CS 346 –
Computational Modeling & Simulation I
(Interdisciplinary Science)

Professor Eric Aaron

Lecture – T R 11:00am

Lecture Meeting Location: It’s complicated…

Lab Meeting Location: It’s complicated…

Business

• Reading: Ch. 2.2-2.3 in the textbook
  – Moving into Ch. 4 soon, maybe today!
• HW1 out now, due Mar. 10 (see assignment sheet for details)
  – This is phase 1 of a 2-phase assignment
  – Please submit work on time, so we can move on to the next phase (see assignment sheet for lateness policy)
• Reading assignment: Webb’s Can robots make good models of biological behaviour?
  – Available via Colby’s library; see link from CS346 Useful Links and Additional Notes
  – A small assignment asking you to read this paper and write short responses (a few sentences) to two questions is posted on course HWs page
  – Due by end of day Friday, Mar. 5
  – Please see email for unusual lateness policy, and please have it in no later than noon, March 10 (treat that deadline as firm)
Business, pt. 2

• Guest Lecture on March 16!
  – Speaker: John Long (Vassar College, Biology & Cognitive Science; author of *Darwin’s Devices*)
  – Topic: Creating and Testing Hypotheses

```plaintext
all_runs = 0;
for k = 1:num_runs
  sum = 0;
  i = 0;
  while sum <= 10
    sum = sum + rand;
    i = i + 1;
  end
  all_runs = all_runs + i;
end
average_iterations = all_runs / num_runs;
disp(average_iterations);
```

```plaintext
sumTotal2 = 0;
for i=1:numIterations
  rnc2 = 0;
  sum2 = 0;
  while sum2<=10
    x = rand;
    sum2 = sum2 + x;
    rnc2 = rnc2 + 1;
  end
  sumTotal2 = sumTotal2 + rnc2;
end
average2 = sumTotal2/numIterations;
disp(average2);
```

One of these two is noticeably slower than the other... do you see which? And what the reason is?
Sum: Subtle Inefficiency

```matlab
all_runs = 0;
for k = 1:num_runs
    sum = 0;
    i = 0;
    while sum <= 10
        sum = sum + rand;
        i = i + 1;
    end
    all_runs = all_runs + i;
end
average_iterations = all_runs / num_runs;
disp(average_iterations);
```

```matlab
sumTotal2 = 0;
for i=1:numIterations
    rnc2 = 0;
    sum2 = 0;
    while sum2<=10
        x = rand;
        sum2 = sum2 + x;
        rnc2 = rnc2 + 1;
    end
    sumTotal2 = sumTotal2 + rnc2;
end
average2 = sumTotal2/numIterations;
disp(average2);
```

The one on the left is noticeably slower than the other... but if we changed “sum” to “sum2”, it wouldn’t be!

It’s best not to use Matlab keywords / functions as variable names

Constrained Growth

- Is the unconstrained growth model a good one for natural organisms in natural environments?
  - Probably not. Limited resources, etc., can constrain growth
- Vocab: The **carrying capacity** \( M \) for an organism in an environment is the maximum population size of that organism supported by that environment.
  - E.g., if a deer refuge can support at most 1000 deer, we say \( M \) for the deer in the refuge is 1000.

**Notation question for programmers and computer scientists:** In the example, \( M \) is used to represent carrying capacity. What do you think of that notation?

- How to revise our unconstrained growth model to incorporate \( M \)?

\[
\frac{dP}{dt} = rP \\
P = P_0 e^{rt}
\]
Constrained Growth

- How to revise our unconstrained growth model to incorporate M?
- Desired behavior of the revised model
  - When population far less than M, growth should be similar to unconstrained model
  - When population near M, growth should be nearly 0 (births nearly equal deaths)
- So, how could we incorporate M in the model?
  - What approaches might we consider?
  - What variables might we add to the model?
  - One suggestion: Let’s say we have the number of births represented as \( rP \) for population size \( P \) and growth constant \( r \)...
  - …we could have a number of deaths also represented. What might be the equation specifying the rate of change of number of deaths?

\[
\frac{dP}{dt} = rP
\]

\[ P = P_0 e^{rt} \]
Constrained Growth

- How to revise our unconstrained growth model to incorporate \( M \)?
- Desired behavior of the revised model
  - When population far less than \( M \), growth should be similar to unconstrained model
  - When population near \( M \), growth should be nearly 0 (births nearly equal deaths)
- So, how could we incorporate \( M \) in the model?
- The number of births is represented as \( rP \). For the number of deaths (call it \( D \)), the rate of change should fit certain constraints
  - When population size \( P \) is much less than \( M \), the death rate should be very low
    - Asymptotically, we might think that when \( P \) nears 0, the death rate should near 0
  - When \( P \) is nearly \( M \), the death rate should be nearly equal to the birth rate
    - Asymptotically, we might think that when \( P \) gets to \( M \), the death rate gets equal to the birth rate
- What is a candidate model (equation) that has this behavior?

\[
\frac{dP}{dt} = rP
\]

\[
P = P_0 e^{rt}
\]
Constrained Growth and Logistic Functions

- Our new model equation for population change:
  \[ \frac{dP}{dt} = rP - \frac{P}{M} \]

- Solving this equation for \( P \), under initial conditions \( P(0) = P_0 \), gives the function
  \[ P(t) = \frac{MP_0}{P_0 + (M - P_0)e^{-rt}} \]

  This is called a logistic function, and if the population starts out less than carrying capacity \( M \), it has a characteristic “S” shaped graph, e.g.

For this graph, \( M = 1000, P_0 = 20, \) and \( r = 0.5 \)

Constrained Growth and Logistic Functions

- Our new model equation for population change:
  \[ \frac{dP}{dt} = rP - \frac{P}{M} \]

- Solving this equation for \( P \), under initial conditions \( P(0) = P_0 \), gives the function
  \[ P(t) = \frac{MP_0}{P_0 + (M - P_0)e^{-rt}} \]

  This is called a logistic function, and if the population starts out less than carrying capacity \( M \), it has a characteristic “S” shaped graph

   And if the population starts out greater than \( M \), it decreases to approach \( M \), as in this graph

For this graph, \( M = 1000, P_0 = 1500, \) and \( r = 0.5 \)
Some Matlab code; The *axis* statement

```matlab
% Some Matlab code; The *axis* statement

t = 0:0.001:15; % times
M = 1000; % carrying capacity
P0 = 20; % init pop'n
irog = 0.5;

% compute and plot the population
figure
q = M*P0 ./ (P0 + (M-P0)*exp(-irog*t)); % q is population curve
plot(t,q)

% change parameters for initial population above carrying capacity
% compute population
P0 = 1500;
q2 = M*P0 ./ (P0 + (M-P0)*exp(-irog*t)); % q2 is population curve

% plot population in case of initial population above carrying capacity
% figure % so as not to overwrite previous figure
% hold % to label axes properly and plot data on same figure
% but only needed if axis command comes before plot
% if axis is after plot, it's okay
% axis([0 15 0 1600]) % what happens if we don't have this line here?
plot(t,q2)
axis([0 15 0 1600]) % what happens if we don't have this line here?
```

---

Discrete Logistic Equations

- For discretized simulations, we use discretized versions of the rate of change equations, but we get similar logistic-y behavior
  - Recall, we had\[ \frac{dD}{dt} = \frac{P}{M} (rP) \]
  - Discretized, the number of deaths over a time interval from \((t-\Delta t)\) to \(t\) can be written as\[ \Delta D = \left( r \frac{P(t-\Delta t)}{M} \right) P(t-\Delta t) \Delta t \]
  - Where \(\Delta D\) is the number of deaths, \(P(t-\Delta t)\) is the population (perhaps estimated) at time \(t-\Delta t\), and \(\Delta t\) is the length of the time interval

You might think of \(\Delta t\) as being in place of \(dt\) in the continuous case

- So, the population change \(\Delta P = \) (births – deaths) over that interval is

\[ \Delta P = \left( rP(t-\Delta t) \right) \Delta t - \left( r \frac{P(t-\Delta t)}{M} \right) P(t-\Delta t) \Delta t \]
Discrete Logistic Equations

- For discretized simulations, we use discretized versions of the rate of change equations, but we get similar logistic-y behavior
  - Recall, we had \( \frac{dD}{dt} = \frac{P}{M}(tP) \)
  - Discretized, the number of deaths over a time interval from \((t-\Delta t)\) to \(t\) can be written as
    \[
    \Delta D = \left( \frac{r}{M} \right) P(t-\Delta t) \Delta t
    \]
    Where \(\Delta D\) is the number of deaths, \(P(t-\Delta t)\) is the population (perhaps estimated) at time \(t - \Delta t\), and \(\Delta t\) is the length of the time interval

- So, the population change \(\Delta P = (\text{births} - \text{deaths})\) over that interval is
  \[
  \Delta P = \left( rP(t-\Delta t) \right) \Delta t - \left( \frac{P(t-\Delta t)}{M} \right) P(t-\Delta t) \Delta t
  \]
  or
  \[
  \Delta P = \left( r\Delta t \right) \left( 1 - \frac{P(t-\Delta t)}{M} \right) P(t-\Delta t)
  \]

And for births...
Do you see how to discretize this, too?

You might think of \(\Delta t\) as being in place of \(dt\) in the continuous case

- For births...