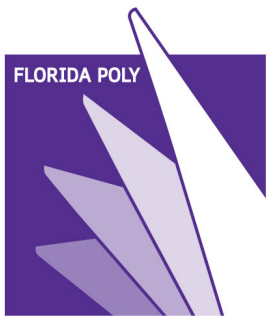
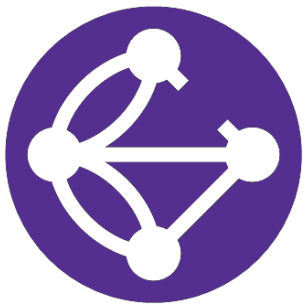
An Accessible Approach to Mathematical Modeling without Differential Equations Prerequisites

Adam Rumpf

arumpf@floridapoly.edu



Florida Polytechnic University
Department of Applied Mathematics



SIMIODE EXPO 2026

February 14, 2026

Introduction

Who I Am

- Adam Rumpf, Ph.D.
- Instructor in Applied Mathematics Department at Florida Polytechnic University since 2021

Who I Am

- Mathematical modeling is a key part of my background
  - It was one of the things that initially got me interested in mathematics
  - My undergraduate and graduate Applied Mathematics degrees both focused on it
- I try to apply a **modeling-first** approach to the courses I teach to bring context and motivation to the material

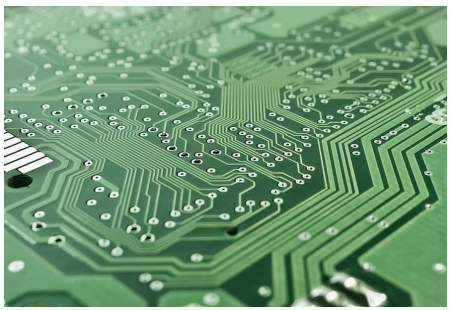
Differential Equations

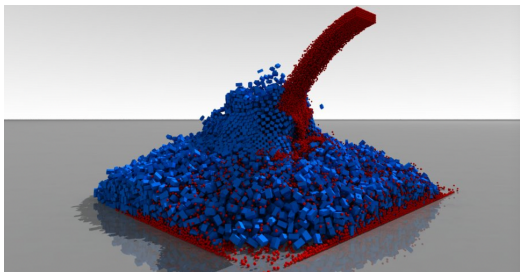
- **Ordinary differential equations (ODEs)** form an essential part of any mathematical modeling curriculum
- They provide the basic language for describing things that *change* over time
- They capture the spirit of what makes mathematical modeling fun and interesting

$$\frac{dx}{dt} = rx\left(1 - \frac{x}{K}\right) - \frac{x^2}{1+x^2}$$

Differential Equations

- They arise in every STEM field, with examples including:
  - mechanical systems
  - electronics
  - population ecology
  - infectious disease spread
  - physics simulations and game design





Differential Equations

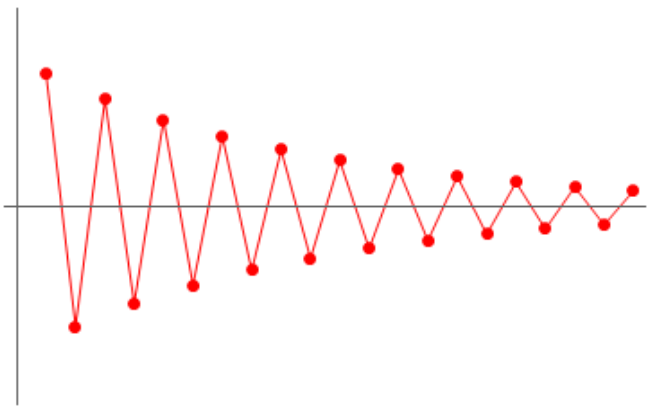
- However, they are also a specialized topic
- Only certain STEM majors typically require a Differential Equations course
- This can make it challenging to teach ODE-based mathematical modeling in an accessible way

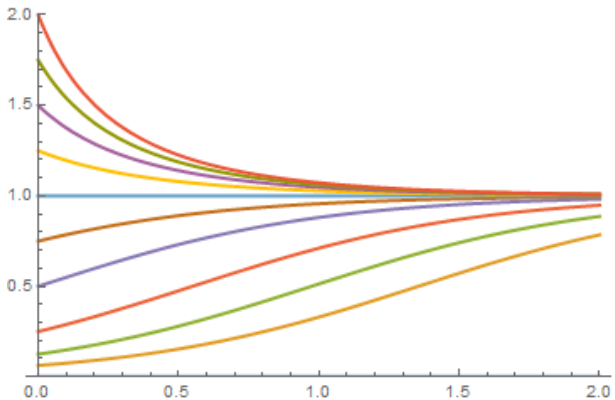
This Talk

- I want to tell you about Florida Poly's first dedicated Mathematical Modeling course
  - Upper-level Applied Mathematics elective
  - Planned in Summer 2025 with Dr. Somak Das
  - Piloted last semester (Fall 2025)
- Some of our design challenges and solutions might be interesting and informative

This Talk

- I will focus only on first half of course, which covered difference equations and differential equations
- The second half went on to cover other areas of mathematics (regression, graph theory, optimization, etc.)





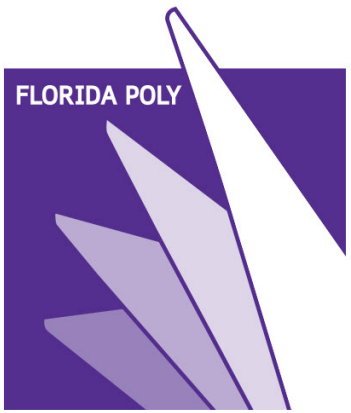
This Talk

- I will discuss:
  - what we did
  - why we did it
  - what we learned
- These lessons will hopefully be useful to anyone planning a Mathematical Modeling course
  - Or a course with a significant modeling component
  - Or for approaching a Differential Equations course in a different way

Background

New Courses

- Florida Poly is a young and fast-growing university
  - Opened in 2014 with 554 students
  - 1,930 students as of Fall 2025
- We have developed and added many new courses in the last few years

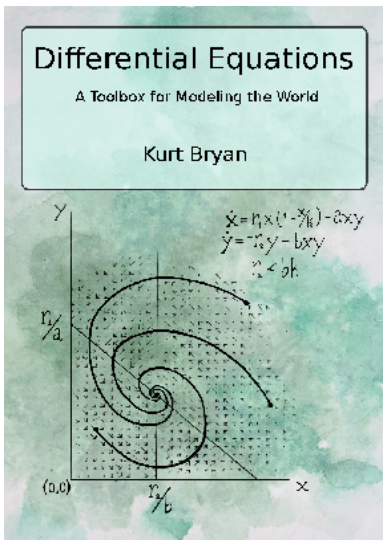


Student Body

- Florida Poly is an all-STEM school
- Our Applied Mathematics program is a very small fraction of the student body
- Upper-division Applied Mathematics electives require interest from other majors
  - Computer Science is our largest major

Differential Equations

- Most majors require our Differential Equations course
- It follows the SIMIODE textbook (*Differential Equations: A Toolbox for Modeling the World*, by Kurt Bryan)
- Its coverage focuses primarily on:
  - single ODEs (not systems)
  - exact solution methods (with some aid from MATLAB)



Prerequisites

- Problem: The Computer Science major does not require Differential Equations
- Making Differential Equations a prerequisite for Mathematical Modeling would immediately exclude most of our upperclassmen

Prerequisites

- Even if we *did* include Differential Equations as a prerequisite, we would want for Mathematical Modeling to still be worthwhile
- That would mean avoiding too much repeated material and focusing more on novel content

Design Goals

Design an upper-level Mathematical Modeling course that:

a. covers ODEs in significant detail as an essential modeling tool
b. does not rely on Differential Equations as a prerequisite
c. offers new content and insights for students that *have* taken Differential Equations

Course Design

(Not) Solving ODEs

- Most of these goals can be met with a simple design philosophy
- ***You do not need to solve a dynamical system to learn everything you need to know about it***
- This opens many new options for how to cover ODEs

(Not) Solving ODEs

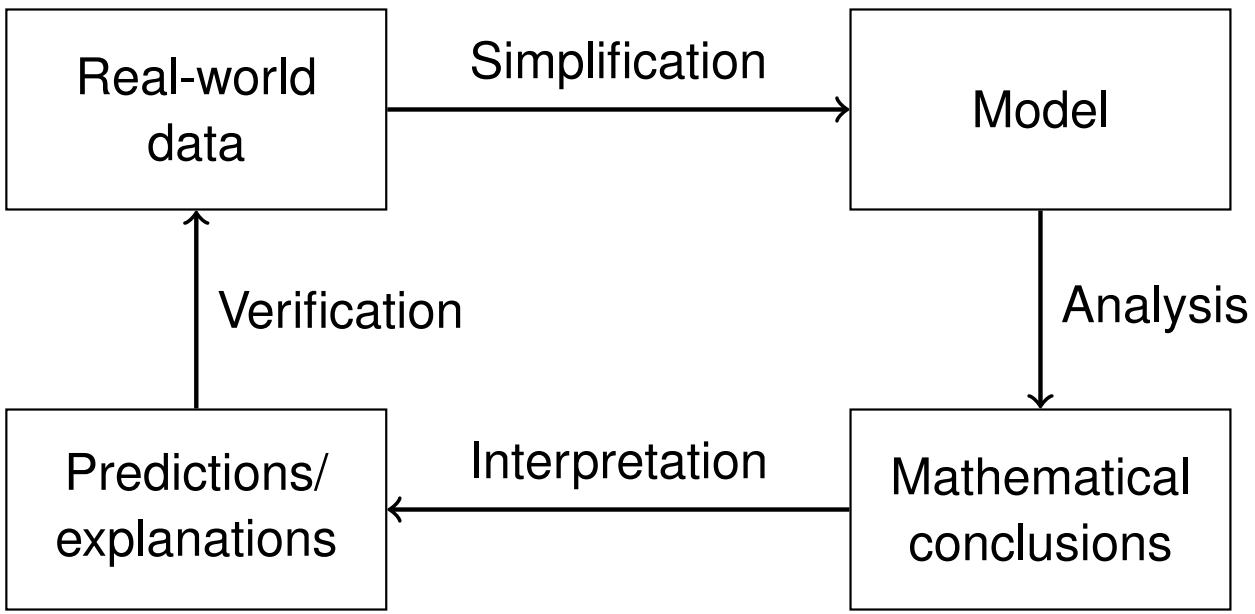
- Rather than focusing on solving ODEs (as in a Differential Equations course), focus on analysis methods that do *not* require solving them
- Major points of focus include:
  - The model development process
  - Qualitative analysis
  - Numerical solutions

(Not) Solving ODEs

- This improves accessibility while showing plenty of new content to students with ODE experience
- In practice, these tools are often needed for systems that cannot be solved analytically

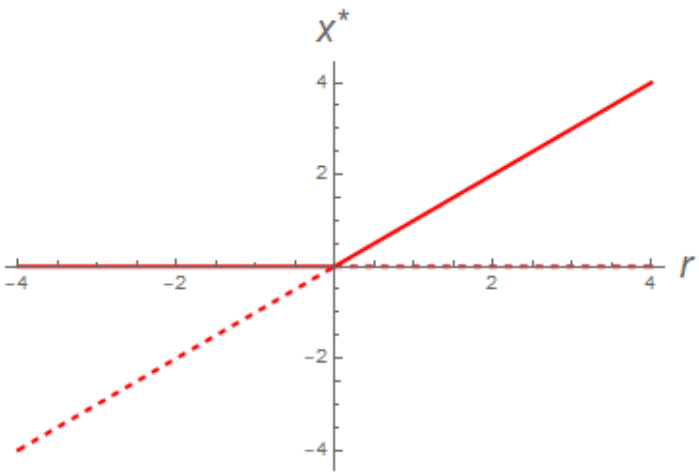
Model Development

- Cyclical process of developing a model
- Focus on precisely identifying simplifying assumptions, defining variables and parameters, and identifying dimensions and units
- Validating model using real-world data



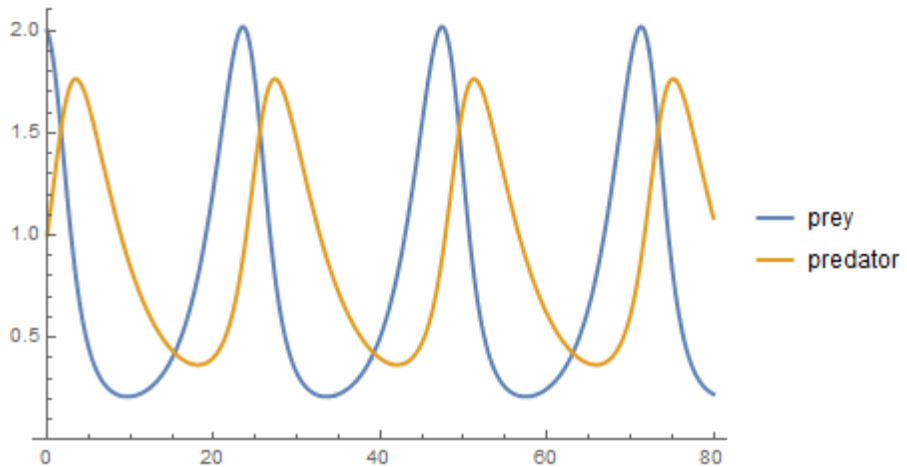
Qualitative Analysis

- Graphical techniques (e.g. phase lines/planes)
- Equilibrium analysis (e.g. linear stability analysis)
- Sensitivity analysis (e.g. bifurcation analysis)



Numerical Solutions

- Numerical simulations are a valuable outcome from developing a model
- They can quickly show expected behavior without requiring deep mathematical background



Linear Algebra

- Make **Linear Algebra** the only prerequisite
- Nearly all majors (including Computer Science) require Linear Algebra
- It also guarantees:
  - some exposure to MATLAB
  - some exposure to ODEs (through its Calculus II prerequisite)

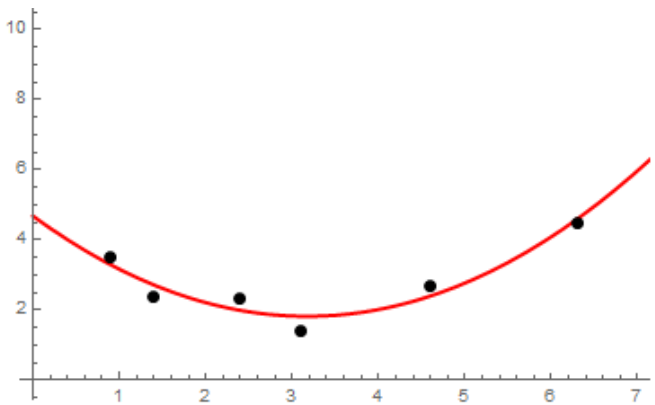
Linear Algebra

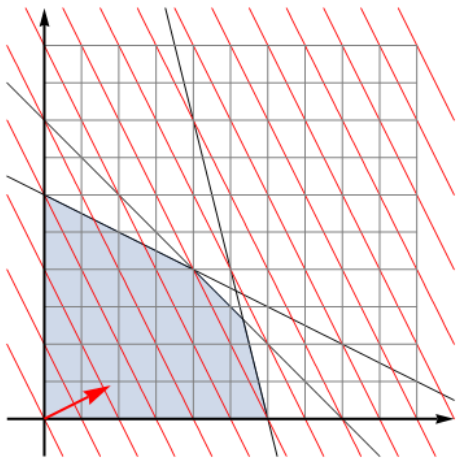
- Within ODE coverage, Linear Algebra unlocks:
  - dimensional analysis
  - working with linear systems of difference/differential equations
  - linear stability analysis

$$\begin{bmatrix} x'(t) \\ y'(t) \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x(t) \\ y(t) \end{bmatrix}$$

Linear Algebra

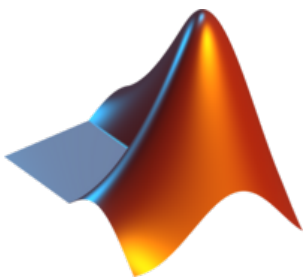
- We could also use it for later Mathematical Modeling topics:
  - regression
  - linear optimization





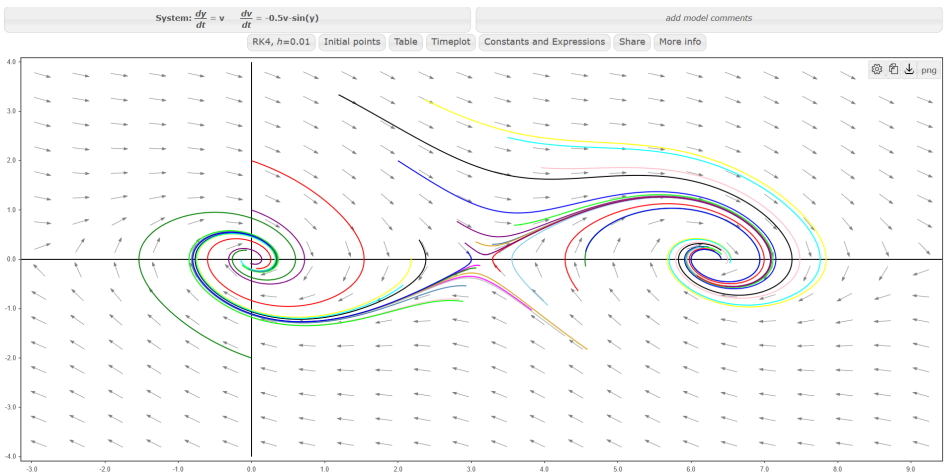
Technology

- We regularly used interactive Desmos demos for activities and to illustrate discrete dynamical systems
- Regular MATLAB lessons, activities, homework questions, and take-home exam questions



Technology

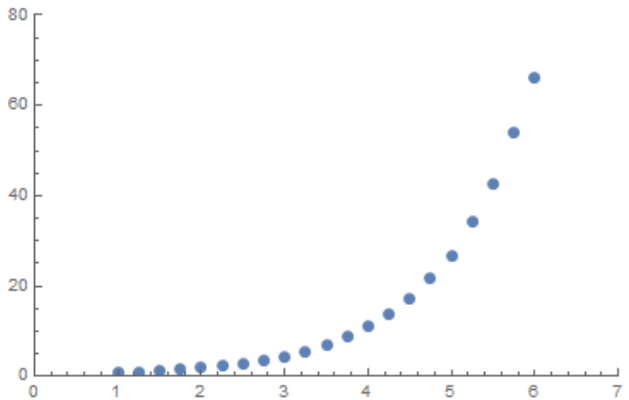
- We regularly used a browser-based slope field generator for studying ODEs
- Slope and Direction Fields: https://homepages.bluffton.edu/~nesterd/apps/slopefields.html
- Similar to *Slopes: Differential Equations* app



Lessons Learned

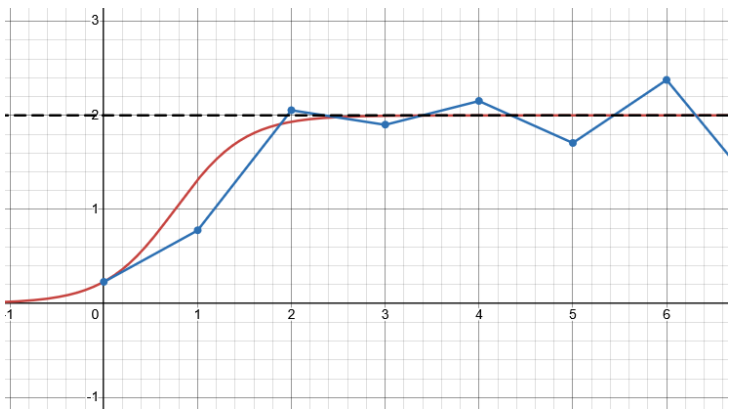
Start with a Mess

- Don’t start with such simple, familiar modeling examples (like exponential population growth)
- Some students see these, assume they know everything, and stop paying attention



Start with a Mess

- We began with *difference* rather than *differential* equations
- Many students wanted to jump immediately to treating difference equations like ODEs
- We had to take time to highlight the fact that they are different types of model with different assumptions and behaviors
  - e.g. Discrete versus Continuous Logistic Growth: https://www.desmos.com/calculator/xjdrcnud8u



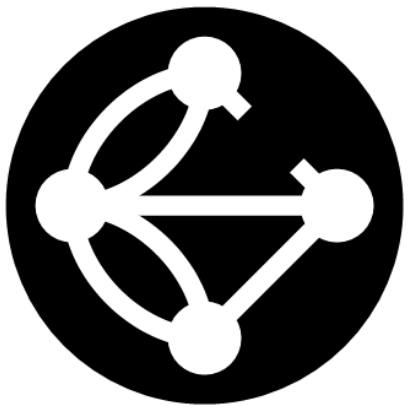
Time Allocation

- Cover less material and try to avoid the traditional "content-driven" structure that most mathematics classes follow
- Dedicate class days to group modeling projects
- Grant more exposure to developing and explaining a model for a complex problem

Conclusions

- ***You do not need to solve a dynamical system to learn everything you need to know about it***
- It is very possible to cover differential equations models with minimal prerequisites
  - Focus on modeling, qualitative analysis, and applying numerical solutions
  - Linear algebra helps
- Keep this in mind for courses that involve significant mathematical modeling (including Differential Equations, itself)

Thank you!



Course Repo: https://github.com/adam-rumpf/map4103-fa25