Outline

1 Recap

2 Boundary value problems
   - Shooting method

3 Eigenvalue problems
   - The Schrödinger equation
   - Shooting eigenvalues

4 Summary

Last quiz next week!
Recap

An \( n \)-th order ODE or \( n \) coupled ODEs require \( n \) conditions to be specified.

**Initial value problems:** All conditions are specified at the same point:

\[
y(x_1) = y_1, \quad y'(x_1) = v_1, \quad y''(x_1) = a_1 \quad \text{etc.}
\]

**Boundary value problems:** Conditions specified at different points:

\[
y(0) = a_0; \quad y(1) = b_0 \quad \text{or}
\]
\[
y'(0) = a_1; \quad b_0 y(1) + b_1 y'(1) = c \quad \text{etc}
\]

- Unique solution not guaranteed
- Must usually start with a guess
- **Shooting method:** Convert to initial value problem by guessing missing initial values, adjust until boundary values are satisfied.
- **Relaxation method:** Guess solution on entire domain with correct boundary conditions, adjust until ODE is satisfied.
Shooting method

Shoot first, ask questions later!

"Hitting the target" is a question of reducing the discrepancy $y(x_2) - b$ to zero.

For a second-order ode we can use bisection, secant, Ridders',... Higher order: Newton–Raphson.
Shooting method

Shoot first, ask questions later!

1. Guess the unknown initial conditions [load & aim]

"Hitting the target" is a question of reducing the discrepancy $y(x^2) - b$ to zero.

For a second-order ode we can use bisection, secant, Ridders',... Higher order: Newton–Raphson.
Shooting method

Shoot first, ask questions later!

1. Guess the unknown initial conditions [load & aim]
2. Integrate system of odes to final point [shoot]
Shooting method

Shoot first, ask questions later!

1. Guess the unknown initial conditions [load & aim]
2. Integrate system of odes to final point [shoot]
3. Check how close you got [ask questions]

"Hitting the target" is a question of reducing the discrepancy \( \rightarrow y(x^2) - \rightarrow b \) to zero.

For a second-order ode we can use bisection, secant, Ridders',... Higher order: Newton–Raphson.
Shooting method

Shoot first, ask questions later!

1. Guess the unknown initial conditions [load & aim]
2. Integrate system of odes to final point [shoot]
3. Check how close you got [ask questions]
4. Adjust your aim, goto 1
Shooting method

Shoot first, ask questions later!

1. Guess the unknown initial conditions [load & aim]
2. Integrate system of odes to final point [shoot]
3. Check how close you got [ask questions]
4. Adjust your aim, goto 1

“Hitting the target” is a question of reducing the discrepancy $\overrightarrow{y}(x_2) - \overrightarrow{b}$ to zero $= \text{root finding}$

For a second-order ode we can use bisection, secant, Ridders’,...
Shooting method

Shoot first, ask questions later!

1. Guess the unknown initial conditions [load & aim]
2. Integrate system of odes to final point [shoot]
3. Check how close you got [ask questions]
4. Adjust your aim, goto 1

“Hitting the target” is a question of reducing the discrepancy $\vec{y}(x_2) - \vec{b}$ to zero $=$ root finding

For a second-order ode we can use bisection, secant, Ridders’,…
Higher order: Newton–Raphson
Shooting method

1. We want
   \[ y(x_1) = a, \ y(x_2) = b. \]
1. We want
   \[ y(x_1) = a, \, y(x_2) = b. \]
2. We guess \( y'(x_1) = v_1 \) and get \( y(x_2) = b_1 < b \).
Shooting method

1. We want
   \[ y(x_1) = a, \quad y(x_2) = b. \]

2. We guess \( y'(x_1) = v_1 \) and get \( y(x_2) = b_1 < b \).

3. We guess \( y'(x_1) = v_2 \) and get \( y(x_2) = b_2 > b \).
Shooting method

1. We want
   \[ y(x_1) = a, \quad y(x_2) = b. \]

2. We guess \( y'(x_1) = v_1 \) and get \( y(x_2) = b_1 < b. \)

3. We guess \( y'(x_1) = v_2 \) and get \( y(x_2) = b_2 > b. \)

4. The correct value for \( y'(x_1) \) is between \( v_1 \) and \( v_2. \)
Shooting method

1. We want
   \[ y(x_1) = a, \quad y(x_2) = b. \]
2. We guess \( y'(x_1) = v_1 \) and get \( y(x_2) = b_1 < b. \)
3. We guess \( y'(x_1) = v_2 \) and get \( y(x_2) = b_2 > b. \)
4. The correct value for \( y'(x_1) \) is between \( v_1 \) and \( v_2. \)

We can find the value for \( y'(x_1) \) using bisection!
This will give us the solution of the boundary value problem.
Eigenvalue problems

Consider 1-dimensional Schrödinger equation:

$$H\psi = \left[ -\frac{\hbar^2}{2m} \frac{\partial^2}{\partial x^2} + V(x) \right] \psi = i\hbar \frac{\partial \psi}{\partial t} = E\psi$$

with boundary conditions $\psi(-\infty) = \psi(\infty) = 0$
Eigenvalue problems

Consider 1-dimensional Schrödinger equation:

\[ H\psi = \left[ -\frac{\hbar^2}{2m}\frac{\partial^2}{\partial x^2} + V(x) \right] \psi = i\hbar \frac{\partial \psi}{\partial t} = E\psi \]

with boundary conditions \( \psi(-\infty) = \psi(\infty) = 0 \)

Solutions

1. \( \psi \equiv 0 \) (for any \( E \))
2. \( \psi_n \neq 0 \) for some \( E = E_n \)

The set of \( E_n \) is the spectrum
Eigenvalue problems

Consider 1-dimensional Schrödinger equation:

\[ H\psi = \left[ -\frac{\hbar^2}{2m} \frac{\partial^2}{\partial x^2} + V(x) \right] \psi = i\hbar \frac{\partial \psi}{\partial t} = E\psi \]

with boundary conditions \( \psi(-\infty) = \psi(\infty) = 0 \)

Solutions

1. \( \psi \equiv 0 \) (for any \( E \))
2. \( \psi_n \neq 0 \) for some \( E = E_n \)

The set of \( E_n \) is the spectrum

The spectral problem: Find all \( E_n, \psi_n \)
Consider 1-dimensional Schrödinger equation:

\[ H\psi = \left[ -\frac{\hbar^2}{2m} \frac{\partial^2}{\partial x^2} + V(x) \right] \psi = i\hbar \frac{\partial \psi}{\partial t} = E\psi \]

with boundary conditions \( \psi(-\infty) = \psi(\infty) = 0 \)

**Solutions**

1. \( \psi \equiv 0 \) (for any \( E \))
2. \( \psi_n \neq 0 \) for some \( E = E_n \)

The set of \( E_n \) is the **spectrum**

**The spectral problem:** Find all \( E_n, \psi_n \)

**Possible solution:** Treat \( E_n \) as additional variable, shoot!
Exploiting symmetry

Problem is greatly simplified if $V(x) = V(-x)$

$$\implies \begin{cases} 
\Psi_n(-x) = \Psi_n(x) & \text{even} \\
\Psi_n(-x) = -\Psi_n(x) & \text{odd}
\end{cases}$$
Exploiting symmetry

Problem is greatly simplified if \( V(x) = V(-x) \)

\[
\begin{align*}
\Psi_n(-x) &= \Psi_n(x) \quad \text{even} \\
\Psi_n(-x) &= -\Psi_n(x) \quad \text{odd}
\end{align*}
\]

But \( \Psi(0) = \Psi(-0) \)

\[
\begin{align*}
\Psi_n(0) &\neq 0, \quad \Psi_n'(0) = 0 \quad \text{even} \\
\Psi_n(0) &= 0, \quad \Psi_n'(0) \neq 0 \quad \text{odd}
\end{align*}
\]

So we can start at \( x = 0 \), not \( x = -\infty \)
Exploiting symmetry

Problem is greatly simplified if $V(x) = V(-x)$

$$\Rightarrow \begin{cases} 
\psi_n(-x) = \psi_n(x) & \text{even} \\
\psi_n(-x) = -\psi_n(x) & \text{odd}
\end{cases}$$

But $\psi(0) = \psi(-0)$

$$\Rightarrow \begin{cases} 
\psi_n(0) \neq 0, \quad \psi'_n(0) = 0 & \text{even} \\
\psi_n(0) = 0, \quad \psi'_n(0) \neq 0 & \text{odd}
\end{cases}$$

So we can start at $x = 0$, not $x = -\infty$

We can also set $\psi(0) = 1$ or $\psi'(0) = 1$ since the value is irrelevant (will normalise solution at end)
Shooting eigenvalues

1. Set $\Psi(0) = 1$, $\Psi'(0) = 0$ (even) or $\Psi(0) = 0$, $\Psi'(0) = 1$ (odd)
Shooting eigenvalues

1. Set $\Psi(0) = 1$, $\Psi'(0) = 0$ (even) or $\Psi(0) = 0$, $\Psi'(0) = 1$ (odd)
2. Choose some $E$
Shooting eigenvalues

1. Set $\Psi(0) = 1$, $\Psi'(0) = 0$ (even) or $\Psi(0) = 0$, $\Psi'(0) = 1$ (odd)
2. Choose some $E$
3. Integrate ODE for $\Psi(x)$, $x \in [0, \infty)$
   In practice: $x \in [0, L]$, $L \sim 10$ depending on problem
Shooting eigenvalues

1. Set $\Psi(0) = 1$, $\Psi'(0) = 0$ (even) or $\Psi(0) = 0$, $\Psi'(0) = 1$ (odd)
2. Choose some $E$
3. Integrate ODE for $\Psi(x)$, $x \in [0, \infty)$
   In practice: $x \in [0, L]$, $L \sim 10$ depending on problem
4. Use root finding algorithm to solve $\Psi(x = L; E) = 0$ for $E$
Boundary value problems much harder than initial value
Must (usually) start by guessing a solution!
Summary

- Boundary value problems much harder than initial value
- Must (usually) start by guessing a solution!

**Shooting method:**
1. Guess unknown initial values $v_i$
2. Solve ODE with these values: $f(x|v_i)$
3. Find solution at final point $x_f$
4. Solve $f(x_f|v_i) - v_f = 0$ — root finding!
Summary

- Boundary value problems much harder than initial value
- Must (usually) start by guessing a solution!
- **Shooting method:**
  1. Guess unknown initial values $v_i$
  2. Solve ODE with these values: $f(x|v_i)$
  3. Find solution at final point $x_f$
  4. Solve $f(x_f|v_i) - v_f = 0$ — root finding!

- **Eigenvalue problems**
  - May be made into boundary value problem by treating eigenvalue as additional variable.