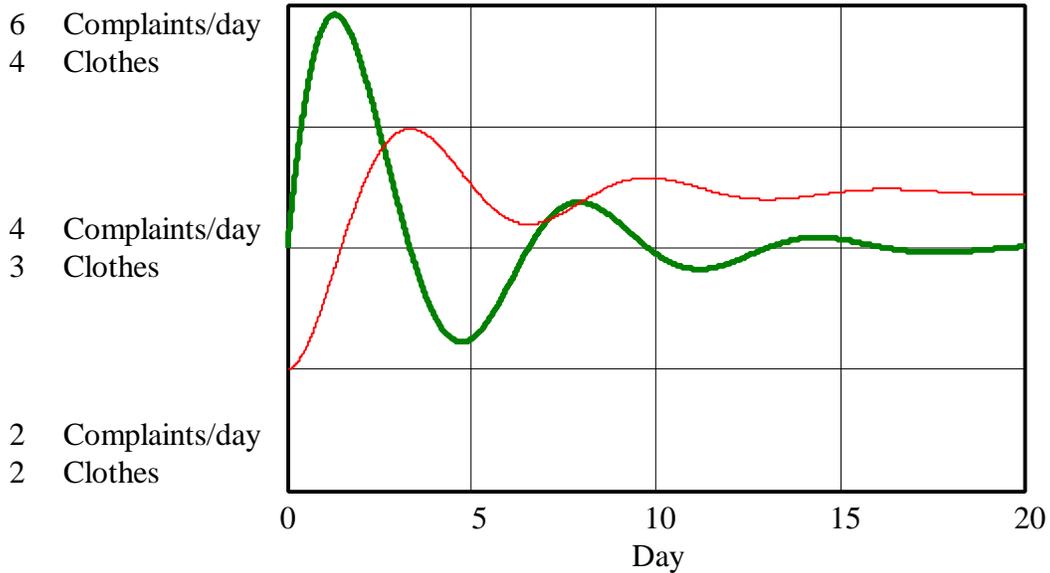
Daily Complaints of My Roommate : Kevin cares ———————— Complaints/day
Laundry on Floor : Kevin cares ▬▬▬▬▬▬▬▬▬ Clothes

B. How would the behavior of the model change if Kevin dropped his clothes on the floor more frequently? Why? Justify your answers with graphs of model behavior.

Increasing the frequency of dropping clothes on the floor (that is, increasing "dropping of dirty clothes") will increase the amplitude of oscillations. Increasing the "dropping of dirty clothes" rate causes the stock of "Laundry on Floor" to change more rapidly, so the stock increases to higher values before a corrective action is enforced. The graphs below show the behavior of the system with "dropping of dirty clothes" equal to 2 ("drop2" simulation run) and to 10 ("drop10" simulation run) clothes per day, compared with the base run behavior:

Laundry - change dropping rate
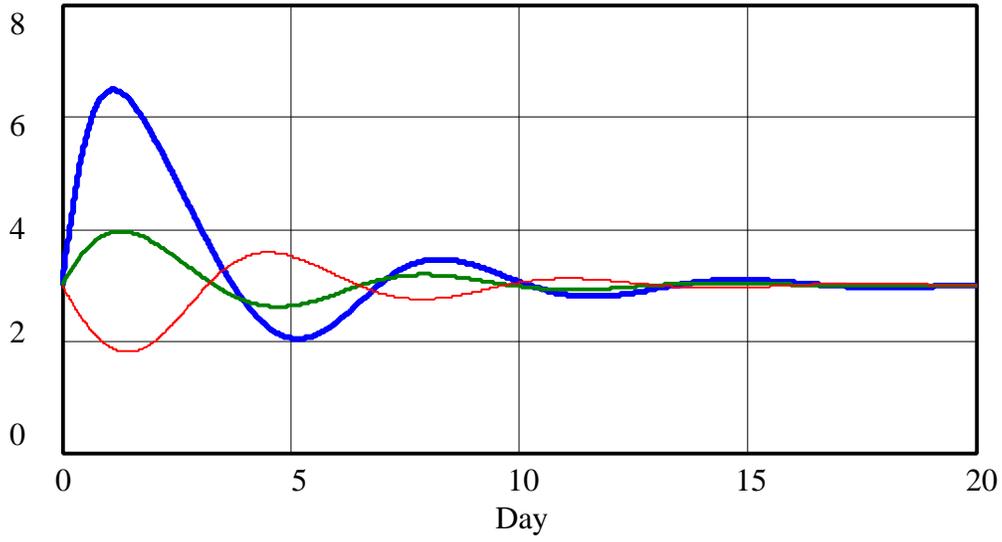
Laundry on Floor : drop2 ———— Clothes
Laundry on Floor : laundry ———— Clothes
Laundry on Floor : drop10 ———— Clothes

## Complaints - change dropping rate
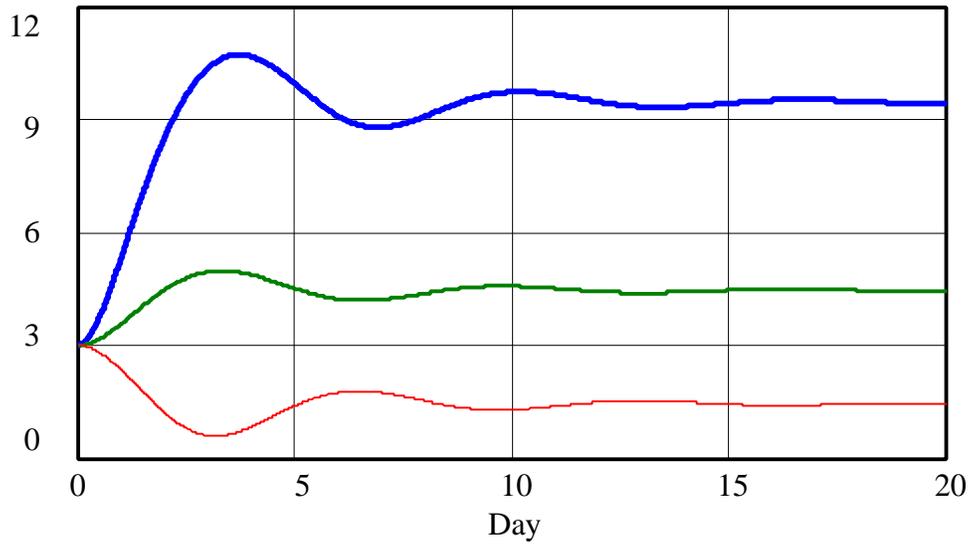
Daily Complaints of My Roommate : drop2 ———— Complaints/day
Daily Complaints of My Roommate : laundry ———— Complaints/day
Daily Complaints of My Roommate : drop10 ———— Complaints/day

C. How would the behavior of the model change if Kevin's tolerance for laundry on his floor increased? Why? Justify your answers with graphs of model behavior.

If Kevin's tolerance is increased, the amplitude of the oscillations will increase. Now, the corrective action that Kevin takes is delayed because he will wait until the pile of clothes is larger before picking up the clothes without complaints. The graphs below show the system behavior with "KEVIN'S TOLERANCE FOR LAUNDRY ON FLOOR" equal to 15 ("tolerance=15" simulation run) and to 20 ("tolerance=20" simulation run) clothes, compared with the base run behavior:

Laundry - change Kevin's tolerance

Laundry on Floor : laundry	————————	Clothes
Laundry on Floor : tolerance=15	————————	Clothes
Laundry on Floor : tolerance=20	————————	Clothes

Complaints - change Kevin's tolerance

Daily Complaints of My Roommate : laundry———— Complaints/day
Daily Complaints of My Roommate : tolerance=15———— Complaints/day
Daily Complaints of My Roommate : tolerance=20———— Complaints/day
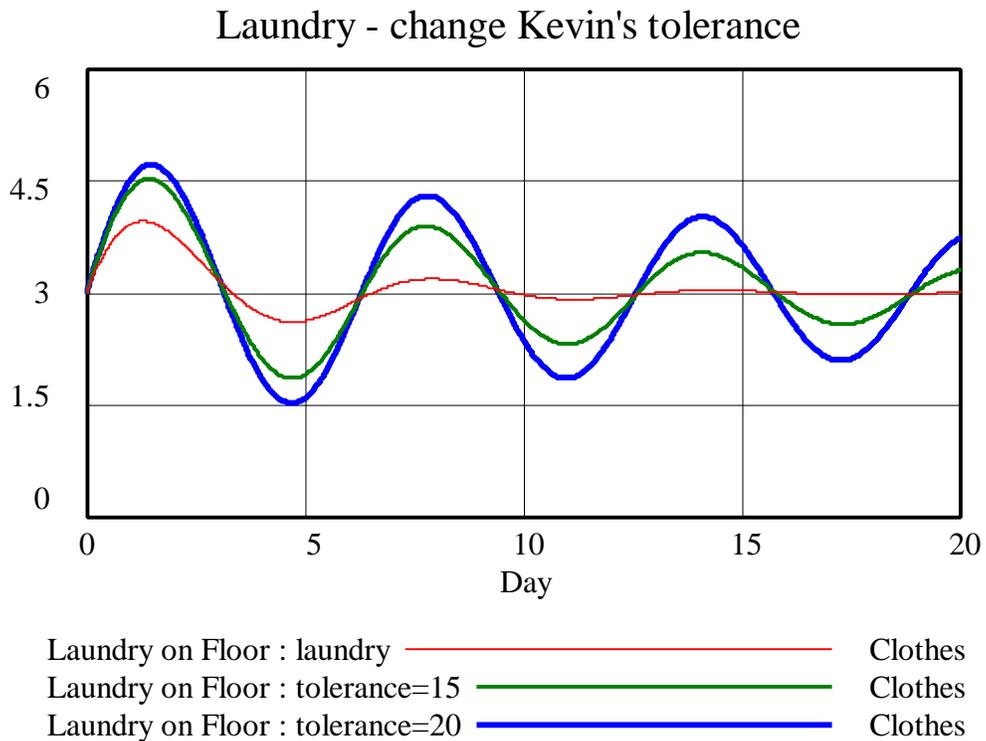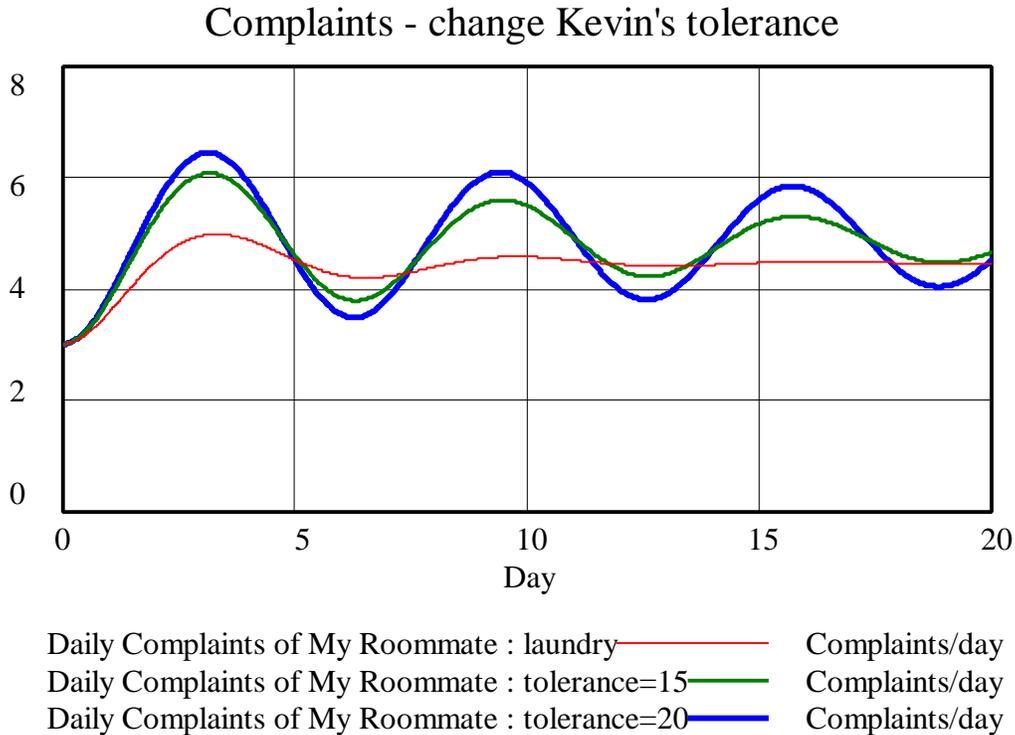
2. *Independent Modeling Exercise*

"Americans have been fighting wildfire full tilt for more than half a century. In those years we've created a powerful system built on brave young firefighters, old bombers, modern trucks, helicopters, and a bear named Smokey. But at the same time foresters and ecologists have been learning that too much fire fighting can be as bad as none. Today many of them agree that if we try to keep fires out of forests completely, dead wood and other fuels build up. Then, instead of low fires that just clear out brush, flames climb into the crowns of monarch trees and kill them. These fires burn so hot they leave total devastation. [...]

"Density and diversity   A healthy ponderosa forest is made of widely spaced, fire-resistant trees. With overprotection, young trees and competing species make a flammable understory so shaded that ponderosa seedlings can't grow.

"Forest floor   If burned often, a forest floor is a shallow layer of duff on soil that accepts ponderosa seeds. Overprotected, the floor builds deep layers of debris. Soil germination declines; fires burn hotter.

"How fires burn   Frequent fires move swiftly across the fire floor, killing few large trees. When fire is rare, accumulated fuels explode into towering crown fires, and the thick floor burns long, hot, and deep, killing roots of grasses and trees.

"Frequency of fire   *Scientists measure frequency by scars on trees or ash layers on the ground. Fire intervals vary: short in ponderosa forests and grasslands, long in coastal Douglas fir forests. Inevitably, this year or next century, fire returns.*

"Recovery from fire   *A healthy ponderosa forest recovers within three years, its roots intact, its trees barely scarred. In unfit forest, fire kills big trees; soil no longer absorbs rain and it erodes. Recovery may take centuries."*

excerpts from "The Essential Element of Fire" by Michael Parfit
*National Geographic, September 1996*

Conceptualize and formulate a system dynamics model that addresses the problem of overprotection of forests by demonstrating the long-term costs of fighting forest fires.

We would like to encourage you to begin by reading the above National Geographic article in its entirety, if possible. Feel free to also refer to other sources in order to conceptualize and formulate the model. In your assignment solutions document, include the model diagram, documented equations, and graphs of model behavior. In a few paragraphs, explain why the system and the model produce the observed behavior. In the model documentation, make sure to explicitly state where you obtained the data for the parameters in the model. If you estimate the parameters, include proof that your model generates consistent and realistic behavior over the range of realistic parameter values, and explain why that range is, indeed, realistic. Then use the model to propose a cost-minimizing policy for fighting forest fires that takes into account both short-term and long-term effects.

Model diagram:

The main purpose of the model is to study the effect of fires on young and old trees in a forest. The intensity of a fire depends on the amount of underbush on the ground. The hypothesis is that small, frequent fires are good for the trees in a forest, but infrequent, large fires can be very damaging.

There are many ways to model this system, depending the level of complexity required and what aspects of a forest one wishes to study. The model below is by no means the only one possible, or absolutely complete or correct, but it helps to study the hypothesis mentioned above.
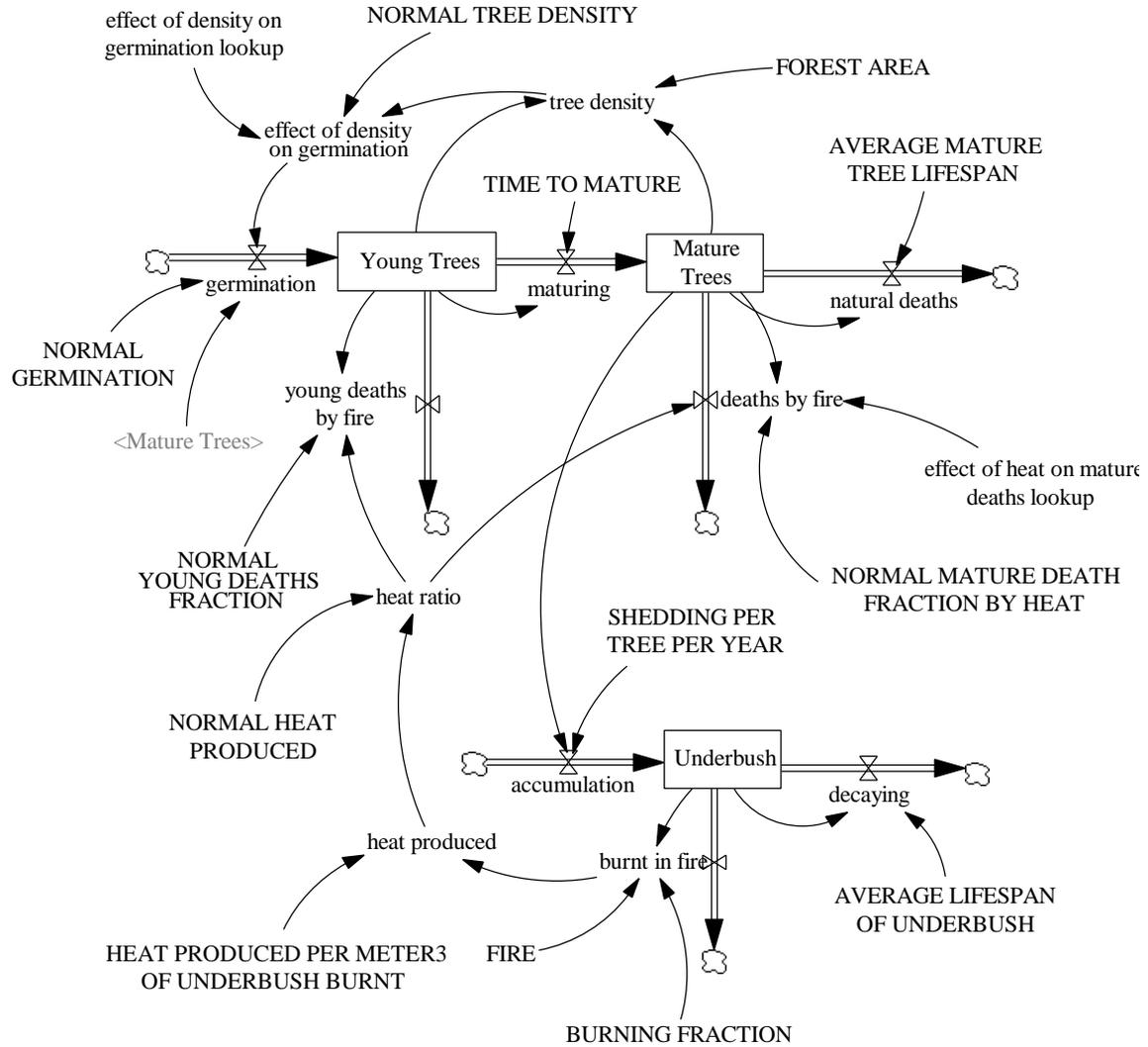
Model Equations:

accumulation = Mature Trees * SHEDDING PER TREE PER YEAR
> Units: meter3/year
> The amount of underbush that accumulates each year.

AVERAGE LIFESPAN OF UNDERBUSH = 0.25
> Units: year
> The average time it takes for leaves and branches to decay and clear away after they fall to the ground.

AVERAGE MATURE TREE LIFESPAN = 180
> Units: year
> The average life span of a mature tree. The total life span of a tree equals the average mature tree lifespan plus the time to mature.

BURNING FRACTION = 0.8
        Units: 1 / year
        The fraction of the existing underbush that will be destroyed by a fire in a year if
        the fire is not fought.

burnt in fire = BURNING FRACTION * Underbush * FIRE
        Units: meter3/year
        The amount of the underbush that is burnt in a fire.

deaths by fire = Mature Trees * NORMAL MATURE DEATH FRACTION BY HEAT *
        effect of heat on mature deaths lookup (heat ratio)
        Units: tree/year
        The number of mature trees killed by forest fires each year.

decaying = Underbush / AVERAGE LIFESPAN OF UNDERBUSH
        Units: meter3/year
        The amount of underbush that decays naturally each year.

effect of density on germination = effect of density on germination lookup (tree density /
        NORMAL TREE DENSITY)
        Units: dimensionless
        The effect of tree crowding on the germination rate.

effect of density on germination lookup ([(0,0) - (2.5,3)], (0,3), (0.25,3), (0.5,2),
        (0.75,1.4), (1,1), (1.5,0.5), (2,0.15), (2.5,0.05))
        Units: dimensionless
        The lookup function showing how the normal germination fraction varies as the
        ratio of density to normal density changes.

effect of heat on mature deaths lookup ([(0,0) - (6,5)], (0,0), (0.5,0.2), (1,1), (1.5,1.6),
        (2,2.2), (3,3.5), (5,4.5), (6,4.5))
        Units: dimensionless
        The lookup function showing the effect of heat produced by a fire on the fraction
        of mature trees killed by heat.

FIRE = (PULSE(20, 0.083) + PULSE(40,0.083) + PULSE(60,0.083) +
        PULSE(80,0.083) + PULSE(100,0.083) + PULSE(120,0.083))*0 +
        PULSE(120,0.5)*0
        Units: dimensionless
        This parameter represents the duration and pattern of fires in the forest. Thus,
        step, ramp and pulse functions of various duration and periodicity can be used to
        represent various types of fires and the ability of forest officials to fight the fires.
        The equation as given here is used in the base case, "no fires" simulation. (See
        Formulation Notes below.)

FOREST AREA = 1000

Units: acre
The area of the forest in consideration.

germination = Mature Trees * NORMAL GERMINATION * effect of density on
germination
Units: tree / year
The rate of new sapling growth.

heat produced = burnt in fire * HEAT PRODUCED PER METER3 OF UNDERBUSH
BURNT
Units: heat unit/year
The amount of heat produced by the fire.

HEAT PRODUCED PER METER3 OF UNDERBUSH BURNT = 1
Units: heat unit / meter3
The amount of heat produced when one meter cube of underbush is burnt in a fire.

heat ratio = heat produced / NORMAL HEAT PRODUCED
Units: dimensionless
The ratio of the heat produced by a fire to the heat produced by a "normal" fire.

Mature Trees = INTEG (maturing - deaths by fire - natural deaths, 27000)
Units: tree
The number of mature and old trees in the forest.

maturing = Young Trees / TIME TO MATURE
Units: tree/year
The number of young trees that become mature each year.

natural deaths = Mature Trees / AVERAGE MATURE TREE LIFESPAN
Units: tree/year
The number of old trees that die of natural causes each year.

NORMAL GERMINATION = 0.01
Units: 1 / year
A seed from an existing tree leads to the germination of a new sapling only twice
in the lifetime of a tree, that is, once every 100 years under normal tree density
conditions. Most seeds fall to the ground and die away.

NORMAL HEAT PRODUCED = 3000
Units: heat unit / year
The amount of heat produced by a "normal" fire in a year.

NORMAL MATURE DEATH FRACTION BY HEAT = 0.1
Units: 1/year

The fraction of old trees destroyed by a normal fire. A normal fire is too weak to destroy any trees, but can harm branches or parts of large trees.

NORMAL TREE DENSITY = 30
    Units: tree / acre
    The normal number of trees per acre in this forest.

NORMAL YOUNG DEATHS FRACTION = 0.8
    Units: 1/year
    The fraction of existing saplings destroyed by a normal fire per year.

SHEDDING PER TREE PER YEAR = 1.5
    Units: meter3 / (tree * year)
    The total compressed volume of the leaves and branches shed by a tree in a year.

TIME TO MATURE = 20
    Units: year
    A tree is considered to be young for the first 20 years of its life.

tree density = (Young Trees + Mature Trees) / FOREST AREA
    Units: tree / acre
    The average number of trees per acre in the forest.

Underbush = INTEG (accumulation - burnt in fire - decaying, 10000)
    Units: meter3
    The total volume of underbush in the forest.

young deaths by fire = heat ratio * NORMAL YOUNG DEATHS FRACTION * Young Trees
    Units: tree/year
    The number of young trees killed by a forest fire is directly proportional to the heat ratio.

Young Trees = INTEG (germination - maturing - young deaths by fire, 3000)
    Units: tree
    The number of young trees (less than 20 years old) in the forest.

no fire

effect of heat on mature deaths lookup

6

4.5

3

1.5

0

0   1.5   3   4.5   6

-X-
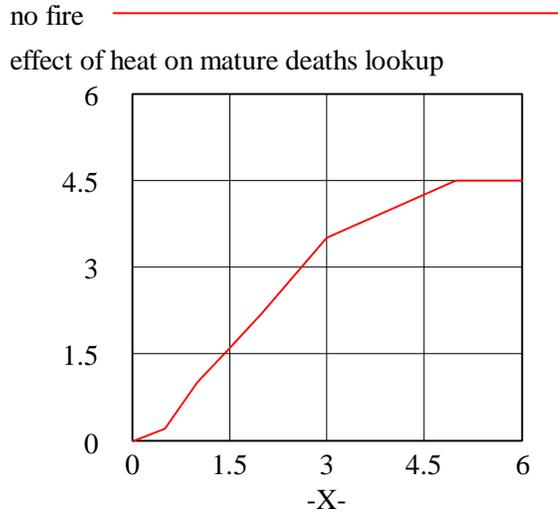
Formulation Notes:
The model makes certain assumptions that are evident in the formulation of the model.

1. The model defines a "normal fire," a fire that produces normal heat equal to 3000 heat units per year. One "heat unit" is the amount of heat produced when one meter cube of underbush is burnt.

- A normal fire, if allowed to burn for one year, will kill "NORMAL YOUNG DEATHS FRACTION" of young trees. It is assumed that a fire of different intensity will kill a linear multiple of the fraction of young trees killed by a normal fire. Thus, a fire twice as intense as a normal fire will kill twice as many young trees as a normal fire.

- The rate at which mature trees burn, however, is more complicated. A normal fire is too weak to significantly harm large, mature trees, and destroys only 10% of the mature trees if the normal fire burns for one year. As the intensity of the fire increases, however, the fraction of mature trees that die increases as indicated by the table function "effect of heat on mature deaths lookup."

2. Fires can be simulated using combinations of pulse, step and ramp functions.

- Frequent, short fires can be simulated by adding several PULSE functions that have a start time close to each other, and a short duration. For example:

FIRE = (PULSE(20, 0.083) + PULSE(40,0.083) + PULSE(60,0.083) + PULSE(80,0.083)
     + PULSE(100,0.083) + PULSE(120,0.083)) + PULSE(120,0.5)*0

creates six fires that start at 20 year intervals and each last for 0.083 of a year, or 1 month each.
The equation activates the first six, short-fire PULSE functions, but the last big fire is inactive. This equation is used in the simulation called "short fires" graphed below.

- Long, infrequent fires can similarly be simulated by using PULSE functions that are far apart and have a longer duration. For example:
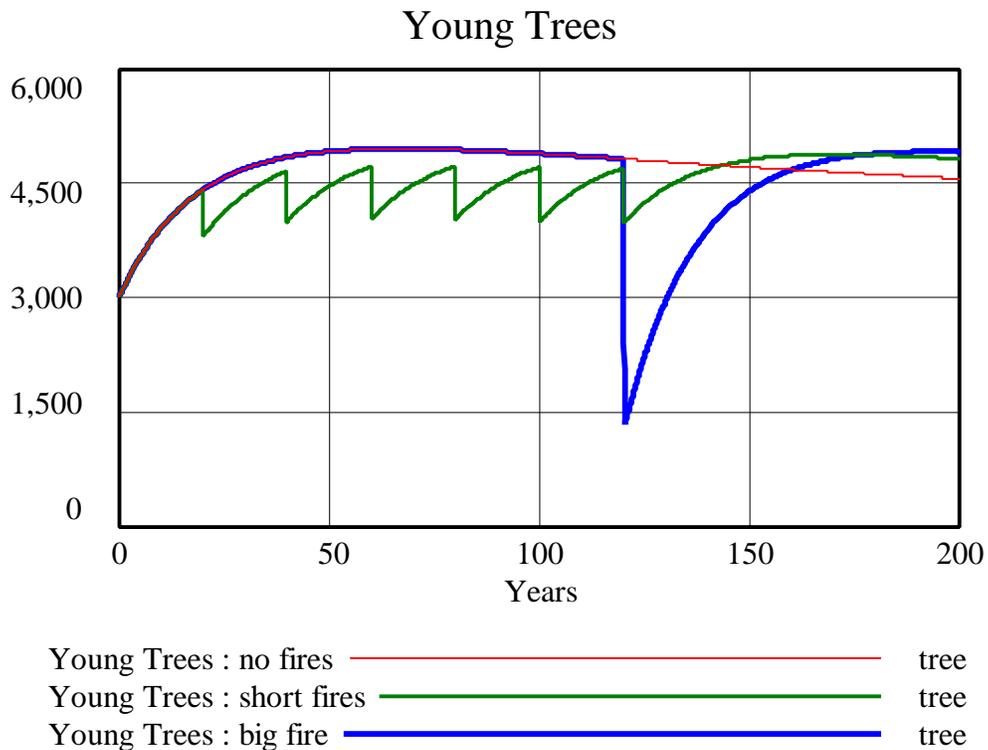
FIRE = (PULSE(20, 0.083) + PULSE(40,0.083) + PULSE(60,0.083) +
        PULSE(80,0.083) + PULSE(100,0.083) + PULSE(120,0.083))*0 +
        PULSE(120,0.5)

creates a fire that starts in the 120th year of the simulation and lasts for six months.
The above equation deactivates the first six, short-fire PULSE functions, and activates the
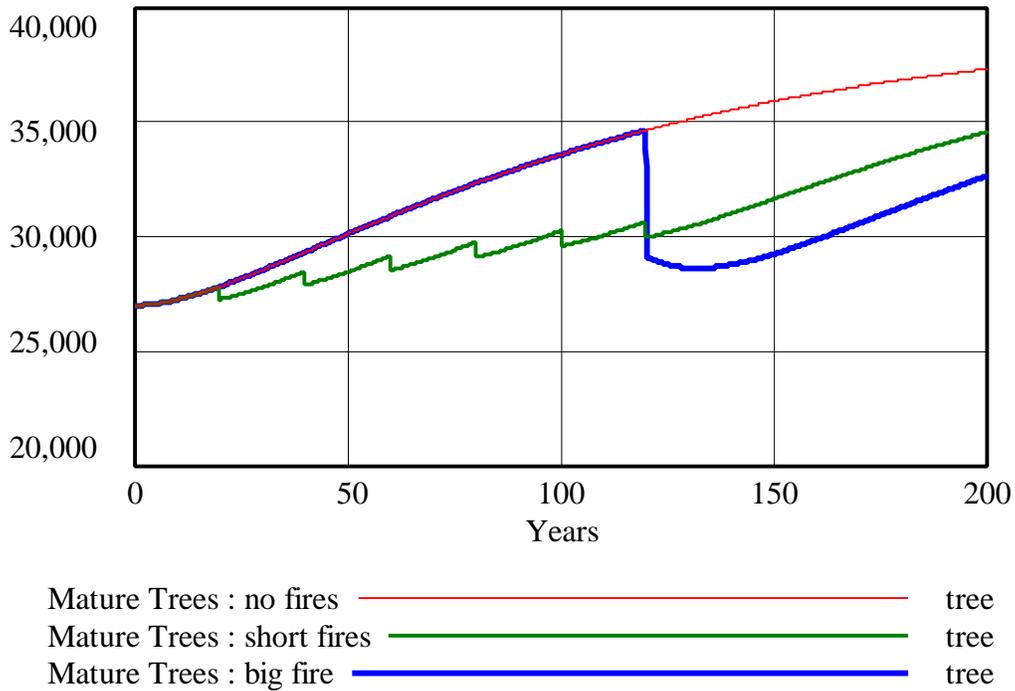last big fire. This equation is used in the simulation called "big fire" graphed below.)

Thus, both equations create a total of six months of burning during the simulation.

Model Behavior:

The graphs below show the behavior of saplings and mature trees over a period of 200
years. The simulation called "short fires" refers several short, frequent fires, while "big
fire" refers to the simulation with one large fire. The "no fires" simulation represents the
base case behavior.

Young Trees

Young Trees : no fires	tree
Young Trees : short fires	tree
Young Trees : big fire	tree

Mature Trees

Mature Trees : no fires	———————	tree
Mature Trees : short fires	———————	tree
Mature Trees : big fire	———————	tree

As the above graphs show, a large fire destroys a large fraction of mature and young trees. Small, frequent fires destroy small fractions of the trees at frequent intervals, but do not cause large-scale damage.

A large, sudden shock to the system can have several negative side effects not studied in this model. Large parts of the forest animal and bird life lose their natural homes. Fighting these large fires will probably be much more difficult and expensive than fighting the smaller ones. The heat and pollution produced by large fires can continue to affect the weather patterns of neighboring regions for years to come.

Thus, this model shows that fire fighters should not try to extinguish every fire as soon as the fire starts. Small fires should be allowed to burn for a while and clear some of the underbush before being put out. The regular burning of the underbush prevents underbush accumulation that could in the future lead to a major fire that can be very harmful.