ODEN INSTITUTE
FOR COMPUTATIONAL ENGINEERING & SCIENCES

FALL 2025

# MEETING #3

Computational Modeling in Engineering and the Sciences

**ASHTON COLE**
Graduate Student, Computational Hydraulics Group, The University of Texas at Austin

# Agenda

- Discussion of Baigent - Lotka-Volterra Dynamics and Runge-Kutta
- Introduction to multidimensional ODEs
- Demo: SEIR and Lotka-Volterra

**Assignment: (Lightly) read Jah - Multiple Object Space Surveillance. Review Newton’s and Kepler’s laws. We’ll discuss next week.**

ODEN INSTITUTE
FOR COMPUTATIONAL ENGINEERING & SCIENCES

ODEN INSTITUTE
FOR COMPUTATIONAL ENGINEERING & SCIENCES

# Discussion of Baigent - Lotka-Volterra Dynamics

ODEN INSTITUTE
FOR COMPUTATIONAL ENGINEERING & SCIENCES

# Introduction to Multidimensional ODEs

# Introduction to Multidimensional ODEs

- **ordinary differential equation:** a differential equation with only one **independent** variable
- Introducing the IVP:

$$\frac{\mathrm{d}u}{\mathrm{d}t} = f(u,t)$$
$$u(t_0) = u_0$$
$$u(t) = ?$$

# Introduction to Multidimensional ODEs

- **system of ordinary differential equations:** a set of differential equations with only one **independent** variable
- Introducing the IVP:

$$
\begin{aligned}
\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} &= f(\mathbf{u}, t) \\
\mathbf{u}(t_0) &= \mathbf{u}_0 \\
\mathbf{u}(t) &= ?
\end{aligned}
$$

# Introduction to Multidimensional ODEs

- **linear system of ordinary differential equations:** writable as a matrix equation
- Exact solution
- Linearization in a small region to understand behavior

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = \mathbf{A}\mathbf{u}$$

$$\mathbf{u}(t_0) = \mathbf{u}_0$$

$$\mathbf{u}(t) = \,?$$

# Introduction to Multidimensional ODEs

- **stiff ordinary differential equations:** difficult to solve without a tiny step size

$$
\begin{aligned}
\mathrm{A} &\xrightarrow{0.04} \mathrm{B} \\
\mathrm{B}+\mathrm{B} &\xrightarrow{3\cdot 10^7} \mathrm{C}+\mathrm{B} \\
\mathrm{B}+\mathrm{C} &\xrightarrow{1\cdot 10^4} \mathrm{A}+\mathrm{C}
\end{aligned}
$$

$$
\begin{aligned}
\dot{x} &= -0.04x + 10^4 y \cdot z \\
\dot{y} &= 0.04x - 10^4 y \cdot z - 3 \cdot 10^7 y^2 \\
\dot{z} &= 3 \cdot 10^7 y^2
\end{aligned}
$$

## Demo: SEIR and Lotka-Volterra

- Get in groups. Complete the my_integrators.py module, and use it to solve the SEIR and Lotka-Volterra systems of ODE’s.
- Bonus tasks:
  - Compare against the exact solution for Lotka-Volterra, which is defined implicitly.
  - Generate phase plots.
  - Try using implicit or high-order solvers, e.g. from scipy.