EOSC 213: Computational Methods in Geological Engineering

Lecture 11: Interpreting systems of ODEs via eigenvalues
+ Forward Euler/ Explicit Midpoint / Backward Euler for systems

Shadab Ahamed

Department of Earth, Ocean and Atmospheric Sciences,
The University of British Columbia

Thu, Feb 10, 2026

Course Logistics and Announcements

Homework 02

- HW02 is posted on Canvas; submission deadline: **Friday, 20th Feb, 2026**
  - No extensions as I want to release HW02 solutions before Midterm I
- This homework is good practice for Midterm I which scheduled on **24th Feb, 2026**
  - Instructions for Midterm I will be posted on Canvas soon
- Please follow submission instructions carefully:
  - Written solutions: **single PDF**
  - Python notebook: submit `.ipynb` **and** `.html` export. These should match EXACTLY
  - -2 if `.html` file is missing
  - Make sure your submitted files are correct and complete

Recap from Lecture 10

Last lecture, we:

- wrote systems of ODEs using vectors: $\mathbf{x}' = \mathbf{F}(\mathbf{x}, t)$
- focused on **linear systems**: $\mathbf{x}' = A\mathbf{x}$
- used **phase portraits** ($y(t)$ vs. $x(t)$ plots for different initial conditions) to observe qualitative behavior of the system
- learned how to compute eigenvalues from $\det(A - \lambda I) = 0$

Today: how eigenvalues predict behavior, and how numerical methods extend to systems.

Lecture 11: What we will learn today

1. **Part I:** Equilibrium points
   - what are equilibrium points?
   - stable, unstable equilibrium and saddle points

2. **Part II:** Interpreting dynamics of the system from eigenvalues of A
   - real positive/negative eigenvalues
   - complex eigenvalues (spirals)
   - purely imaginary eigenvalues (centers)

3. **Part III:** Forward Euler / Explicit Midpoint / Backward Euler for systems
   - vector form of the updates
   - for $\mathbf{x}' = A\mathbf{x}$: amplification matrices
   - stability depends on eigenvalues again

Learning goals

By the end of today, you should be able to:

- predict decay/growth/oscillation of $\mathbf{x}' = A\mathbf{x}$ from eigenvalues of A
- recognize the main eigenvalue cases (real, complex, imaginary)
- write Forward Euler / Midpoint / Backward Euler for a **system**
- for $\mathbf{x}' = A\mathbf{x}$, identify the **amplification matrix** of each method
- connect numerical stability to eigenvalues (same idea as scalar case)

Practice problem: diagonalizable 3D system

Consider the linear system of ODEs

$$\frac{d\mathbf{x}}{dt} = A\mathbf{x}, \qquad A = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 3 & 1 \\ 0 & 0 & 4 \end{bmatrix}.$$

Tasks:

1. Compute the eigenvalues of A.
2. Find one eigenvector corresponding to each eigenvalue.
3. Write down the general solution of the system.

Hint: For an upper-triangular matrix, the eigenvalues are the diagonal entries.

Solution: Eigenvalues

We are given

$$A = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 3 & 1 \\ 0 & 0 & 4 \end{bmatrix}, \qquad \frac{d\mathbf{x}}{dt} = A\mathbf{x}.$$

Since A is upper triangular, the eigenvalues are the diagonal entries:

$$\lambda_1 = 2, \quad \lambda_2 = 3, \quad \lambda_3 = 4.$$

All eigenvalues are distinct.

Solution: Eigenvalues via characteristic equation

The eigenvalues satisfy the characteristic equation

$$\det(A - \lambda I) = 0.$$

Solution: Eigenvalues via characteristic equation

The eigenvalues satisfy the characteristic equation

$$\det(A - \lambda I) = 0.$$

For

$$A - \lambda I = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 3 & 1 \\ 0 & 0 & 4 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 2 - \lambda & 1 & 0 \\ 0 & 3 - \lambda & 1 \\ 0 & 0 & 4 - \lambda \end{bmatrix},$$

Solution: Eigenvalues via characteristic equation

The eigenvalues satisfy the characteristic equation

$$\det(A - \lambda I) = 0.$$

For

$$A - \lambda I = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 3 & 1 \\ 0 & 0 & 4 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 2-\lambda & 1 & 0 \\ 0 & 3-\lambda & 1 \\ 0 & 0 & 4-\lambda \end{bmatrix},$$

the determinant is

$$\det(A - \lambda I) = (2-\lambda)(3-\lambda)(4-\lambda).$$

Solution: Eigenvalues via characteristic equation

The eigenvalues satisfy the characteristic equation

$$\det(A - \lambda I) = 0.$$

For

$$A - \lambda I = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 3 & 1 \\ 0 & 0 & 4 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 2 - \lambda & 1 & 0 \\ 0 & 3 - \lambda & 1 \\ 0 & 0 & 4 - \lambda \end{bmatrix},$$

the determinant is

$$\det(A - \lambda I) = (2 - \lambda)(3 - \lambda)(4 - \lambda).$$

Setting this equal to zero gives

$$(2 - \lambda)(3 - \lambda)(4 - \lambda) = 0.$$

Eigenvalues

$$\lambda = 2, \quad 3, \quad 4.$$

Solution: Eigenvector for eigenvalue $\lambda_1 = 2$

Computing the eigenvector for the eigenvalue $\lambda_1 = 2$, let us solve,

$$(A - \lambda_1 I)\mathbf{v}_1 = \mathbf{0} \quad \Rightarrow \quad (A - 2I)\mathbf{v}_1 = \mathbf{0}.$$

We compute

$$A - 2I = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 3 & 1 \\ 0 & 0 & 4 \end{bmatrix} - \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 2 \end{bmatrix}.$$

Solution: Eigenvector for eigenvalue $\lambda_1 = 2$

Computing the eigenvector for the eigenvalue $\lambda_1 = 2$, let us solve,

$$(A - \lambda_1 I)\mathbf{v}_1 = \mathbf{0} \quad \Rightarrow \quad (A - 2I)\mathbf{v}_1 = \mathbf{0}.$$

We compute

$$A - 2I = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 3 & 1 \\ 0 & 0 & 4 \end{bmatrix} - \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 2 \end{bmatrix}.$$

So we solve

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

This gives the system $x_2 = 0, \quad x_3 = 0$.

So one eigenvector is

$$\mathbf{v}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}.$$

Solution: Eigenvector for eigenvalue $\lambda_2 = 3$

Computing the eigenvector for the eigenvalue $\lambda_2 = 3$, let us solve,

$$(A - \lambda_2 I)\mathbf{v}_2 = \mathbf{0} \quad \Rightarrow \quad (A - 3I)\mathbf{v}_2 = \mathbf{0}.$$

We compute

$$A - 3I = \begin{bmatrix} -1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}.$$

Solution: Eigenvector for eigenvalue $\lambda_2 = 3$

Computing the eigenvector for the eigenvalue $\lambda_2 = 3$, let us solve,

$$(A - \lambda_2 I)\mathbf{v}_2 = \mathbf{0} \quad \Rightarrow \quad (A - 3I)\mathbf{v}_2 = \mathbf{0}.$$

We compute

$$A - 3I = \begin{bmatrix} -1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}.$$

So we solve

$$\begin{bmatrix} -1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

This gives the system $x_2 = x_1, \quad x_3 = 0$.

So one eigenvector is

$$\mathbf{v}_2 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}.$$

Solution: Eigenvector for eigenvalue $\lambda_3 = 4$

Computing the eigenvector for the eigenvalue $\lambda_3 = 4$, let us solve,

$$(A - \lambda_3 I)\mathbf{v}_3 = \mathbf{0} \quad \Rightarrow \quad (A - 4I)\mathbf{v}_3 = \mathbf{0}.$$

We compute

$$A - 4I = \begin{bmatrix} -2 & 1 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}.$$

Solution: Eigenvector for eigenvalue $\lambda_3 = 4$

Computing the eigenvector for the eigenvalue $\lambda_3 = 4$, let us solve,

$$(A - \lambda_3 I)\mathbf{v}_3 = \mathbf{0} \quad \Rightarrow \quad (A - 4I)\mathbf{v}_3 = \mathbf{0}.$$

We compute

$$A - 4I = \begin{bmatrix} -2 & 1 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}.$$

So we solve

$$\begin{bmatrix} -2 & 1 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

This gives the system $x_2 = 2x_1, \quad x_3 = 2x_1$.

So one eigenvector is

$$\mathbf{v}_3 = \begin{bmatrix} 1 \\ 2 \\ 2 \end{bmatrix}.$$

Solution: General solution of the ODE

For a diagonalizable matrix, the solution is a linear combination of eigenmodes:

$$\mathbf{x}(t) = c_1 e^{2t} \mathbf{v}_1 + c_2 e^{3t} \mathbf{v}_2 + c_3 e^{4t} \mathbf{v}_3.$$

Explicitly,

$$\mathbf{x}(t) = c_1 e^{2t} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + c_2 e^{3t} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} + c_3 e^{4t} \begin{bmatrix} 1 \\ 2 \\ 2 \end{bmatrix}.$$

Key takeaway

Each eigenvalue contributes an independent exponential mode. The values of c_1, c_2 and c_3 are obtained from the initial conditions.

Equilibrium points for a dynamical system

We often write a system of ODEs as

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, t),$$

- $\mathbf{x}(t)$ is the *state* (a vector of unknowns)
- $\mathbf{f}(\mathbf{x})$ is the *vector field* (it tells us the direction of change)

Equilibrium point (steady state)

Definition. A vector $\mathbf{x}_e$ is an equilibrium point if,

$$\mathbf{f}(\mathbf{x}_e) = \mathbf{0}.$$

Equilibrium point (steady state)

Definition. A vector $\mathbf{x}_e$ is an equilibrium point if,

$$\mathbf{f}(\mathbf{x}_e) = \mathbf{0}.$$

Interpretation:

- If $\mathbf{x}(0) = \mathbf{x}_e$, then $\dfrac{d\mathbf{x}}{dt} = \mathbf{0}$ and the solution stays constant:

$$\mathbf{x}(t) \equiv \mathbf{x}_e.$$

- Equilibria are the "no net change" states of the system.

How to judge stability in 1D?

Consider a scalar ODE

$$\frac{dx}{dt} = f(x).$$

An equilibrium x_e satisfies $f(x_e) = 0$.

To determine stability, ask:

- What is the sign of $f(x)$ *just left* of x_e?
- What is the sign of $f(x)$ *just right* of x_e?

This tells us whether trajectories move toward or away from x_e.

Stable vs. unstable equilibrium (intuition)

Stable equilibrium: small perturbation $\Rightarrow$ trajectories move back toward $\mathbf{x}_e$.

Unstable equilibrium: small perturbation $\Rightarrow$ trajectories move away from $\mathbf{x}_e$.

Stable vs. unstable equilibrium (intuition)

Stable equilibrium: small perturbation $\Rightarrow$ trajectories move back toward $\mathbf{x}_e$.

Unstable equilibrium: small perturbation $\Rightarrow$ trajectories move away from $\mathbf{x}_e$.

Mental picture

- Bottom of a bowl: stable
- Top of a hill: unstable

Example 1: Stable equilibrium in 1D

Consider

$$\frac{dx}{dt} = -x.$$

Equilibrium:

$$-x = 0 \quad \Rightarrow \quad x_e = 0.$$

Sign analysis:

- If $x > 0$, then $\dot{x} < 0$ (moves left)
- If $x < 0$, then $\dot{x} > 0$ (moves right)

Conclusion

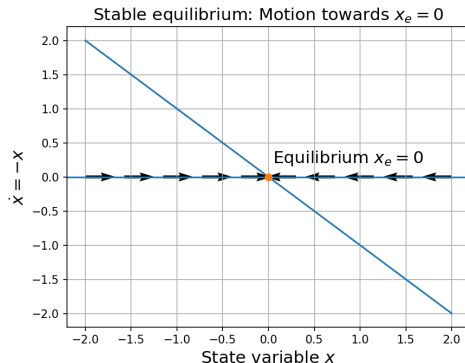All nearby trajectories move toward $x = 0$.
$\Rightarrow$ **Stable equilibrium**

Figure: Stable equilibrium at $x_e = 0$.
Perturbations around $x_e = 0$ will bring the
state back to $x_e = 0$.

Example 2: Unstable equilibrium in 1D

Consider

$$\frac{dx}{dt} = x.$$

Equilibrium:

$$x = 0 \quad \Rightarrow \quad x_e = 0.$$

Sign analysis:

- If $x > 0$, then $\dot{x} > 0$ (moves right)
- If $x < 0$, then $\dot{x} < 0$ (moves left)

Conclusion

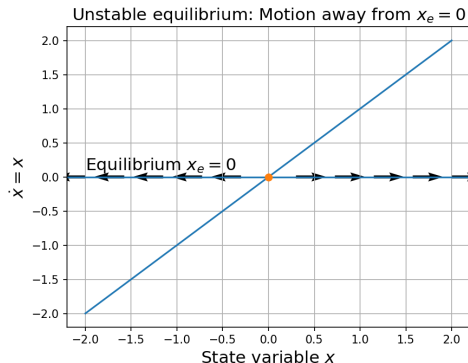All nearby trajectories move away from $x = 0$. $\Rightarrow$ **Unstable equilibrium**

Figure: Unstable equilibrium at $x_e = 0$. Perturbations around $x_e = 0$ will move the state away from $x_e = 0$.

Example 3: Semi-stable equilibrium in 1D

Consider

$$\frac{dx}{dt} = x^2.$$

Equilibrium:

$$x^2 = 0 \quad \Rightarrow \quad x_e = 0.$$

Sign analysis:

- If $x > 0$, then $\dot{x} > 0$ (moves right)
- If $x < 0$, then $\dot{x} > 0$ (moves right)

Conclusion

Trajectories approach $x = 0$ from the left, but move away on the right.
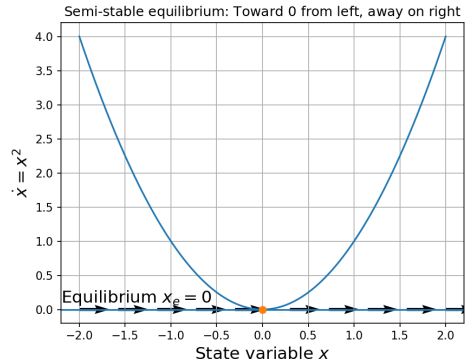$\Rightarrow$ **Semi-stable equilibrium**

Figure: Semi-stable equilibrium at $x_e = 0$

Takeaway from 1D examples

- Stability in 1D can be understood purely from the sign of $f(x)$
- Small perturbations tell the full story
- This intuition generalizes to higher dimensions using eigenvalues

  In higher dimensions, different directions behave like different 1D problems.

Why equilibria matter

Equilibria are the reference points for understanding dynamics:

- Many systems spend a lot of time *near* equilibria
- Long-time behavior is often about approaching or moving away from equilibria
- Stability questions are naturally posed around equilibria

Key question

If we start near $\mathbf{x}_e$, do we return towards it or move away from it?

Example: find equilibria and their stability

Consider the scalar ODE,

$$\frac{dx}{dt} = x^2 - 1.$$

Equilibria satisfy $\frac{dx}{dt} = 0$, i.e.

$$x^2 - 1 = 0 \quad \Rightarrow \quad x_e = -1,\ 1.$$

Conclusion

The sign of the local dynamics makes $x_e = -1$ stable and $x_e = +1$ unstable.
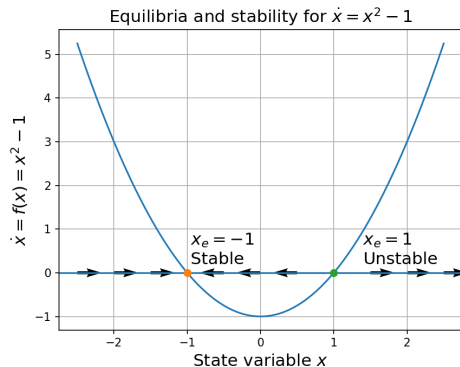

Figure: Two equilibrium points, $x_e = -1$ (stable) and $x_e = 1$ (unstable).

Example 1: Stable equilibrium in 2D

Consider the linear system,

$$\frac{dx}{dt} = -x, \quad \frac{dy}{dt} = -2y.$$

Example 1: Stable equilibrium in 2D

Consider the linear system,

$$\frac{dx}{dt} = -x, \quad \frac{dy}{dt} = -2y.$$

Equilibrium:

$$x = 0, \quad y = 0 \quad \Rightarrow \quad (x_e, y_e) = (0, 0).$$

Example 1: Stable equilibrium in 2D

Consider the linear system,

$$\frac{dx}{dt} = -x, \quad \frac{dy}{dt} = -2y.$$

Equilibrium:

$$x = 0, \quad y = 0 \quad \Rightarrow \quad (x_e, y_e) = (0, 0).$$

Behavior along each direction:

- x-direction: $\dot{x} = -x \Rightarrow$ decay toward 0
- y-direction: $\dot{y} = -2y \Rightarrow$ faster decay toward 0

Example 1: Stable equilibrium in 2D

Consider the linear system,

$$\frac{dx}{dt} = -x, \quad \frac{dy}{dt} = -2y.$$

Equilibrium:

$$x = 0, \quad y = 0 \quad \Rightarrow \quad (x_e, y_e) = (0, 0).$$

Behavior along each direction:

- x-direction: $\dot{x} = -x \Rightarrow$ decay toward 0
- y-direction: $\dot{y} = -2y \Rightarrow$ faster decay toward 0

Conclusion

All nearby trajectories decay toward the equilibrium.
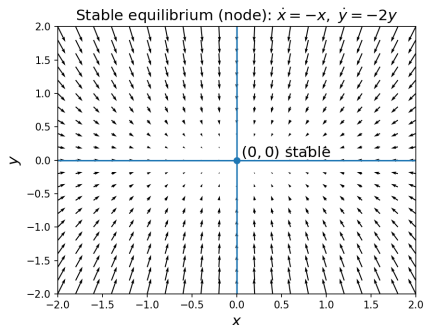$\Rightarrow$ **Stable equilibrium (stable node)**

Figure: All solutions decay to $(x_e, y_e) = (0, 0)$ making this a stable equilibrium point.

Example 2: Unstable equilibrium in 2D

Consider the linear system,

$$\frac{dx}{dt} = x, \quad \frac{dy}{dt} = 2y.$$

Example 2: Unstable equilibrium in 2D

Consider the linear system,

$$\frac{dx}{dt} = x, \quad \frac{dy}{dt} = 2y.$$

Equilibrium:
$$x = 0, \quad y = 0 \quad \Rightarrow \quad (x_e, y_e) = (0, 0).$$

Example 2: Unstable equilibrium in 2D

Consider the linear system,

$$\frac{dx}{dt} = x, \quad \frac{dy}{dt} = 2y.$$

Equilibrium:

$x = 0, \quad y = 0 \quad \Rightarrow \quad (x_e, y_e) = (0, 0).$

Behavior along each direction:

- x-direction: $\dot{x} = x \Rightarrow$ growth away from 0
- y-direction: $\dot{y} = 2y \Rightarrow$ growth away from 0

Example 2: Unstable equilibrium in 2D

Consider the linear system,

$$\frac{dx}{dt} = x, \quad \frac{dy}{dt} = 2y.$$

Equilibrium:

$x = 0, \quad y = 0 \quad \Rightarrow \quad (x_e, y_e) = (0, 0).$

Behavior along each direction:

- x-direction: $\dot{x} = x \Rightarrow$ growth away from 0
- y-direction: $\dot{y} = 2y \Rightarrow$ growth away from 0

Conclusion

All nearby trajectories move away from the equilibrium. $\Rightarrow$ **unstable node**
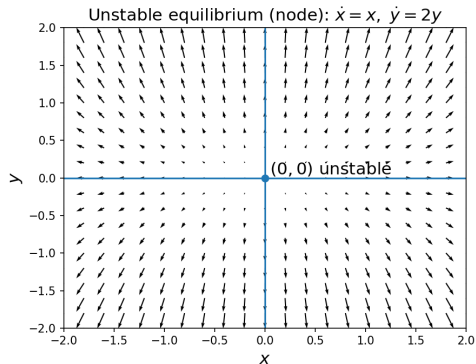

Figure: All solutions move away from $(x_e, y_e) = (0, 0)$ making this an unstable equilibrium point.

Example 3: Two equilibria with different stability

Consider the nonlinear system,

$$\frac{dx}{dt} = x - x^2, \quad \frac{dy}{dt} = -y + x - x^2.$$

Example 3: Two equilibria with different stability

Consider the nonlinear system,

$$\frac{dx}{dt} = x - x^2, \quad \frac{dy}{dt} = -y + x - x^2.$$

Equilibria:

$$x - x^2 = 0 \implies x = 0, 1$$
$$-y + x - x^2 = 0 \implies y = 0$$

So the equilibrium points are: $(0,0)$ and $(1,0)$.

Example 3: Two equilibria with different stability

Consider the nonlinear system,

$$\frac{dx}{dt} = x - x^2, \quad \frac{dy}{dt} = -y + x - x^2.$$

Equilibria:

$$x - x^2 = 0 \Rightarrow x = 0, 1$$
$$-y + x - x^2 = 0 \Rightarrow y = 0$$

So the equilibrium points are: $(0, 0)$ and $(1, 0)$.

Stability:

- $(0, 0)$: stable in one direction and unstable in another (saddle point)
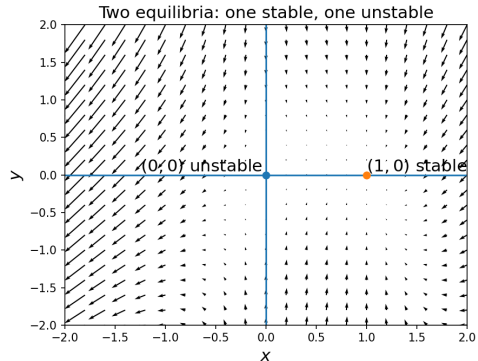- $(1, 0)$: trajectories decay toward the equilibrium in all directions

Figure: Two equilibrium points: (0,0) is unstable while (1,0) is stable.

Example 4: Saddle point equilibrium in 2D

Consider the linear system,

$$\frac{dx}{dt} = x, \quad \frac{dy}{dt} = -y.$$

Example 4: Saddle point equilibrium in 2D

Consider the linear system,

$$\frac{dx}{dt} = x, \quad \frac{dy}{dt} = -y.$$

Equilibrium: $x = 0, y = 0 \quad \Rightarrow \quad (x_e, y_e) = (0, 0).$

Example 4: Saddle point equilibrium in 2D

Consider the linear system,

$$\frac{dx}{dt} = x, \quad \frac{dy}{dt} = -y.$$

Equilibrium: $x = 0, y = 0 \quad \Rightarrow \quad (x_e, y_e) = (0, 0).$

Behavior along each direction:

- x-direction: $\dot{x} = x \Rightarrow$ solutions grow away from 0 (unstable)
- y-direction: $\dot{y} = -y \Rightarrow$ solutions decay toward 0 (stable)

Example 4: Saddle point equilibrium in 2D

Consider the linear system,

$$\frac{dx}{dt} = x, \quad \frac{dy}{dt} = -y.$$

Equilibrium: $x = 0, y = 0 \quad \Rightarrow \quad (x_e, y_e) = (0, 0).$

Behavior along each direction:

- x-direction: $\dot{x} = x \Rightarrow$ solutions grow away from 0 (unstable)
- y-direction: $\dot{y} = -y \Rightarrow$ solutions decay toward 0 (stable)

Conclusion

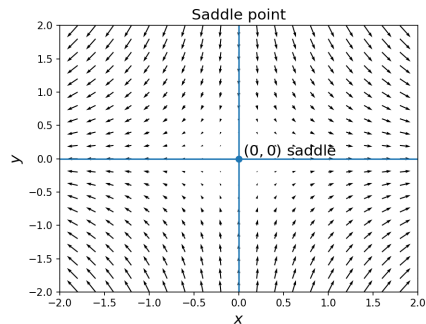The equilibrium is stable in one direction and unstable in another.

Figure: The equilibrium point $(0, 0)$ is unstable along x axis and stable along y axis, making it a saddle point.

Linear systems: equilibria

For a linear (affine) system,

$$\frac{d\mathbf{x}}{dt} = A\mathbf{x} + \mathbf{b},$$

equilibria satisfy,

$$A\mathbf{x}_e + \mathbf{b} = \mathbf{0}.$$

Linear systems: equilibria

For a linear (affine) system,

$$\frac{d\mathbf{x}}{dt} = A\mathbf{x} + \mathbf{b},$$

equilibria satisfy,

$$A\mathbf{x}_e + \mathbf{b} = \mathbf{0}.$$

- If $\mathbf{b} = \mathbf{0}$, then $\mathbf{x}_e = \mathbf{0}$ is always an equilibrium.
- If A is invertible, then the equilibrium is

$$\mathbf{x}_e = -A^{-1}\mathbf{b}.$$

Preview: from equilibria to eigenvalues

To analyze behavior *near* an equilibrium, we study how small perturbations evolve.

This motivates linear models of the form

$$\frac{d\mathbf{u}}{dt} = A\mathbf{u},$$

and leads directly to eigenvalues/eigenvectors as the tools for stability and "modes".

Takeaway

- Equilibrium point $\mathbf{x}_e$ means $\mathbf{f}(\mathbf{x}_e) = \mathbf{0}$.
- Starting at $\mathbf{x}_e$ gives a constant solution $\mathbf{x}(t) \equiv \mathbf{x}_e$.
- Understanding motion *near* equilibria sets up stability and eigenvalue analysis.

The key idea: eigenvectors are modes

For the linear system,

$$\mathbf{x}'(t) = A\mathbf{x}(t),$$

eigenpairs $(\lambda, \mathbf{v})$ of A satisfy,

$$A\mathbf{v} = \lambda\mathbf{v}.$$

A fundamental building block (a "mode") is,

$$\mathbf{x}(t) = e^{\lambda t}\,\mathbf{v}.$$

So, eigenvalues tell us:

- **how fast** a mode grows/decays (via $\Re(\lambda)$)
- whether it **oscillates** (via $\Im(\lambda)$)

From modes to the general solution

For linear systems,

$$\mathbf{x}'(t) = A\mathbf{x}(t),$$

the full solution is built by **combining modes**.

From modes to the general solution

For linear systems,

$$\mathbf{x}'(t) = A\mathbf{x}(t),$$

the full solution is built by **combining modes**. If A has n independent eigenpairs $(\lambda_k, \mathbf{v}_k)$,

then the solution can be written as

$$\mathbf{x}(t) = \sum_{k=1}^{n} c_k \, e^{\lambda_k t} \, \mathbf{v}_k,$$

where the constants c_k are determined by the initial condition $\mathbf{x}(0)$.

From modes to the general solution

For linear systems,

$$\mathbf{x}'(t) = A\mathbf{x}(t),$$

the full solution is built by **combining modes**. If A has n independent eigenpairs $(\lambda_k, \mathbf{v}_k)$,

then the solution can be written as

$$\mathbf{x}(t) = \sum_{k=1}^{n} c_k \, e^{\lambda_k t} \, \mathbf{v}_k,$$

where the constants c_k are determined by the initial condition $\mathbf{x}(0)$.

- Each term is an independent **eigenmode**
- The system's behavior is the **superposition** of all modes
- Modes with larger $\Re(\lambda)$ dominate at long times

Key message: eigenvalues and eigenvectors fully determine the qualitative behavior of the system.

Important note: when do exponential modes apply?

So far, we have described solutions of $\mathbf{x}' = A\mathbf{x}$ using exponential modes of the form,

$$\mathbf{x}(t) = e^{\lambda t}\mathbf{v}.$$

Important clarification:
- This description is exact when the matrix A is **diagonalizable**
- Diagonalizable means A has enough independent eigenvectors to form a basis
- In that case, the solution can be written as a combination of exponential eigenmodes

If A is **not** diagonalizable:
- solutions may involve additional polynomial factors multiplying $e^{\lambda t}$
- the analysis becomes more technical

For this course

You do *not* need to know these details. Just remember that superposition of exponential eigenmodes is *not* the most general solution for a general matrix A, and a full treatment is beyond the scope of EOSC 213.

Real eigenvalues: sign matters

If $\lambda \in \mathbb{R}$:

$$e^{\lambda t} = \begin{cases} \text{decays to 0} & \lambda < 0, \\ \text{stays constant} & \lambda = 0, \\ \text{grows (blows up)} & \lambda > 0. \end{cases}$$

For 2D systems:

- both eigenvalues $< 0 \Rightarrow$ stable (trajectories go to origin)
- one > 0 and one $< 0 \Rightarrow$ saddle (some directions grow, some decay)
- both $> 0 \Rightarrow$ unstable (trajectories move away)

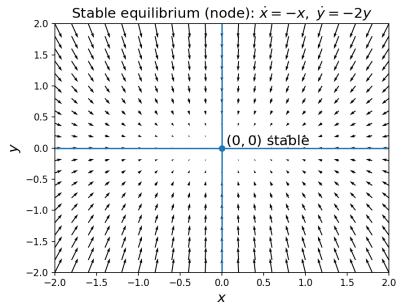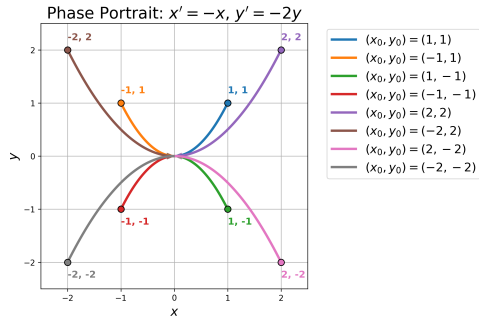# Example 1: Stable node (both eigenvalues negative)

Take

$$A = \begin{bmatrix} -1 & 0 \\ 0 & -2 \end{bmatrix}$$

Eigenvalues: $\lambda_1 = -1$, $\lambda_2 = -2$ (both negative).

Qualitative behavior:

- all trajectories decay toward $(0, 0)$
- y decays faster than x (two time scales)
- phase portrait looks like an inward "sink"

Phase Portrait: $x' = -x$, $y' = -2y$

Stable equilibrium (node): $\dot{x} = -x$, $\dot{y} = -2y$

Example 2: Saddle
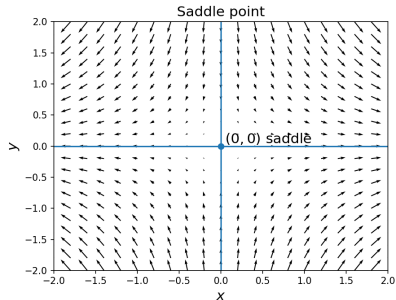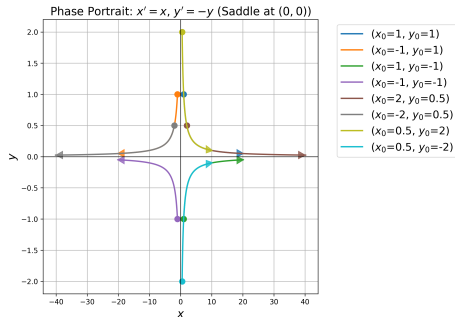# (one positive, one negative)

Take

$$A = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

Eigenvalues: $\lambda_1 = 1$ (growth), $\lambda_2 = -1$ (decay).

Qualitative behavior:

- along one direction trajectories move away from origin
- along the other direction trajectories move toward origin
- phase portrait shows a **saddle** (stable manifold + unstable manifold)

Phase Portrait: $x' = x$, $y' = -y$ (Saddle at (0, 0))

- $(x_0=1, y_0=1)$
- $(x_0=-1, y_0=1)$
- $(x_0=1, y_0=-1)$
- $(x_0=-1, y_0=-1)$
- $(x_0=2, y_0=0.5)$
- $(x_0=-2, y_0=0.5)$
- $(x_0=0.5, y_0=2)$
- $(x_0=0.5, y_0=-2)$

Saddle point

(0, 0) saddle

Complex eigenvalues (spirals)

If eigenvalues are complex:

$$\lambda = a \pm ib \quad (a, b \in \mathbb{R}, \ b \neq 0),$$

then

$$e^{\lambda t} = e^{at} \big(\cos(bt) + i \sin(bt) \big).$$

Interpretation:

- $\Re(\lambda) = a$ controls growth/decay of the amplitude
- $\Im(\lambda) = b$ controls oscillation frequency

So trajectories typically spiral.

Complex eigenvalues: Spiral sink vs. spiral source

For $\lambda = a \pm ib$:

- $a < 0$: amplitude decays $\Rightarrow$ **spiral sink** (stable spiral)
- $a > 0$: amplitude grows $\Rightarrow$ **spiral source** (unstable spiral)
- $a = 0$: constant amplitude $\Rightarrow$ **center** (closed orbits)

Takeaway: $\Re(\lambda)$ tells stability, $\Im(\lambda)$ tells oscillation.

Example 1: Spiral sink (complex eigenvalues with negative real part)

Take

$$A = \begin{bmatrix} -1 & -3 \\ 3 & -1 \end{bmatrix}$$

Eigenvalues: $\lambda_{1,2} = -1 \pm 3i$

This matrix produces rotation (off-diagonal terms or imaginary part of eigenvalues) plus decay (negative diagonal terms or real part of eigenvalues).

Qualitative behavior:

- trajectories rotate while shrinking
- phase portrait spirals into the origin

Phase Portrait & Vector Field: Spiral Sink ($A = [[-1, -3], [3, -1]]$)
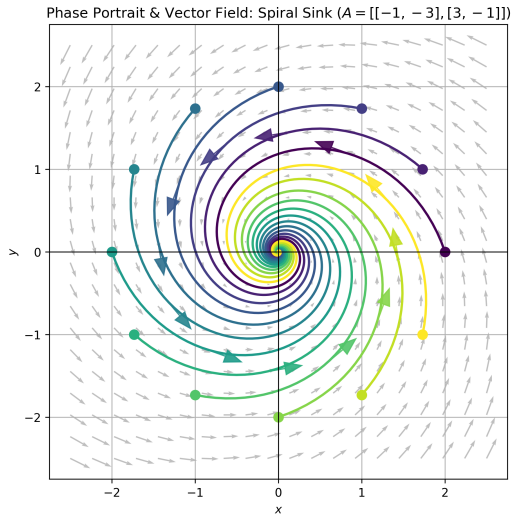
Figure: Spiral sink: complex eigenvalue with negative real part.

Example 2: Spiral source (complex eigenvalues with positive real part)

Take

$$A = \begin{bmatrix} 1 & -3 \\ 3 & 1 \end{bmatrix}$$

Eigenvalues: $\lambda_{1,2} = 1 \pm 3i$

This matrix produces rotation (off-diagonal terms or imaginary part of eigenvalues) plus decay (positive diagonal terms or real part of eigenvalues).

Qualitative behavior:

- trajectories rotate and move outward
- phase portrait spirals out away from the origin towards infinity

Phase Portrait & Vector Field: Spiral Source ($A = [[1, -3], [3, 1]]$)
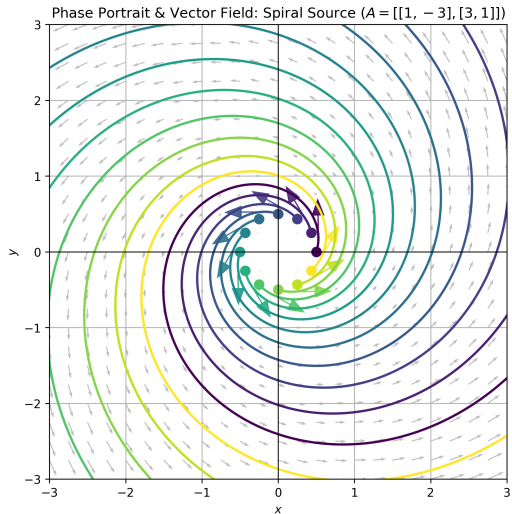
Figure: Spiral source: complex eigenvalue with positive real part.

Example 3: Closed orbits (purely imaginary eigenvalues)

Take the matrix for the undamped oscillator,

$$A = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

Eigenvalues: $\lambda_{1,2} = \pm i$

Qualitative behavior:

- no decay, no growth
- closed orbits around the origin
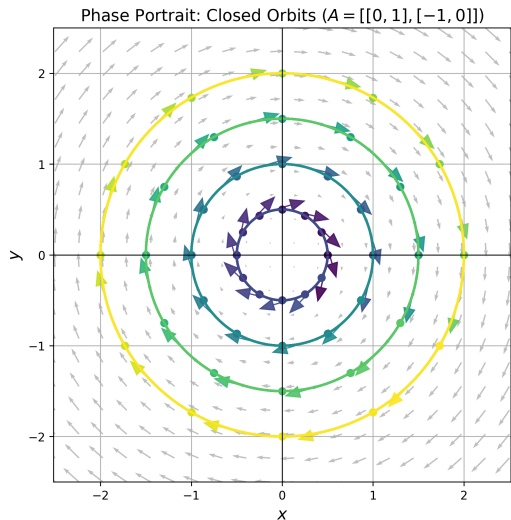- sustained oscillations (constant amplitude)

Figure: Closed orbits: eigenvalue are purely imaginary.

Quick classification summary (2D)

For $\dot{\mathbf{x}} = A\mathbf{x}$ with eigenvalues λ_1, λ_2:

- $\lambda_1, \lambda_2 < 0 \quad \Rightarrow$ stable node (sink)
- $\lambda_1, \lambda_2 > 0 \quad \Rightarrow$ unstable node (source)
- opposite signs $\quad \Rightarrow$ saddle
- $a \pm ib$ with $a < 0 \quad \Rightarrow$ spiral sink
- $a \pm ib$ with $a > 0 \quad \Rightarrow$ spiral source
- $\pm ib$ (pure imaginary) $\quad \Rightarrow$ center

Eigenvalues $\Rightarrow$ qualitative phase portrait.

A useful diagnostic: trace and determinant (2D)

For

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

$$\text{tr}(A) = a + d, \qquad \det(A) = ad - bc.$$

Eigenvalues satisfy:

$$\lambda^2 - \text{tr}(A)\lambda + \det(A) = 0.$$

The discriminant

$$\Delta = \text{tr}(A)^2 - 4\det(A)$$

tells if eigenvalues are real ($\Delta \geq 0$) or complex ($\Delta < 0$).

Systems look the same to numerical methods

For a general system:

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t, \mathbf{x}(t)), \qquad \mathbf{x}(t_0) = \mathbf{x}_0,$$

numerical methods work almost exactly like the scalar case:

- replace x by $\mathbf{x}$
- replace f by $\mathbf{F}$
- updates become vector updates

The key difference: **vectors and (sometimes) linear solves**.

Forward Euler for systems

Forward Euler:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + h\,\mathbf{F}(t_n, \mathbf{x}_n).$$

- explicit: easy to implement
- one function evaluation per step
- stability can still restrict h

Explicit Midpoint for systems

Explicit Midpoint:

Half-step:

$$\mathbf{x}_{n+\frac{1}{2}} = \mathbf{x}_n + \frac{h}{2}\mathbf{F}(t_n, \mathbf{x}_n)$$

Full step:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + h\,\mathbf{F}\left(t_n + \frac{h}{2}, \mathbf{x}_{n+\frac{1}{2}}\right)$$

- second-order accurate
- still explicit
- still can have a stability limit

Backward Euler for systems

Backward Euler:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + h \, \mathbf{F}(t_{n+1}, \mathbf{x}_{n+1}).$$

- implicit: $\mathbf{x}_{n+1}$ appears on both sides
- requires solving an equation (often nonlinear) each step
- can be much more stable for stiff decay problems

Special case: linear system $\dot{\mathbf{x}} = A\mathbf{x}$

Now let

$$\dot{\mathbf{x}} = A\mathbf{x}$$

with constant matrix A.

Then numerical methods become:

- matrix-vector multiplications (explicit methods)
- linear system solves (implicit methods)

This is where **amplification matrices** appear.

Forward Euler on $\dot{\mathbf{x}} = A\mathbf{x}$

Forward Euler:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + hA\mathbf{x}_n = (I + hA)\mathbf{x}_n.$$

Amplification matrix:

$$G_{\mathsf{FE}} = I + hA.$$

After n steps:

$$\mathbf{x}_n = (I + hA)^n \mathbf{x}_0.$$

Backward Euler on $\dot{\mathbf{x}} = A\mathbf{x}$

Backward Euler:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + hA\mathbf{x}_{n+1}.$$

Rearrange:

$$(I - hA)\mathbf{x}_{n+1} = \mathbf{x}_n$$

so

$$\mathbf{x}_{n+1} = (I - hA)^{-1}\mathbf{x}_n.$$

Amplification matrix:

$$G_{\text{BE}} = (I - hA)^{-1}.$$

Explicit Midpoint on $\dot{\mathbf{x}} = A\mathbf{x}$

Half-step:

$$\mathbf{x}_{n+\frac{1}{2}} = \mathbf{x}_n + \frac{h}{2}A\mathbf{x}_n = \left(I + \frac{h}{2}A\right)\mathbf{x}_n.$$

Full step:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + hA\mathbf{x}_{n+\frac{1}{2}} = \mathbf{x}_n + hA\left(I + \frac{h}{2}A\right)\mathbf{x}_n.$$

So:

$$\mathbf{x}_{n+1} = \left(I + hA + \frac{(hA)^2}{2}\right)\mathbf{x}_n.$$

Amplification matrix:

$$G_{\text{EM}} = I + hA + \frac{(hA)^2}{2}.$$

Stability idea for systems (same philosophy)

For decay systems, we want the numerical method to **damp** modes.

For a matrix update:

$$\mathbf{x}_{n+1} = G\mathbf{x}_n,$$

stability is governed by eigenvalues of G.

Rule of thumb:

$$|\mu| < 1 \quad \text{for all eigenvalues } \mu \text{ of } G.$$

So again: **eigenvalues determine stability**.

Eigenvalues of G relate to eigenvalues of A

If $A\mathbf{v} = \lambda\mathbf{v}$, then:

Forward Euler:

$$G_{\text{FE}}\mathbf{v} = (I + hA)\mathbf{v} = (1 + h\lambda)\mathbf{v} \quad \Rightarrow \quad \mu = 1 + h\lambda.$$

Backward Euler:

$$G_{\text{BE}}\mathbf{v} = (I - hA)^{-1}\mathbf{v} = \frac{1}{1 - h\lambda}\mathbf{v} \quad \Rightarrow \quad \mu = \frac{1}{1 - h\lambda}.$$

Explicit Midpoint:

$$\mu = 1 + h\lambda + \frac{(h\lambda)^2}{2}.$$

Interpretation: stability reduces to checking each eigenvalue

For $\dot{\mathbf{x}} = A\mathbf{x}$, each eigenvalue λ of A behaves like a scalar mode.

So stability is controlled by the set $\{h\lambda_k\}$.

- if some λ_k is large negative (fast decay), explicit methods may require very small h
- this is the beginning of **stiffness**
- implicit methods can remain stable for much larger h

A quick example: choosing h from eigenvalues

Suppose A has eigenvalues:

$$\lambda_1 = -1, \qquad \lambda_2 = -50.$$

Forward Euler stability on each mode requires roughly:

$$|1 + h\lambda_k| < 1 \quad \Rightarrow \quad 0 < h < \frac{2}{|\lambda_k|}.$$

So the tightest constraint comes from -50:

$$h < \frac{2}{50} = 0.04.$$

Even though one mode is "slow" (-1), the "fast" mode forces a tiny step size.

Computational note: what changes in code?

Forward Euler / Midpoint:

- compute $\mathbf{F}(t, \mathbf{x})$ and do vector arithmetic
- (for linear systems) compute $A\mathbf{x}$ via matrix-vector product

Backward Euler:

- solve a system each step
- linear case: solve $(I - hA)\mathbf{x}_{n+1} = \mathbf{x}_n$
- nonlinear case: often use Newton or fixed-point iteration

Summary: what to remember

- For $\dot{\mathbf{x}} = A\mathbf{x}$, eigenvalues predict behavior:
  - $\Re(\lambda) < 0$ decay (stable), $\Re(\lambda) > 0$ growth (unstable)
  - $\Im(\lambda) \neq 0$ introduces oscillations (spirals/centers)
- Forward Euler / Midpoint / Backward Euler extend to systems by replacing x with $\mathbf{x}$
- For linear systems, methods become $\mathbf{x}_{n+1} = G\mathbf{x}_n$ with an amplification matrix G
- Numerical stability depends on eigenvalues of G, which are functions of eigenvalues of A