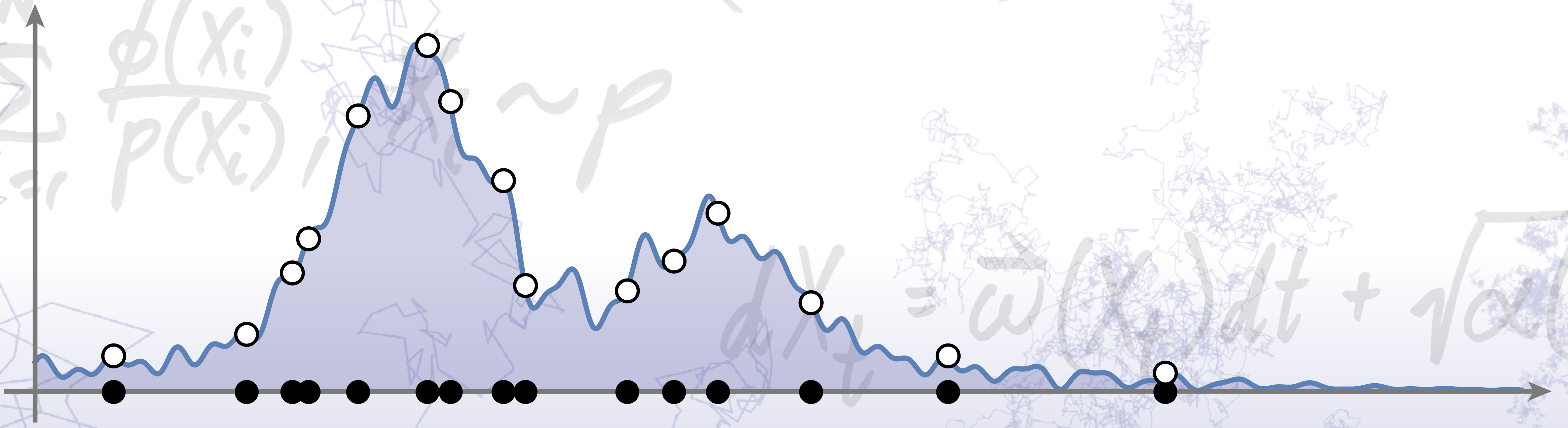
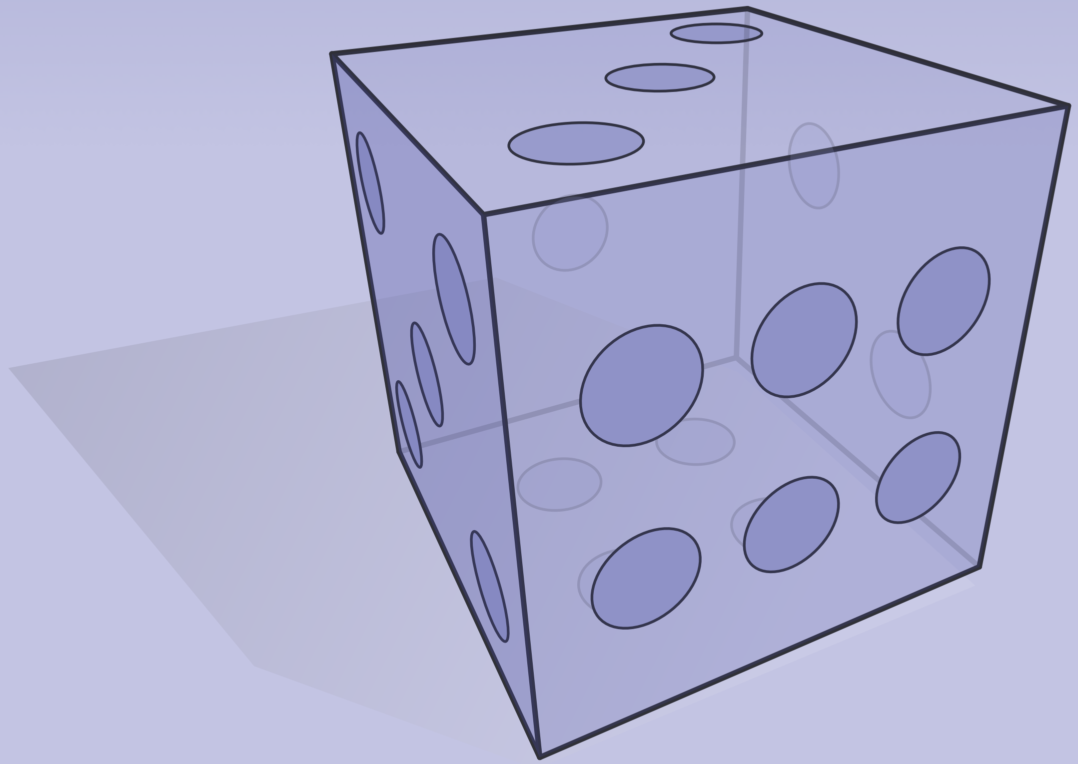
Monte Carlo Methods and Applications



Carnegie Mellon University | 21-387 / 15-327 / 15-627 / 15-860 | Fall 2023

LECTURE 12

PDES & STOCHASTIC PROCESSES



MONTE CARLO METHODS
AND APPLICATIONS

Carnegie Mellon University | 21-387 | 15-327 | 15-627 | 15-860 | Fall 2023

Recap—SDEs & PDEs

- **Ordinary & Stochastic Differential Equations** (last lecture)
  - how do we describe systems evolving *over time*? **(ODEs)**
  - how do we incorporate randomness? **(SDEs)**
  - how do we simulate motion numerically?
- **Partial Differential Equations** (this lecture)
  - how do we describe systems evolving over *time & space?* **(PDEs)**
  - how do we simulate these systems numerically?
- SDE ⟷ PDE connection
  - Somewhat surprising perspective: can use **stochastic** ODEs to understand—*and simulate*—**deterministic** PDEs
  - …and vice-versa!

analogy: *trajectory of rock (+wind)*

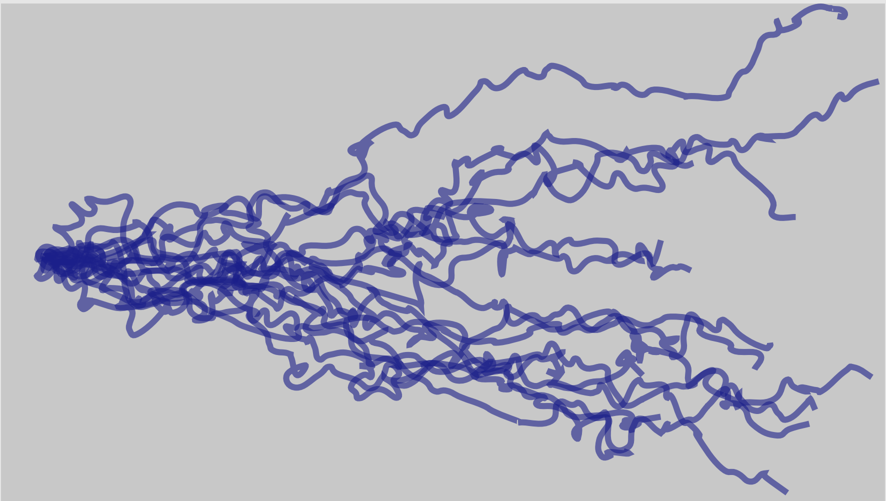
analogy: *ripples on pond*

Goal: Connect "microscopic" & "macroscopic"

Understand statements of two major **concepts**
and see how they can be used for **computation**.

Feynman-Kac formula

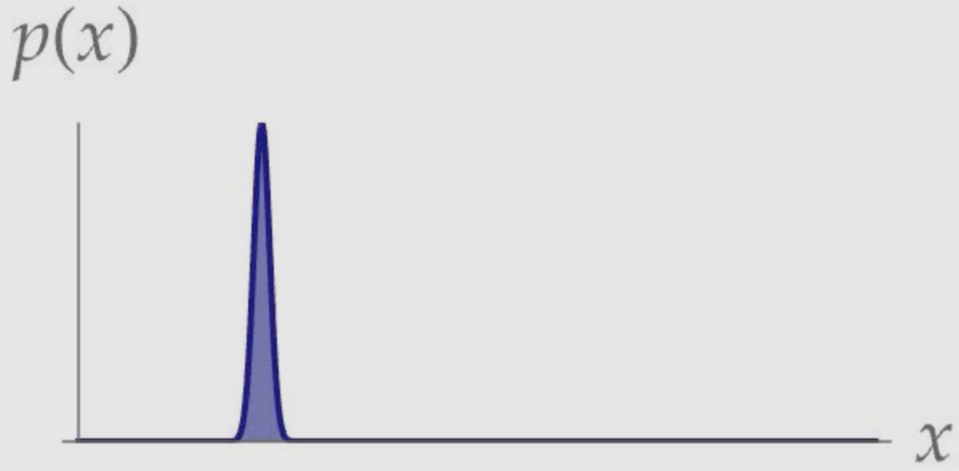
$$u(x)=\mathbb{E}\left[\int_0^T e^{-\int_0^t \sigma(X_s)ds} f(X_t)dt + e^{-\int_0^T \sigma(X_t)dt} g(X_t)\right]$$



use random walks to solve PDEs

Fokker-Planck equation

$$\frac{\partial p}{\partial t} = \alpha\Delta p - \nabla\cdot(p\omega)$$

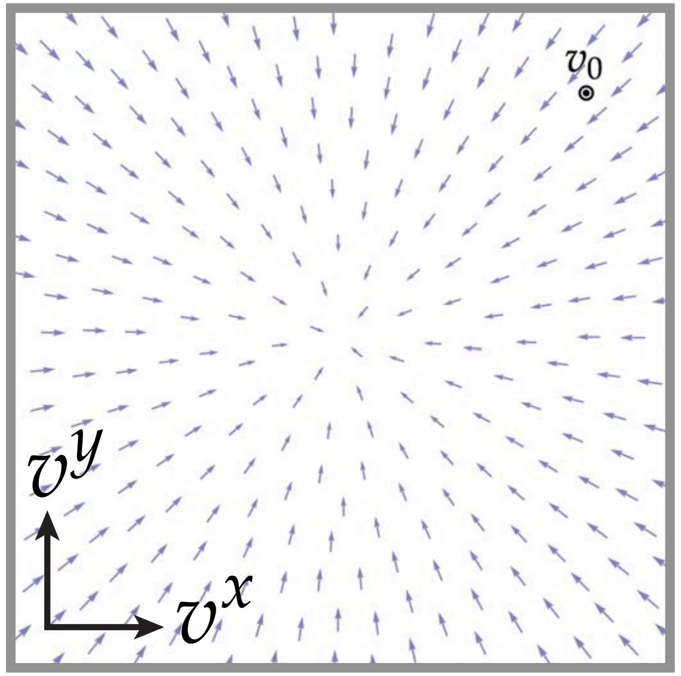


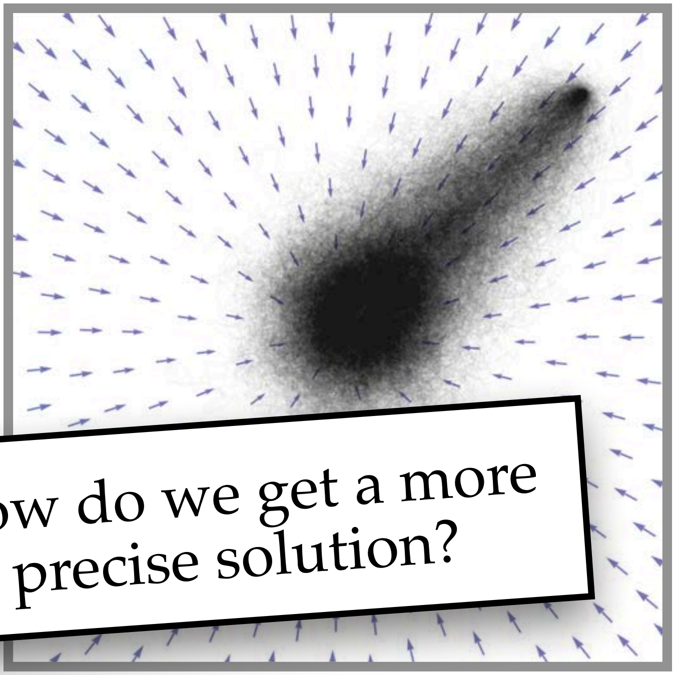
solve PDEs to model random walks

Today: will develop numerical algorithms to actually do this…

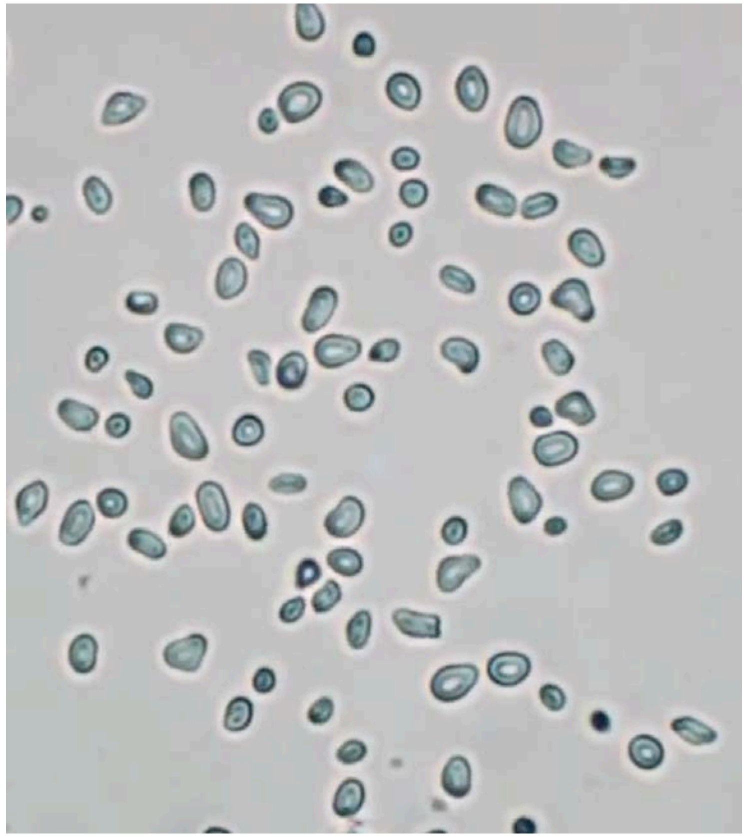
Motivating Example—Basic Langevin Equation

- Consider our famous "jiggling pollen grains"
- How do we write an equation for the <u>velocity</u> v_t of a single particle as a function of time t?
- In vector field picture, drift term always points toward origin (here the components are components of a d-dimensional velocity, rather than position & velocity)
- Hence, velocity tends toward zero (due to **drag force**), but then hovers around the origin (due to **Langevin force**)
- Numerically integrating SDE many times gives <u>rough</u> sense of how velocity is distributed over space & time, i.e., $P(v_t = w)$





How do we get a more precise solution?



$$dv_t = \boxed{-\gamma v_t} + \boxed{dW_t}$$

drag force

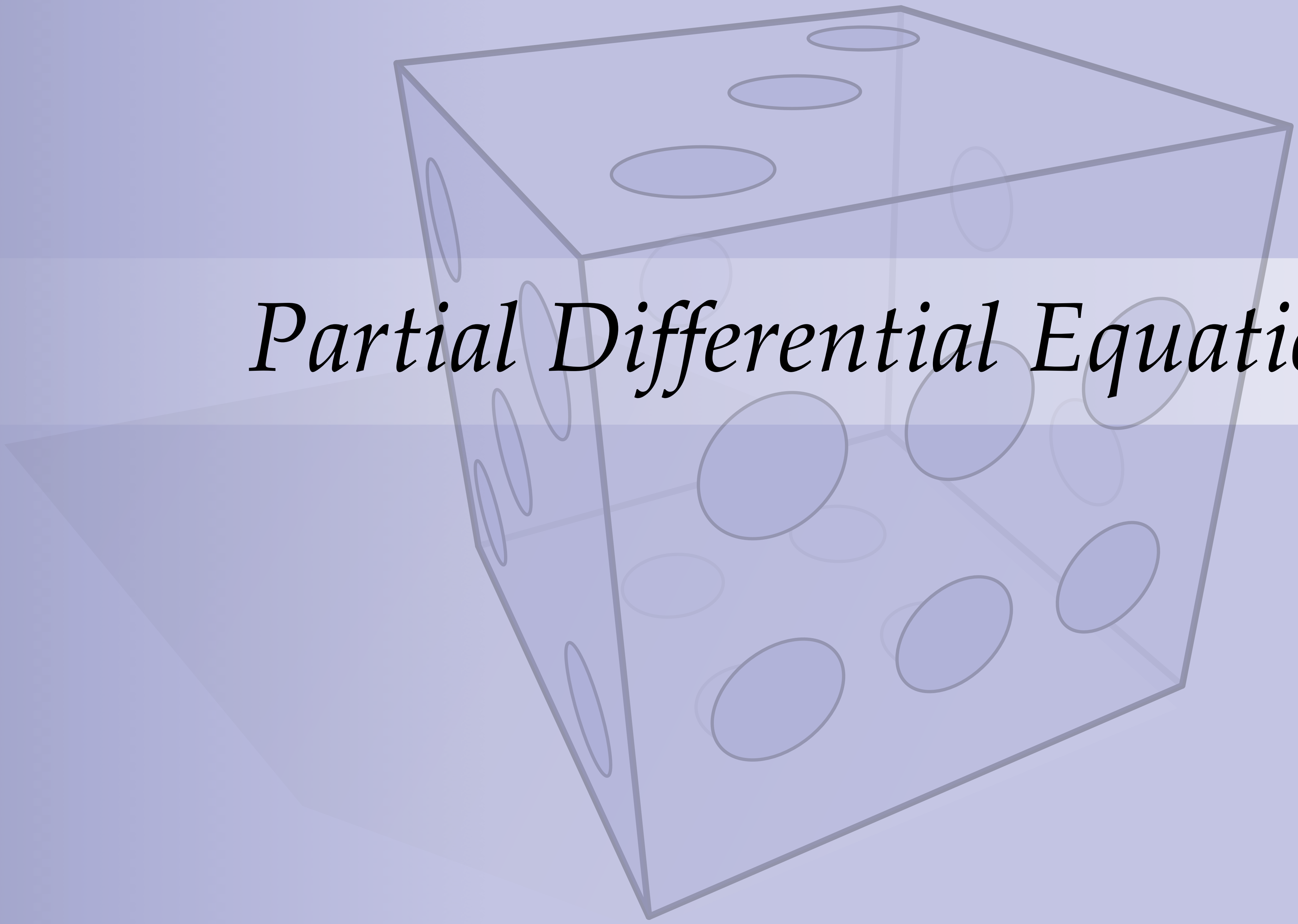
due to viscosity/ "friction"

Langevin force

due to thermal fluctuations of <u>liquid</u>

Overview—PDEs & Stochastic Processes

- **Partial Differential Equations (PDEs)**
  - describe function *implicitly* via relative rates of change in space & time
- **Numerical Methods for PDEs**
- **Feynman-Kac formula**
  - for a given PDE, can (sometimes) write down its solution in terms of a stochastic process X_t
  - can simulate this SDE numerically to compute solution to PDE
- **Fokker-Planck equation**
  - for a given stochastic process X_t, we can (sometimes) write down a PDE describing the evolution of its probability distribution over time
  - can solve this PDE numerically to evaluate average properties of X_t



Partial Differential Equations

Ordinary Differential Equation (Review)

Recall general formula for an **ordinary differential equation:**

$$F\left(t, x, x', ..., x^{(n)}\right) = 0$$

where F is any function of the (unknown) function $x(t)$ and its first n derivatives in time.

We said this ODE is:

- *nth order in time* (or simply *nth order*)
- *linear* (or *nonlinear*) if F is a linear (or nonlinear) function of its inputs

Partial Differential Equation

A **partial differential equation** is similar, but adds spatial dependence:

$$F\left(t, u, \frac{\partial u}{\partial t}, \frac{\partial^2 u}{\partial t^2}, \ldots, \frac{\partial u}{\partial x}, \frac{\partial^2 u}{\partial x^2}, \ldots\right) = 0, \quad t > 0, x \in \Omega$$

governing equation

$$\phi\left(t, u, \frac{\partial u}{\partial t}, \frac{\partial^2 u}{\partial t^2}, \ldots, \frac{\partial u}{\partial x}, \frac{\partial^2 u}{\partial x^2}, \ldots\right) = 0, \quad t > 0, x \in \partial\Omega$$

boundary conditions

$$u(x, 0) = u_0, \; x \in \Omega$$

initial conditions

space & time derivatives of any order

"nth order in space, mth order in time"

linear if F and ϕ are linear; **nonlinear** otherwise

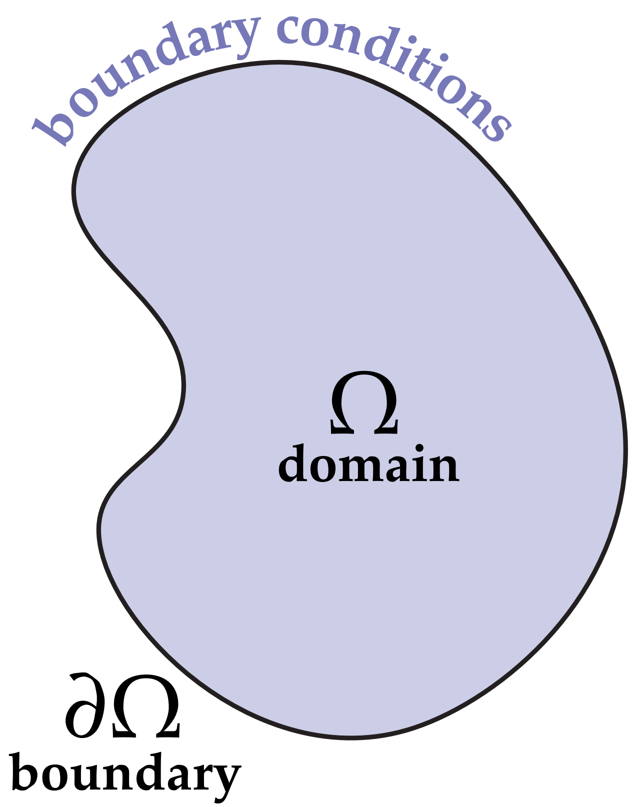
Like ODEs, nonlinear generally harder

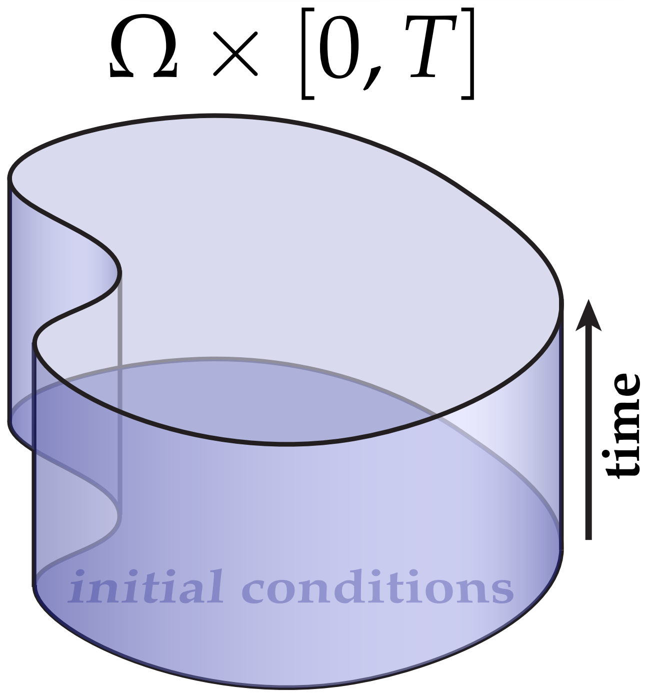
unknown function

$$\underbrace{u}_{\text{solution}} : \overset{\text{space}}{\Omega} \times \overset{\text{time}}{[0, T]} \to \mathbb{R}$$

could take other kinds of values

could be, e.g., $+\infty$





Example—1D Heat Equation

Example. Heat equation on the real line:

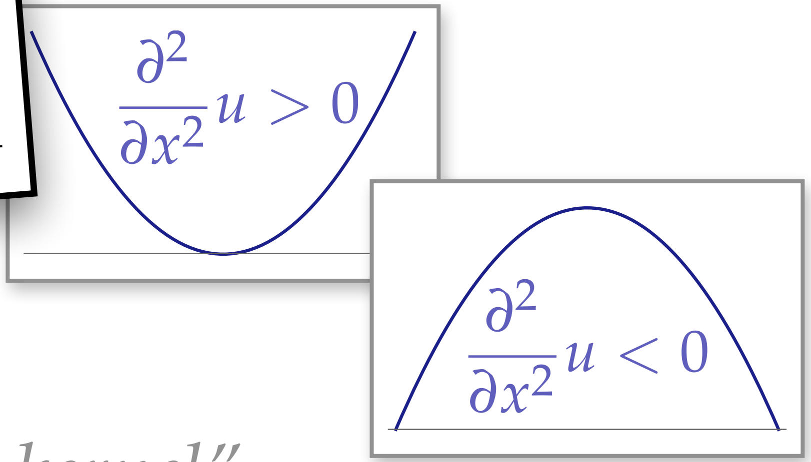
(no boundary conditions: domain has no boundary)

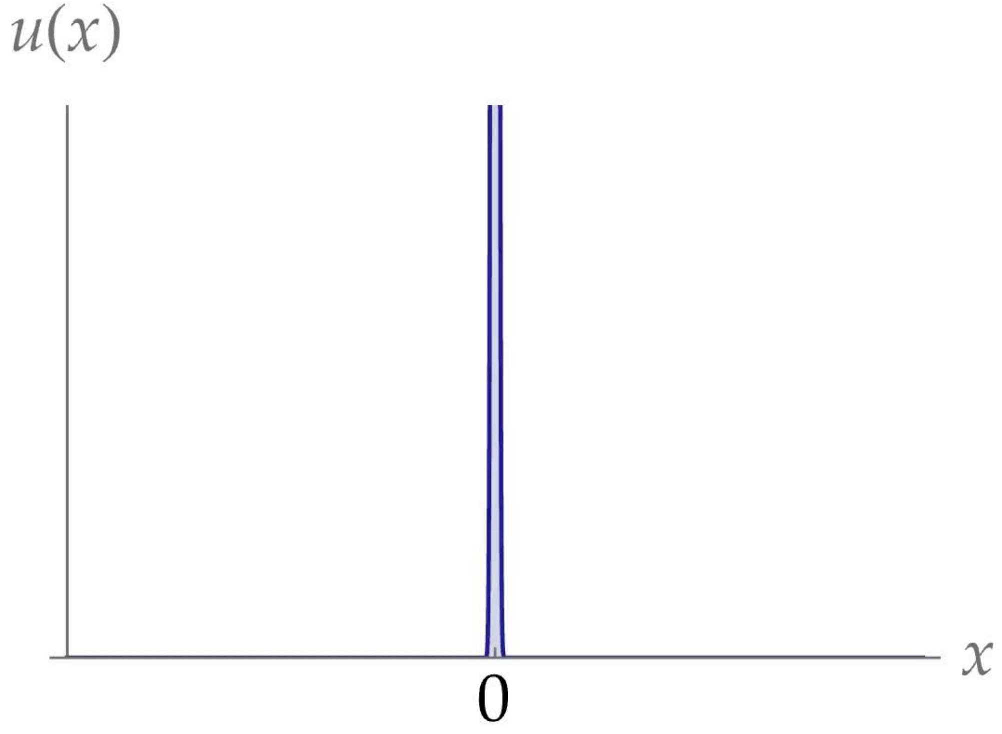
$$u : \mathbb{R} \times [0, \infty) \to \mathbb{R}$$
$$(\Omega = \mathbb{R}, T = +\infty)$$

$$\frac{\partial}{\partial t} u(x,t) = \frac{\partial^2}{\partial x^2} u(x,t), \; t > 0, \; x \in \mathbb{R}$$

$$u(x,0) = \delta(x), x \in \mathbb{R}$$

Intuition: concave / convex bumps get pushed up / down





Solution.

“heat kernel”

$$u(x,t) = \frac{1}{\sqrt{4\pi t}} e^{-x^2/(4t)}$$

Differential Operators

- In dimension $d > 1$, becomes ugly to write out PDE in terms of all d spatial directions
- Can instead express in terms of *differential operators*
  - divergence
  - gradient
  - curl
  - Laplacian
  - …
- (Also helps to drop arguments…)

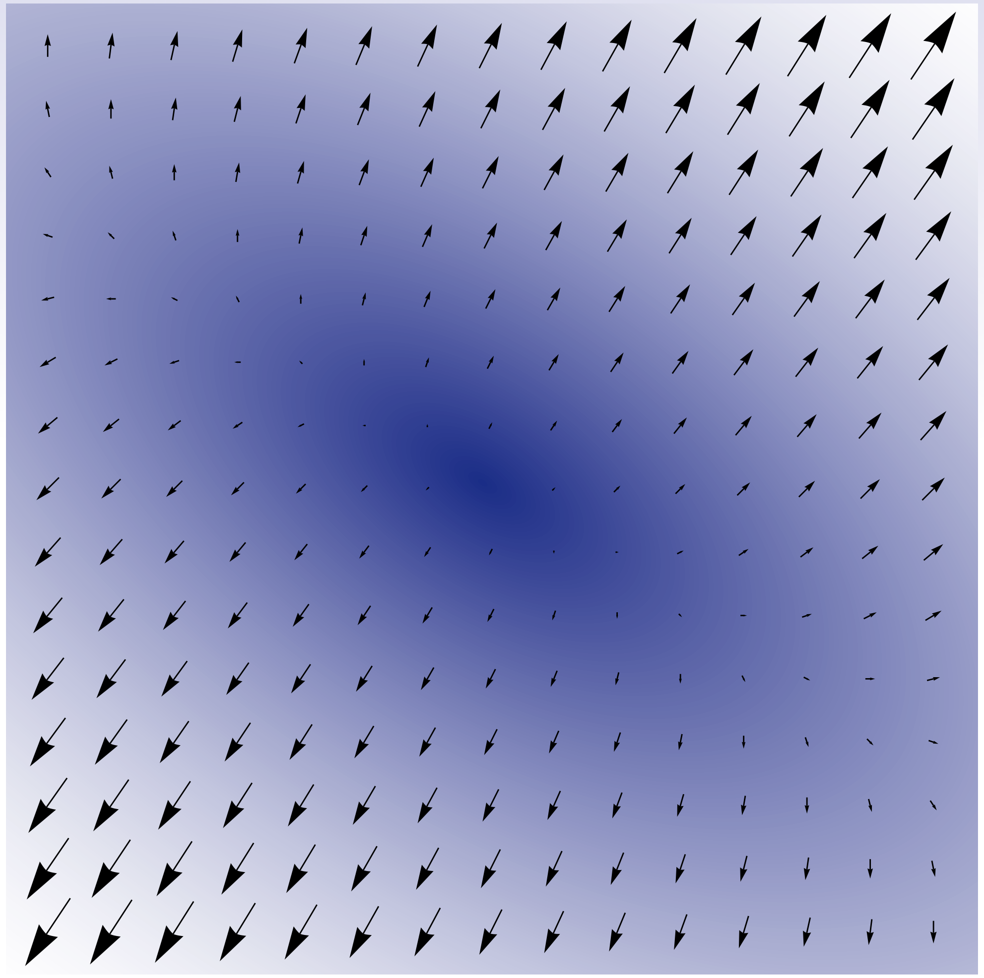
heat equation (component-wise)

$$\frac{\partial}{\partial t}u(x,y,z,t) = \frac{\partial^2}{\partial x^2}u(x,y,z,t) + \frac{\partial^2}{\partial y^2}u(x,y,z,t) + \frac{\partial^2}{\partial z^2}u(x,y,z,t)$$

heat equation (differential operator)

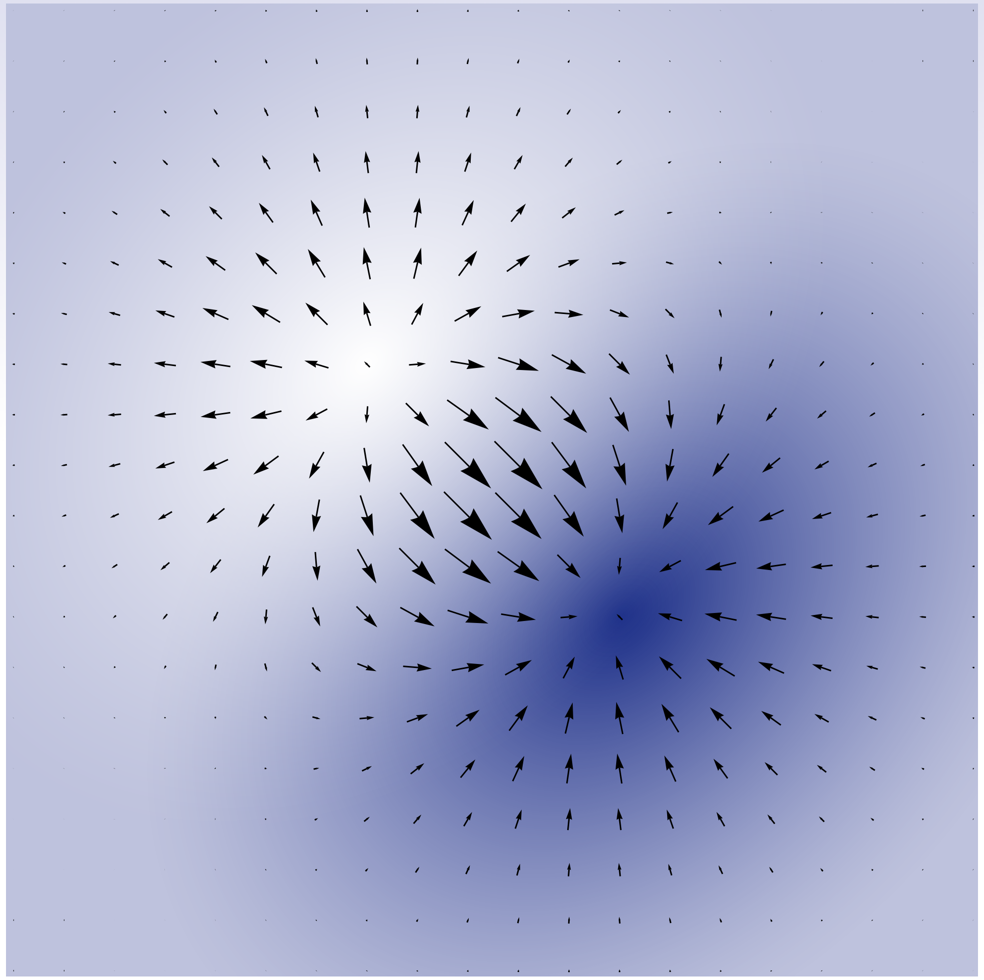
$$\frac{\partial}{\partial t}u = \Delta u$$

Review: Divergence, Gradient, and Curl



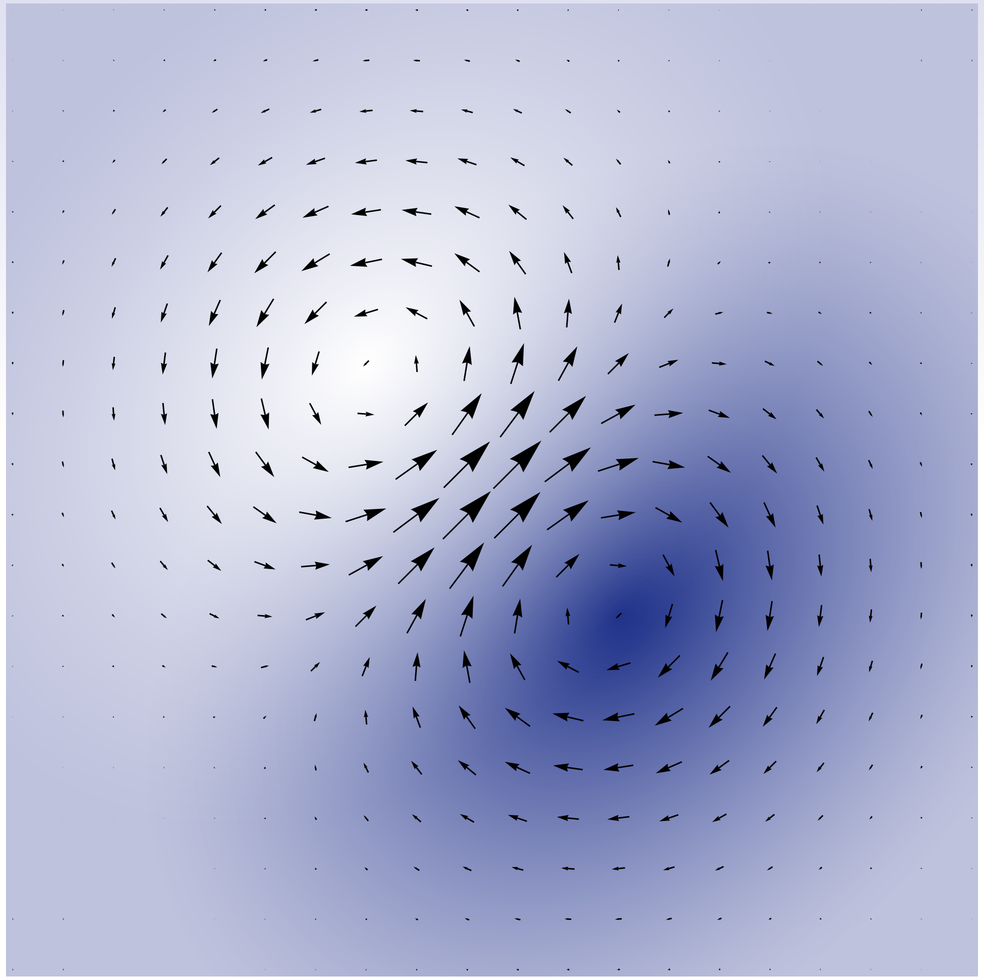
ϕ

grad ϕ



X

div X



Y

curl Y

Review—Vector Derivatives in Coordinates

How do we express grad, div, and curl in coordinates? Consider:

- a scalar function $\phi : \mathbb{R}^3 \to \mathbb{R}$
- a vector field $X = (u, v, w)$
  - functions u, v, w: $\mathbb{R}^n \to \mathbb{R}$ giving components of the vector field at each point

grad

$$\nabla\phi = \begin{bmatrix} \partial\phi/\partial x \\ \partial\phi/\partial y \\ \partial\phi/\partial z \end{bmatrix}$$

div

$$\nabla \cdot X = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}$$

curl

$$\nabla \times \phi = \begin{bmatrix} \frac{\partial w}{\partial y} - \frac{\partial v}{\partial z} \\ \frac{\partial u}{\partial z} - \frac{\partial w}{\partial x} \\ \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \end{bmatrix}$$

Review: Laplacian

$u : \mathbb{R}^n \to \mathbb{R}$ (twice differentiable)

Coordinates.

$$\Delta u = \sum_{i=1}^{n} \frac{\partial^2}{\partial x_i^2} u$$

e.g., in 2D:

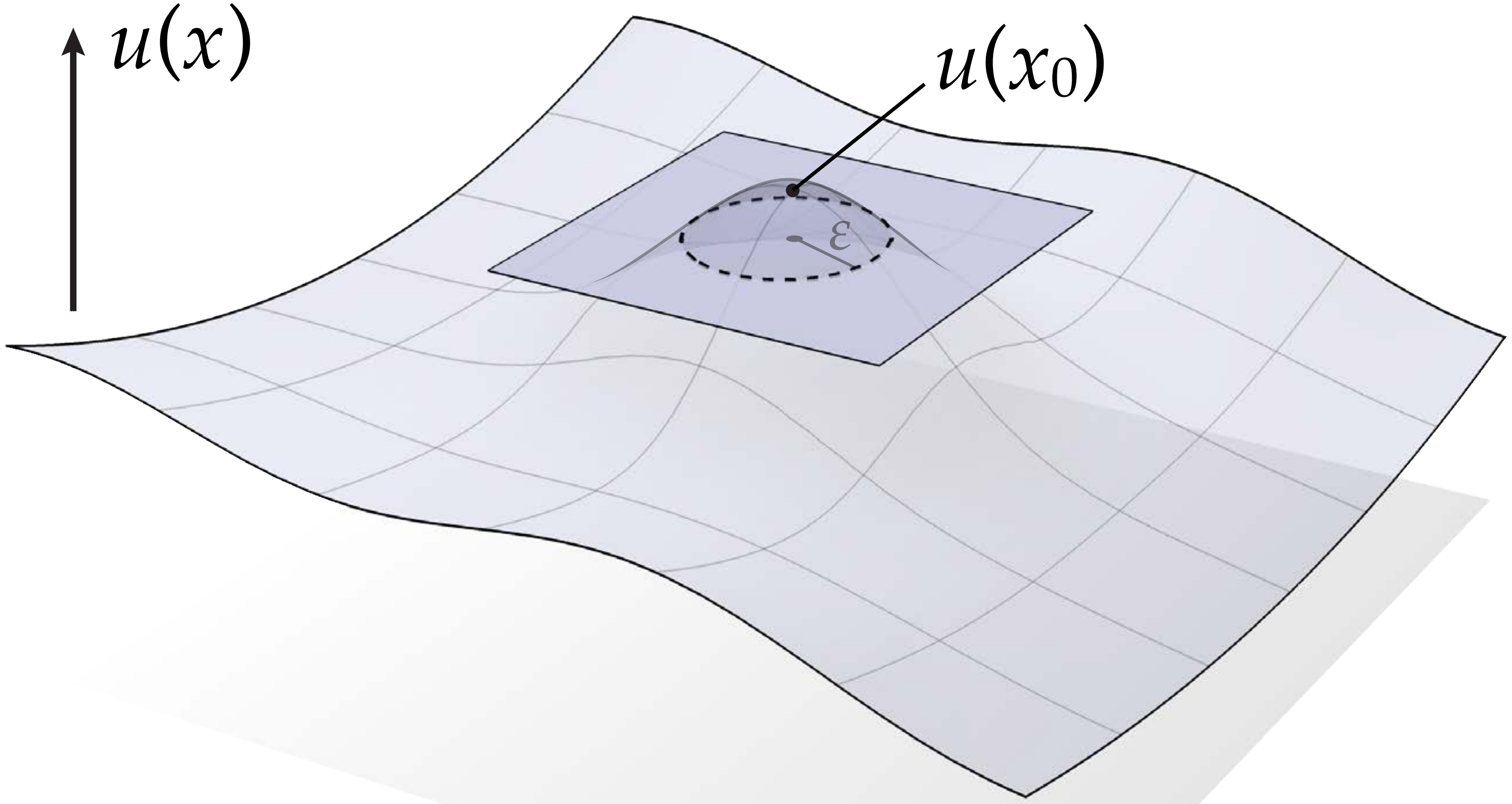
$$\Delta u(x, y) = \frac{\partial^2 u}{\partial x^2}(x, y) + \frac{\partial^2 u}{\partial y^2}(x, y)$$

Differential operators.

$$\Delta u = \nabla \cdot \nabla u = \operatorname{div} \circ \operatorname{grad} u$$

Laplacian Gives Deviation from Local Average

More intuitively, can think of the Laplacian of a function u as difference between value at a point x_0, and the average value over a small sphere (or ball) around x_0.



$$\Delta u(x_0) \propto \lim_{\varepsilon\to 0}\frac{1}{\varepsilon^2}\left(\underbrace{\frac{1}{|S_\varepsilon(x_0)|}}_{\text{sphere area}}\underbrace{\int_{S_\varepsilon(x_0)} u(x)\,dx}_{\text{integral over sphere}} - \underbrace{u(x_0)}_{\text{value at center}}\right)$$

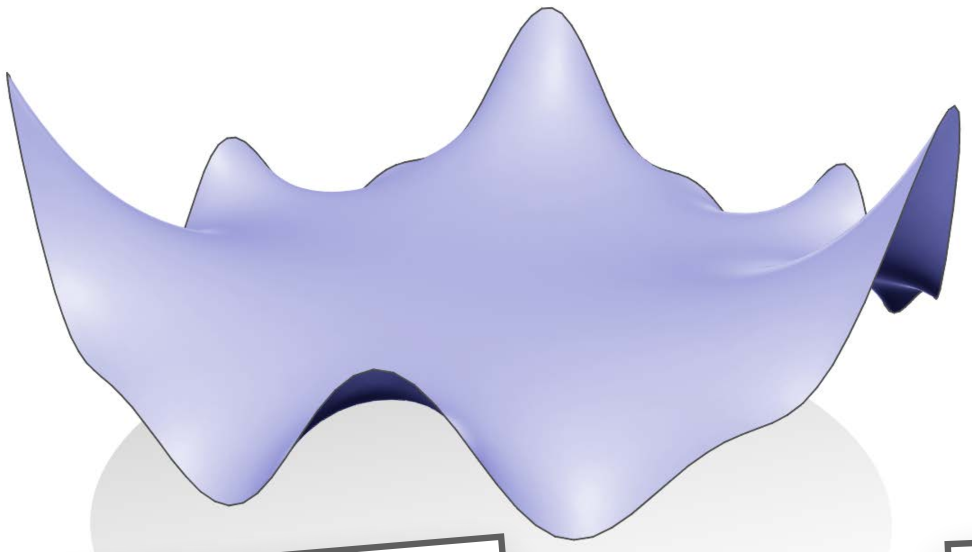
Elliptic / Parabolic / Hyperbolic PDEs

In general: check *symbol* of differential operator

Three basic types of behavior for PDEs, encapsulated by three model equations:

Laplace equation

(elliptic)



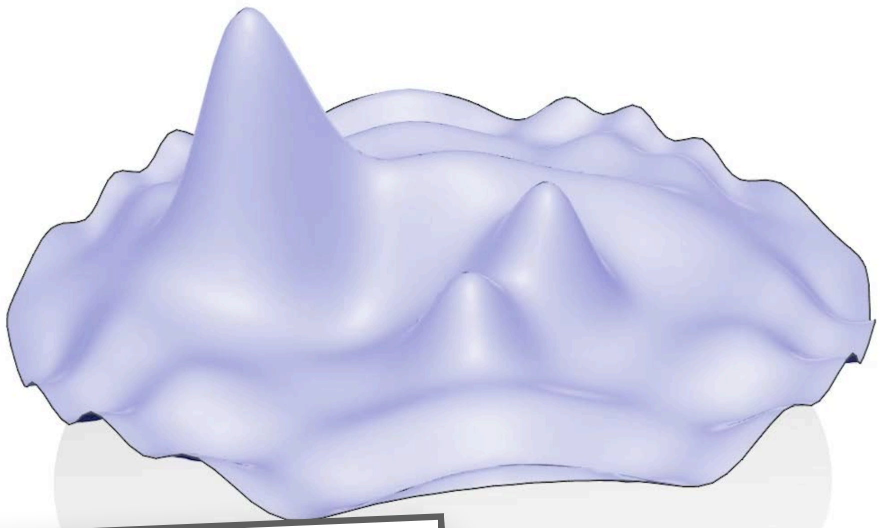
"time-independent"

$$\Delta u = 0$$

● EASIER

heat equation

(parabolic)



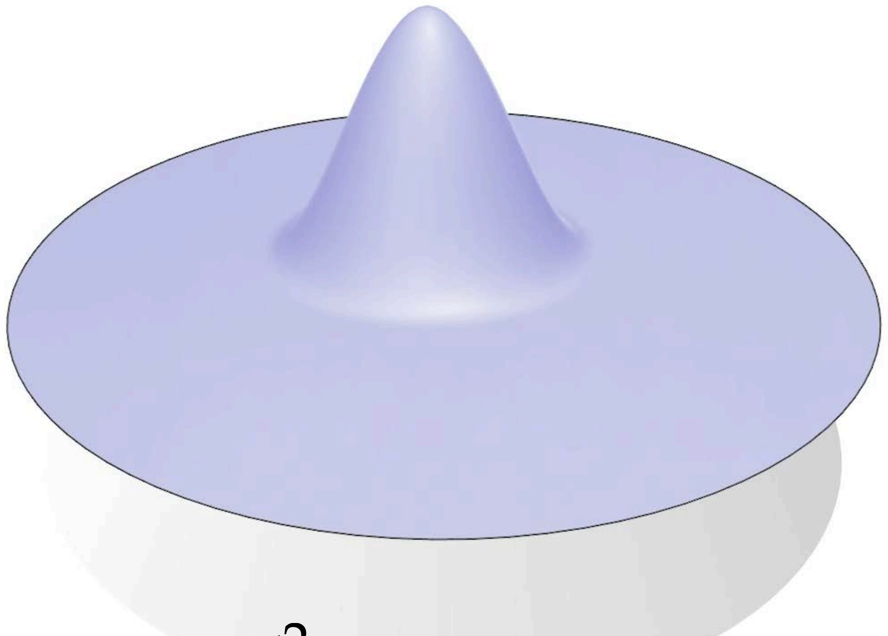
"time-dependent"

$$\frac{d}{dt} u = \Delta u$$

■ MORE DIFFICULT

wave equation

(hyperbolic)

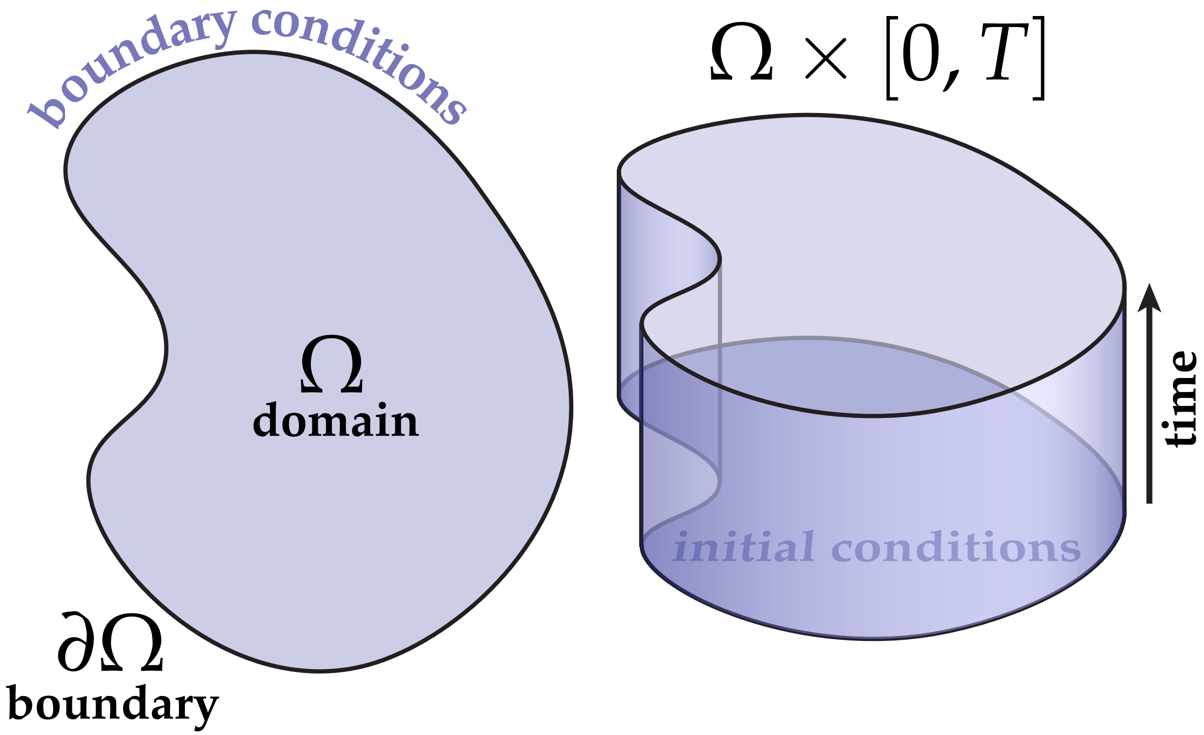


$$\frac{d^2}{dt^2} u = \Delta u$$

◆ MOST DIFFICULT

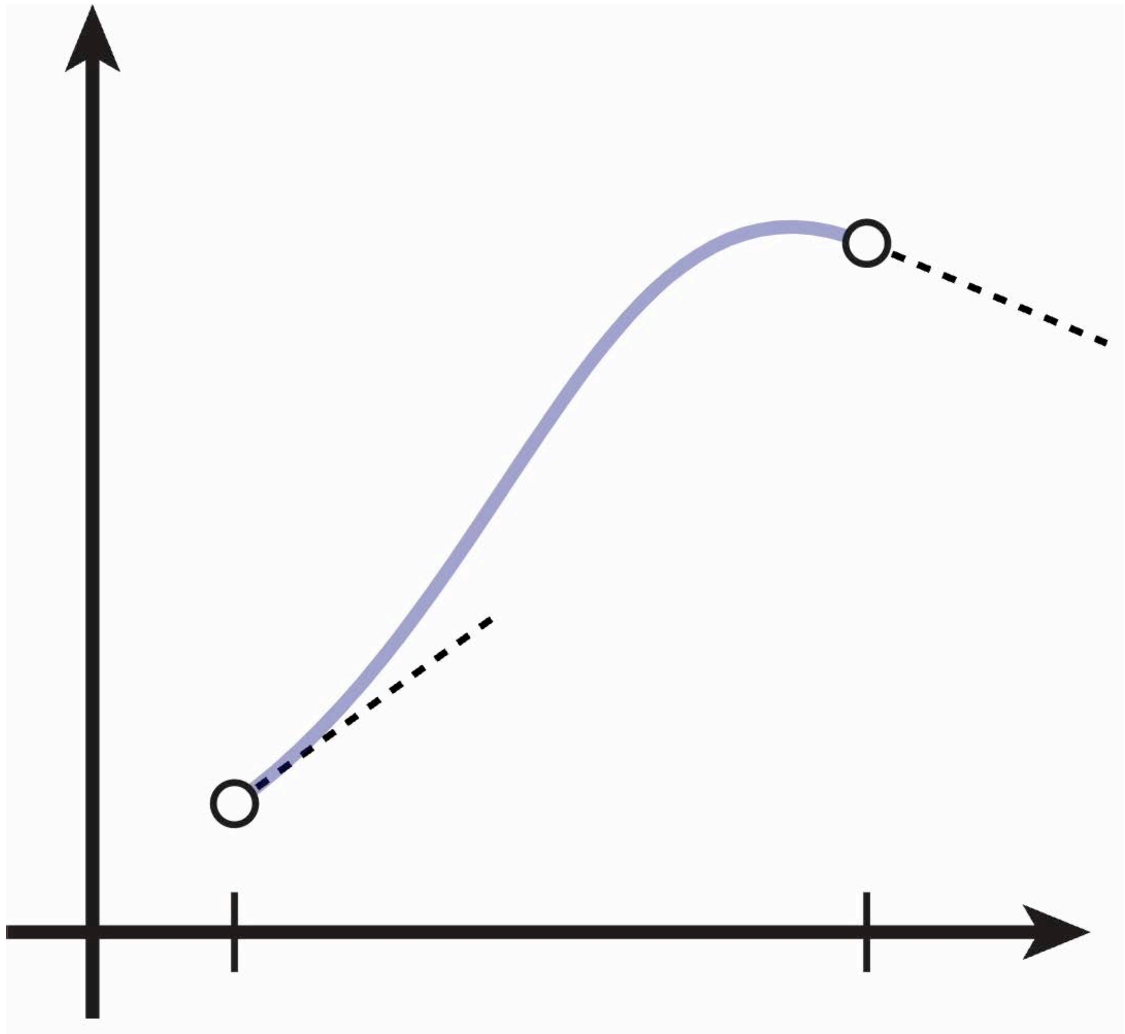
Initial & Boundary Conditions

- **Initial & boundary conditions** essentially define which instance of the problem we're talking about
- **Initial conditions** *(time-dependent problems only)*
  - E.g., what does the weather look like today, so we can predict what it will look like tomorrow?
- **Boundary conditions** *(domains with boundary only!)*
  - E.g., if we can measure the temperature only on the exterior of a body, what might we guess the temperature distribution looks like inside?



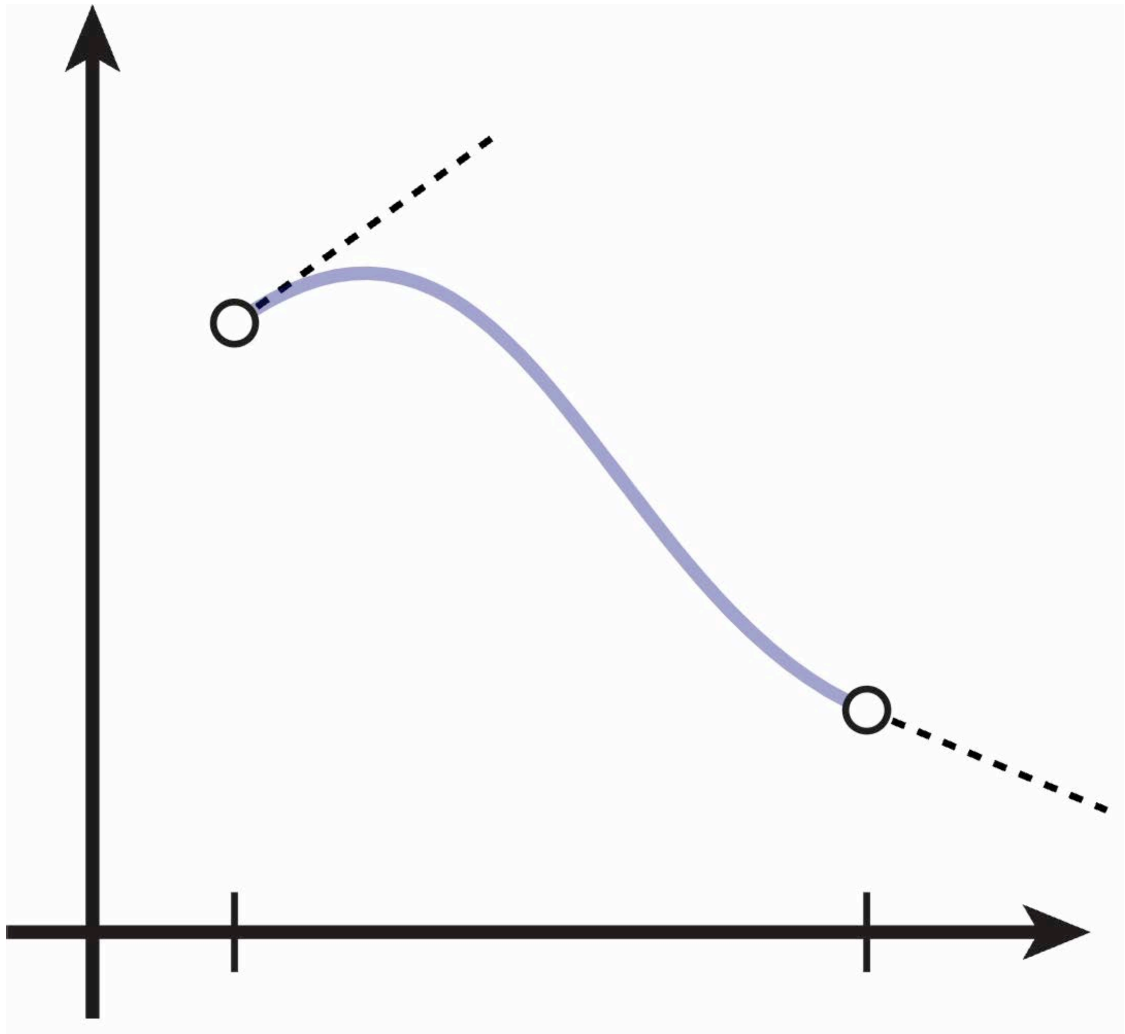
Dirichlet vs. Neumann Boundary Conditions

Dirichlet



interpolate given *values*

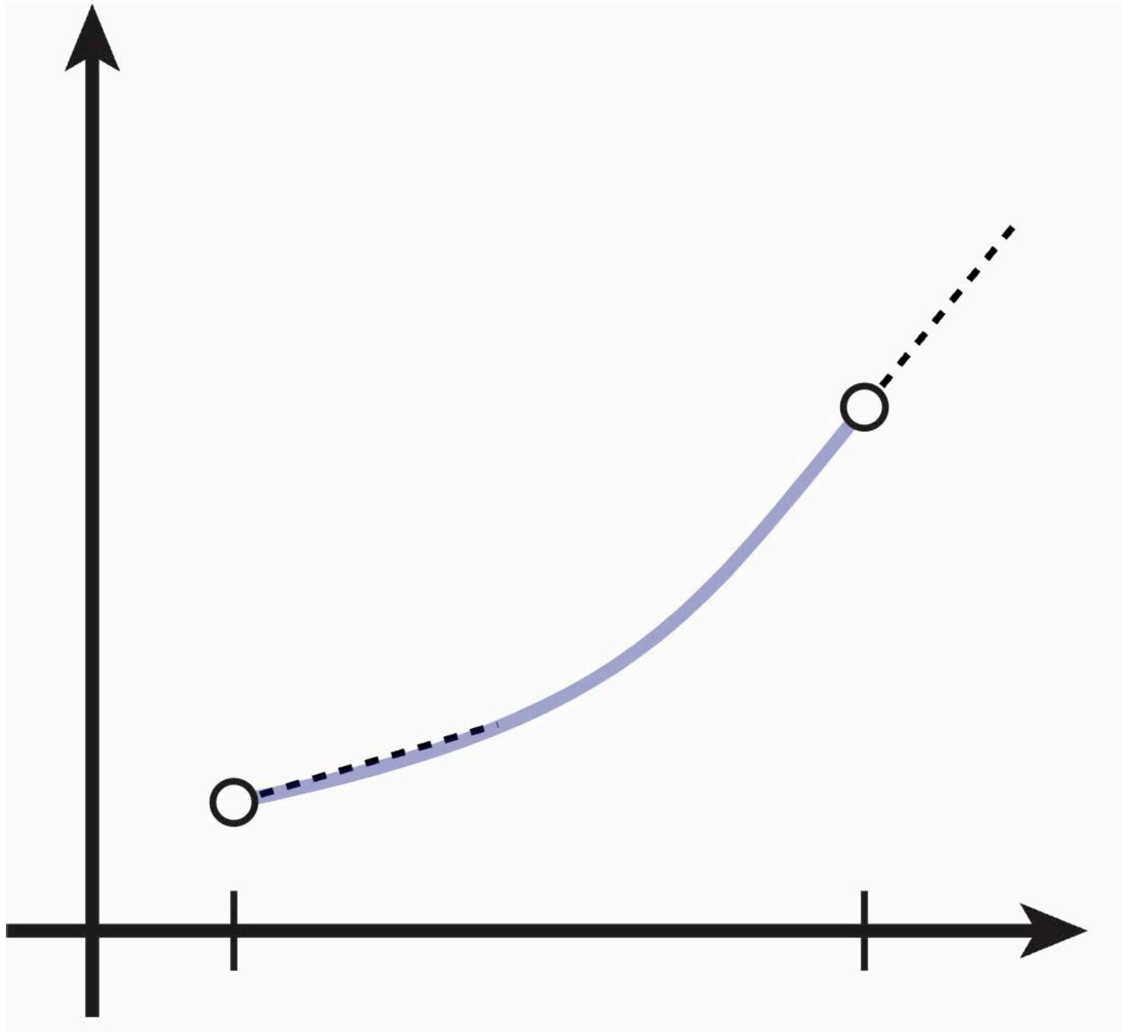
Neumann



interpolate given *derivatives*

Dirichlet vs. Neumann Boundary Conditions

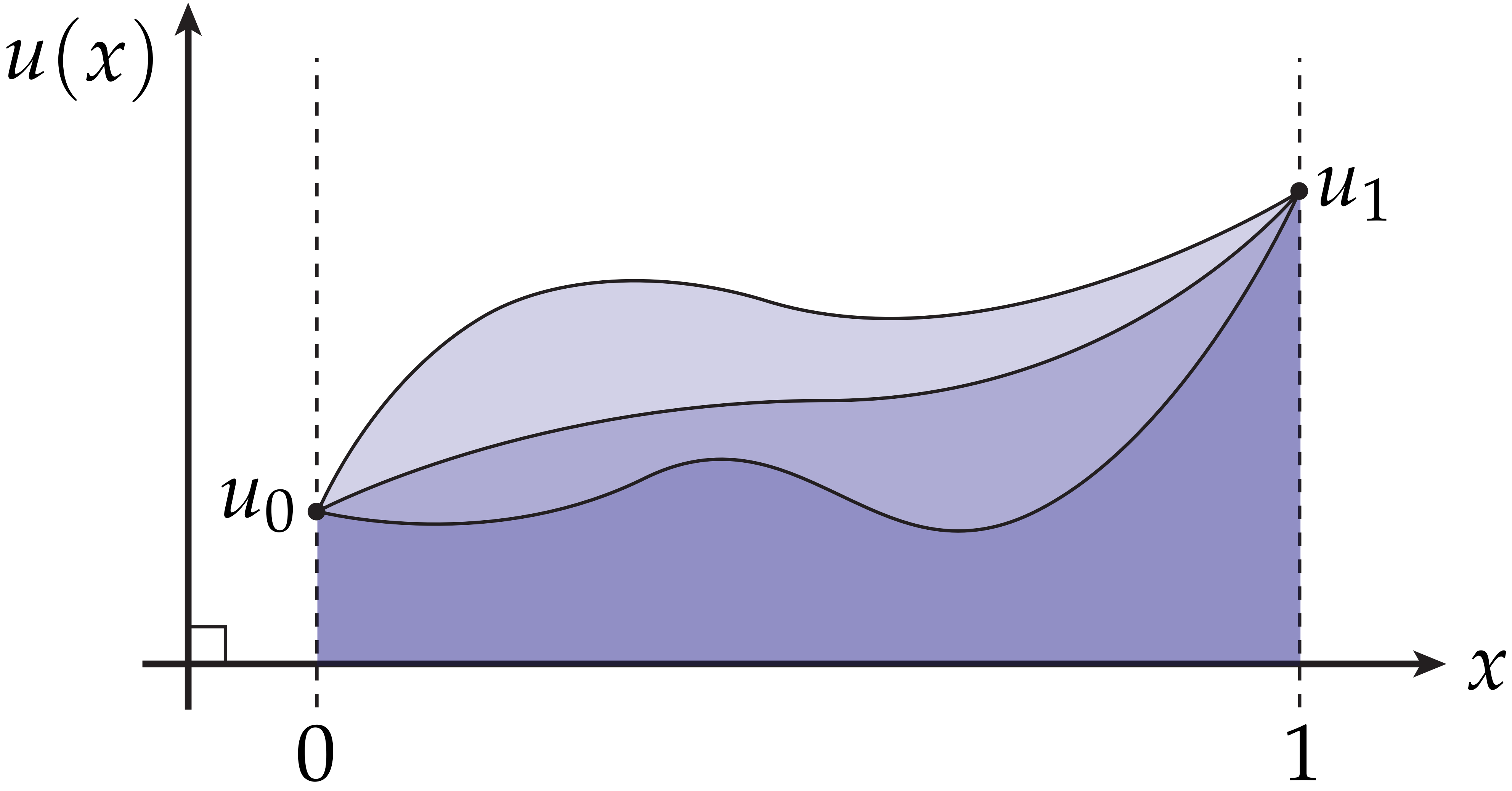
Mixed



fixed values *or* derivatives

Dirichlet Boundary Conditions

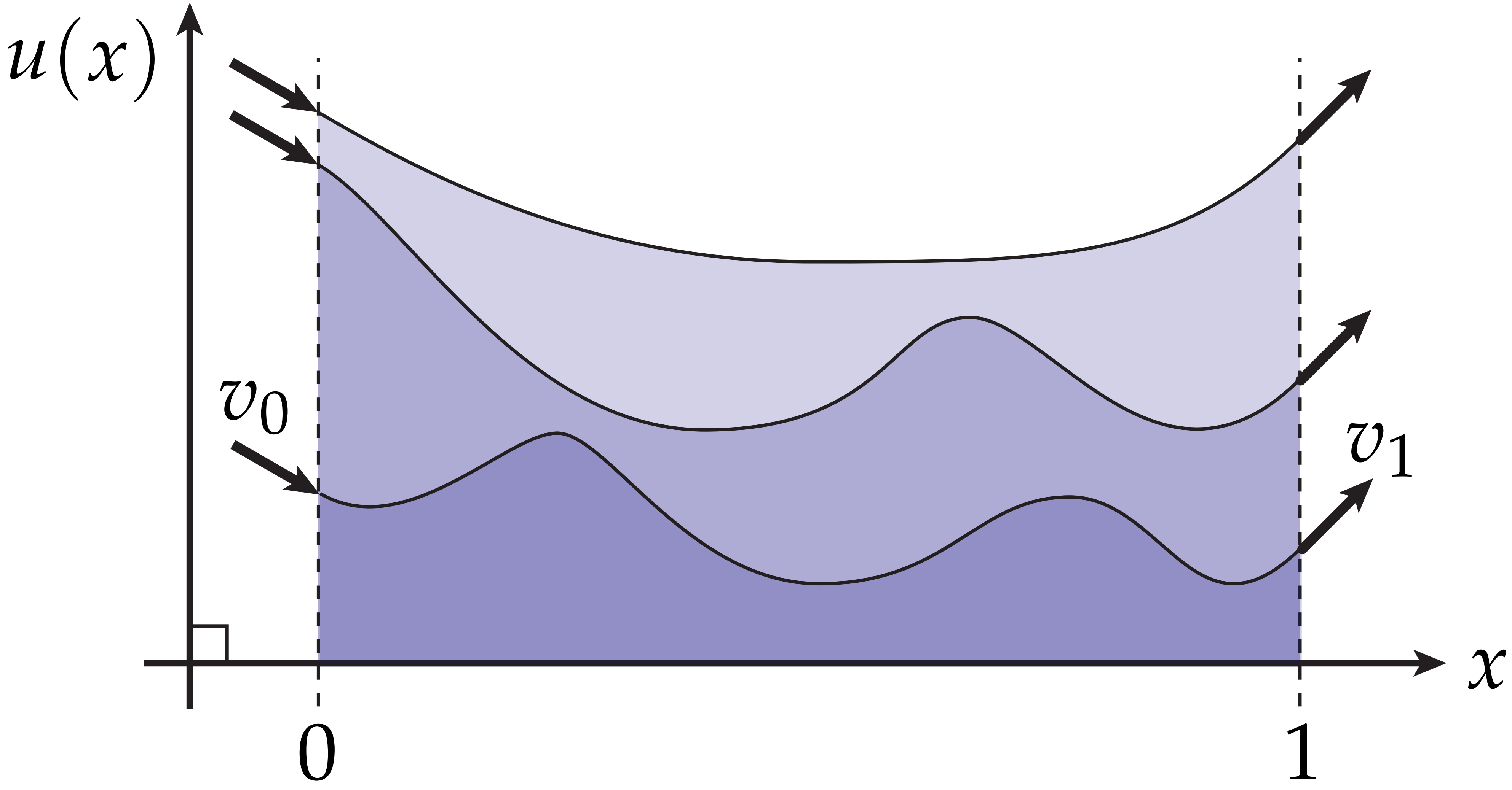
On interval [0,1], many possible functions w/ values u_0, u_1 at endpoints:



Key idea: “Dirichlet” just means boundary <u>values</u> are fixed.

Dirichlet Boundary Conditions

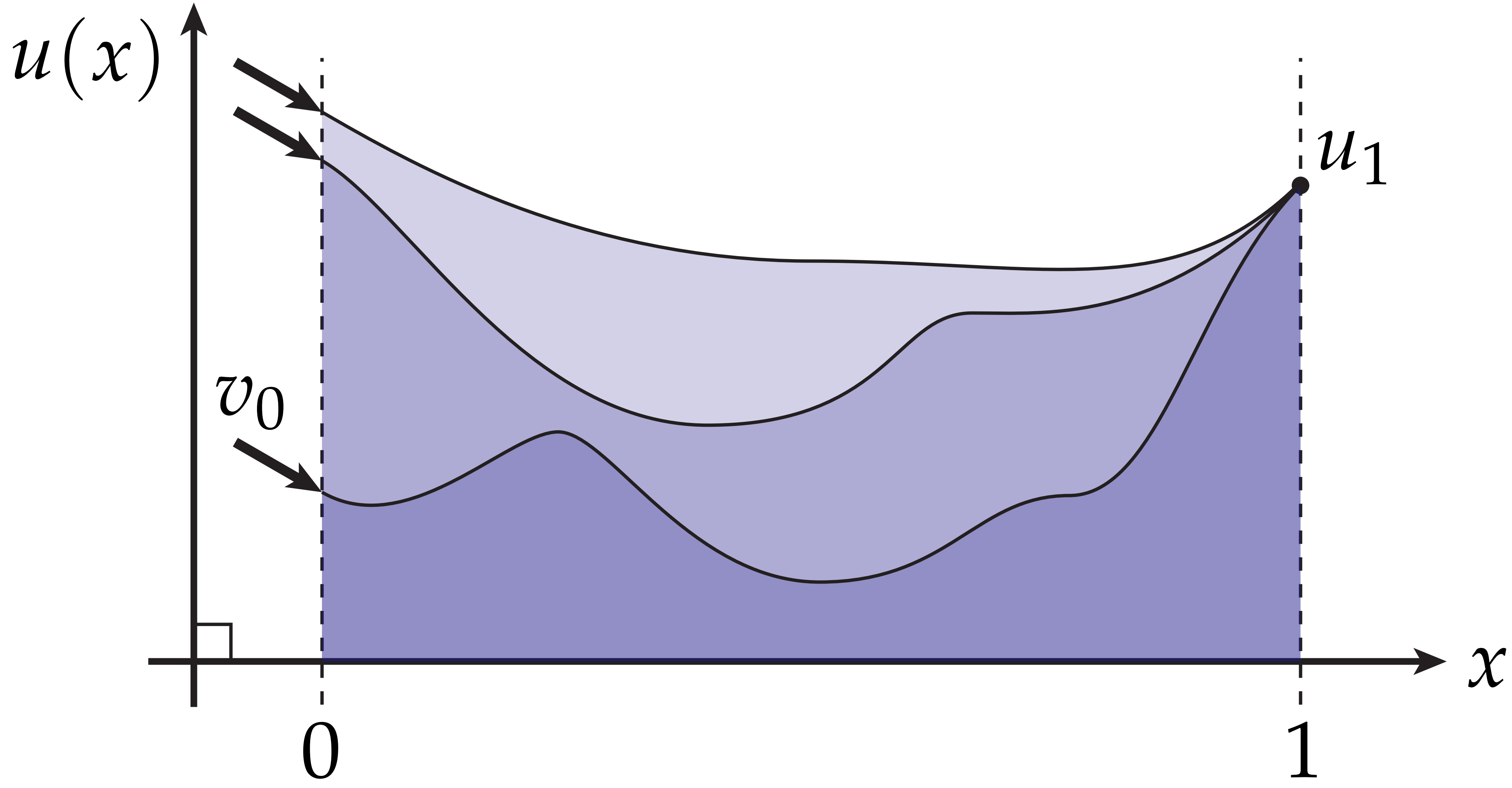
Likewise many possible functions w/ slope v_0, v_1 at endpoints:



Key idea: “Neumann” just means boundary <u>derivatives</u> are fixed.

Mixed Dirichlet & Neumann

Can also prescribe some values, some derivatives:



But what if we also have conditions on the interior?

Laplace w/ Dirichlet Boundary Conditions (1D)

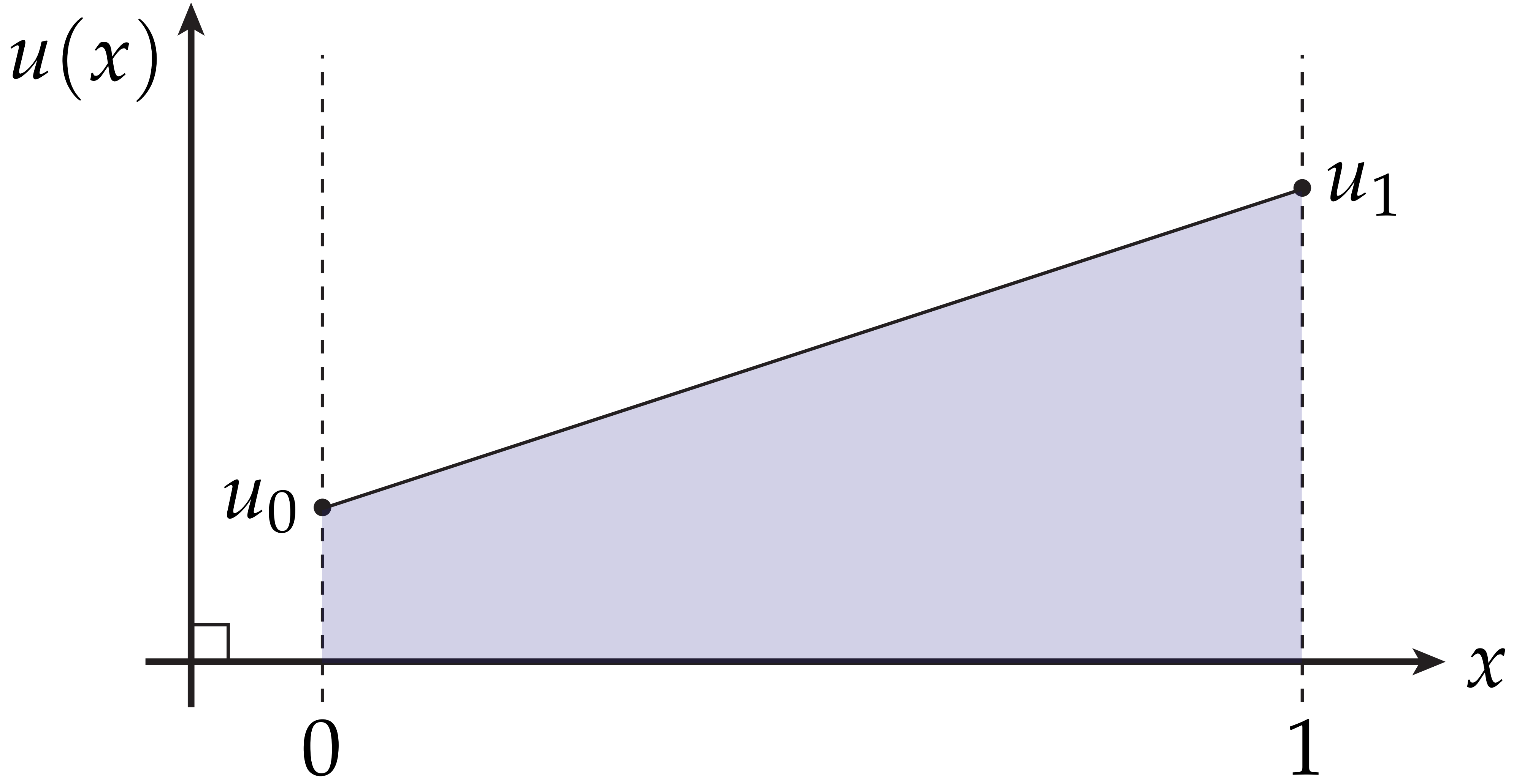
For a 1D Laplace equation, can we always satisfy Dirichlet conditions?

1D Laplace:

$$\frac{\partial^2 u}{\partial^2 x} = 0$$

Solutions:

$$u(x) = ax + b$$



Yes: a line can interpolate any two points.

Laplace w/ Neumann Boundary Conditions (1D)

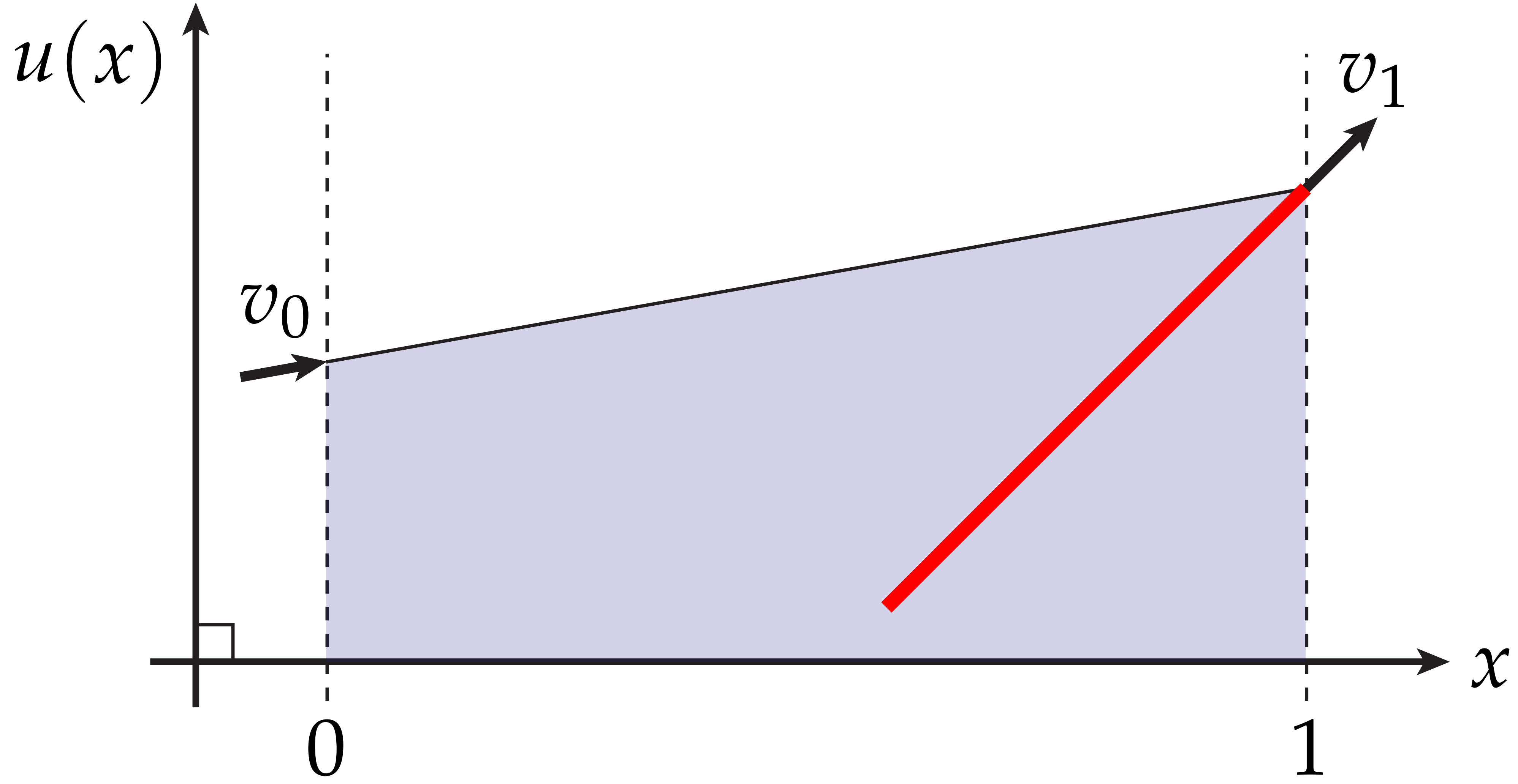
What about Neumann—can we prescribe the *derivative* at both ends?

1D Laplace:

$$\frac{\partial^2 u}{\partial^2 x} = 0$$

Solutions:

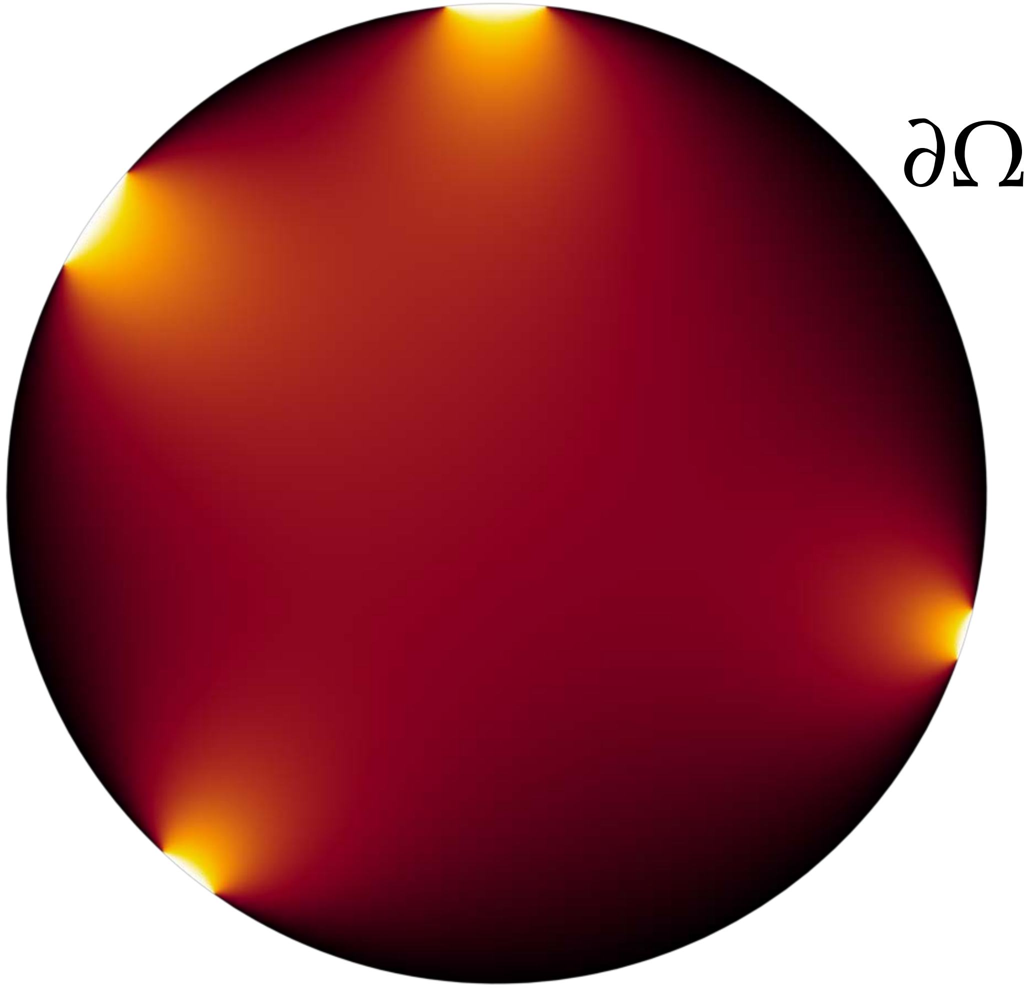
$$u(x) = ax + b$$



No! A line can have only one slope.

Laplace w/ Dirichlet Boundary Conditions (2D)

- Let's now consider a Laplace equation in 2D
- Can we always satisfy Dirichlet boundary conditions?
- Yes*: Laplace is steady-state solution to heat flow—just let it run for a long time…
  - Dirichlet data is "heat" along boundary



$$\begin{aligned} \Delta u &= 0 \quad \text{on} \quad \Omega \\ u &= g \quad \text{on} \quad \partial\Omega \end{aligned}$$

*Subject to very mild/reasonable conditions on boundary geometry, boundary data

Laplace w/ Neumann Boundary Conditions (2D)

- Suppose instead we prescribe the *normal derivative* along the boundary
- Can we always find a solution to the Laplace equation?
- Well, consider the divergence theorem—*"what goes in, must come out!"*

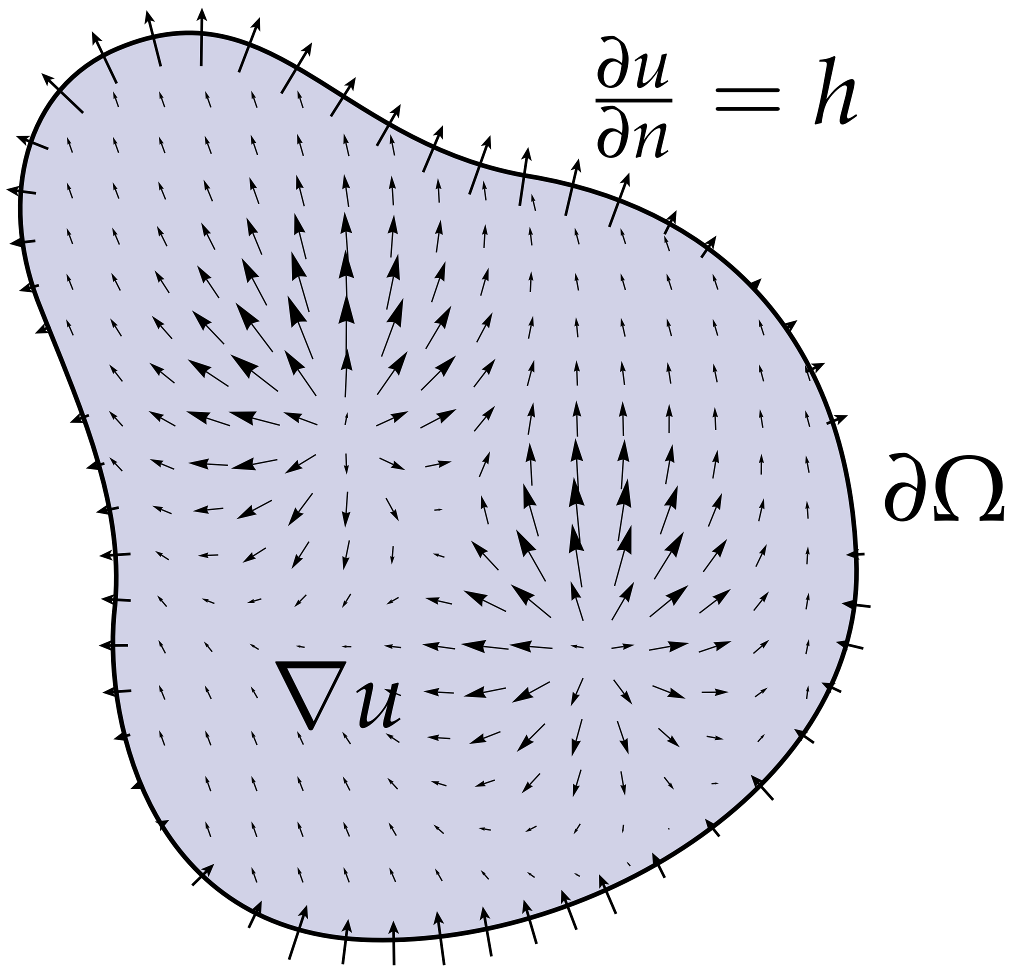
$$\int_\Omega 0 \, dA \overset{!}{=} \int_\Omega \Delta u \, dA = \int_\Omega \nabla \cdot \nabla u \, dA = \int_{\partial\Omega} \underbrace{n \cdot \nabla u}_{\partial u / \partial n} \, dA$$

- Can only solve if Neumann data h integrates to zero over the boundary

Important: in general, a PDE <u>may not have solutions</u> for given boundary conditions (*"ill-posed"*)

$$\begin{aligned} \Delta u &= 0 \quad \text{on} \quad \Omega \\ \frac{\partial u}{\partial n} &= h \quad \text{on} \quad \partial\Omega \end{aligned}$$

$$(h : \partial\Omega \to \mathbb{R})$$



Convection-Diffusion Equation

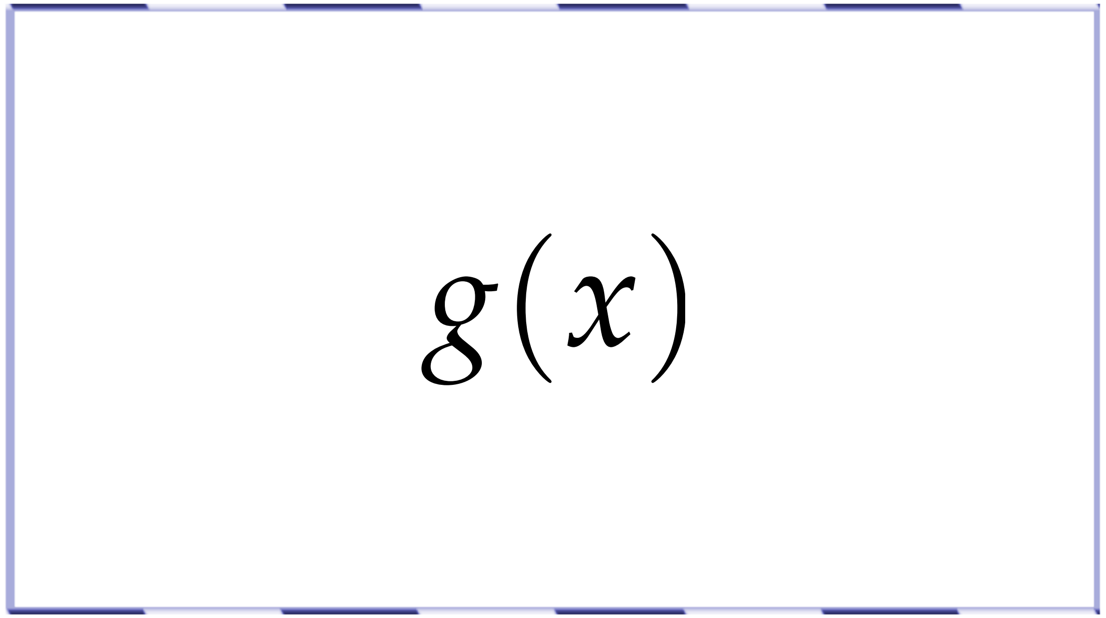
$$\begin{aligned} \nabla \cdot (\alpha \nabla u) + \vec{\omega} \cdot \nabla u - \lambda u &= f \text{ on } \Omega \\ \underset{\text{TEMPERATURE}}{\boxed{u}} &= g \text{ on } \partial\Omega \end{aligned}$$

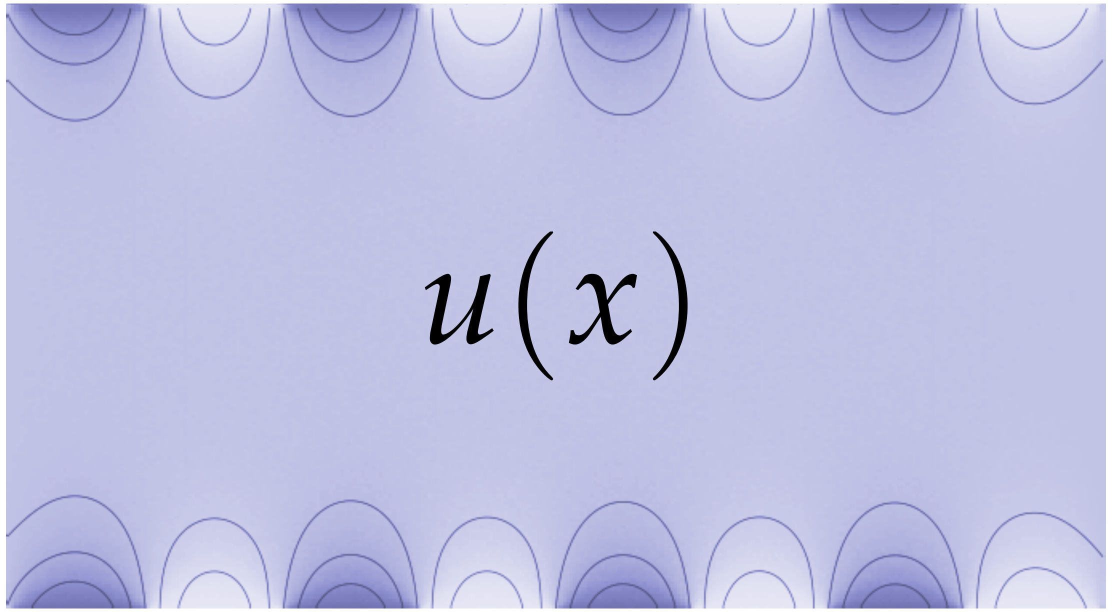
The **convection-diffusion equation** provides link between PDEs and **diffusion processes** (SDEs).

Convection-Diffusion Equation

$$\begin{aligned} \nabla \cdot (\alpha \nabla u) + \vec{\omega} \cdot \nabla u - \lambda u &= 0 \quad \text{on } \Omega \\ u &= \boxed{g} \quad \text{on } \partial\Omega \end{aligned}$$

BOUNDARY VALUE

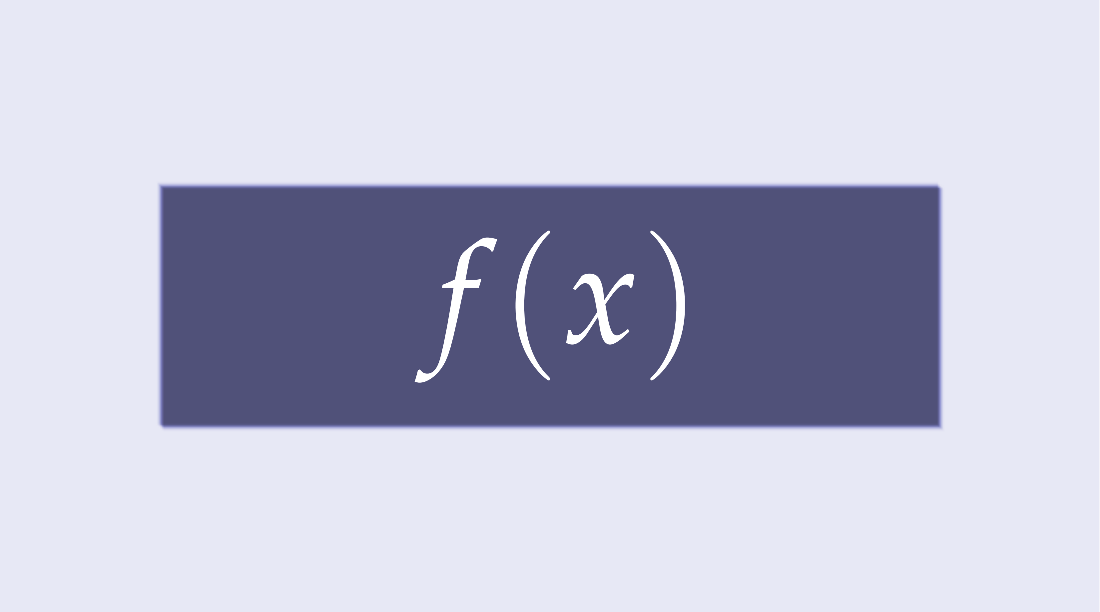


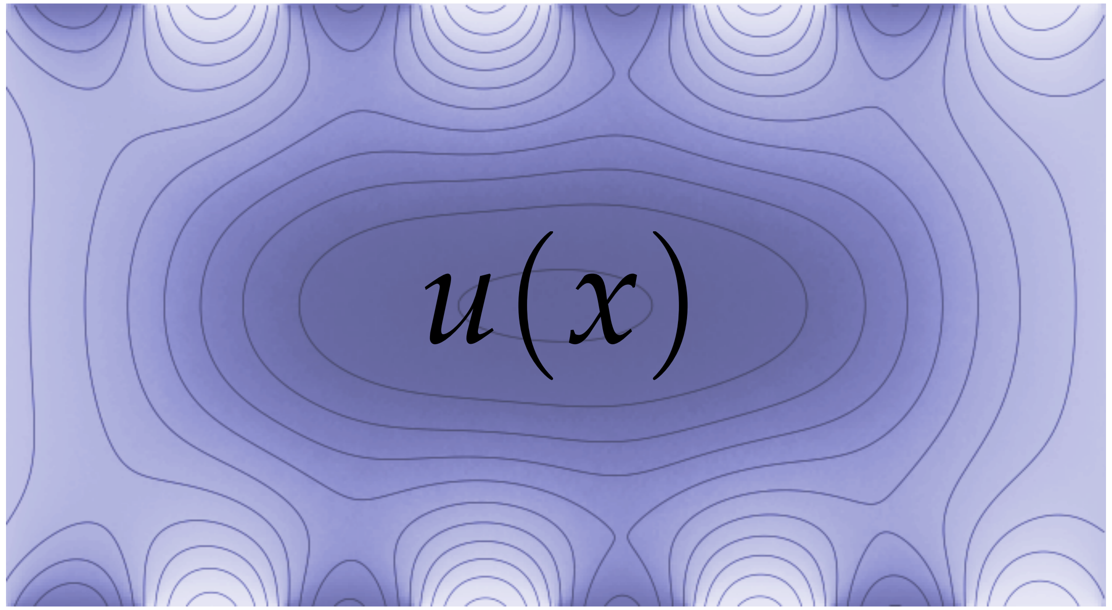


Intuition: temperature along boundary is fixed.

Convection-Diffusion Equation

$$\begin{aligned} \nabla \cdot (\alpha \nabla u) + \vec{\omega} \cdot \nabla u - \lambda u &= \overset{\textsc{Source}}{f} \text{ on } \Omega \\ u &= g \text{ on } \partial\Omega \end{aligned}$$





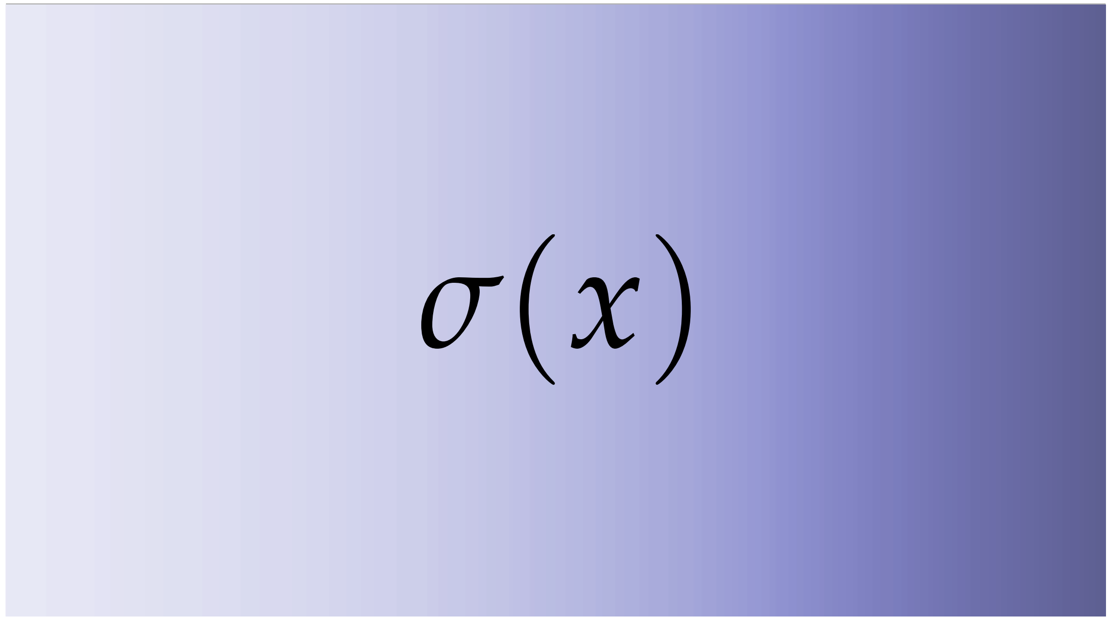
Intuition: adds additional “background temperature.”

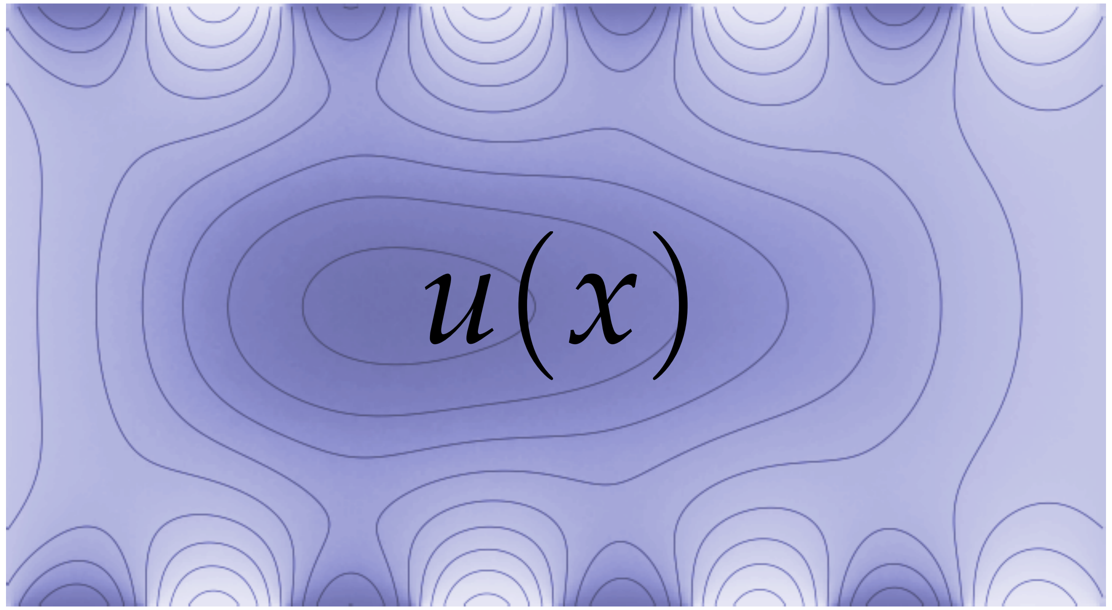
Convection-Diffusion Equation

ABSORPTION

$$\nabla \cdot (\alpha \nabla u) + \vec{\omega} \cdot \nabla u - \boxed{\lambda u} = f \quad \text{on } \Omega$$

$$u = g \quad \text{on } \partial\Omega$$

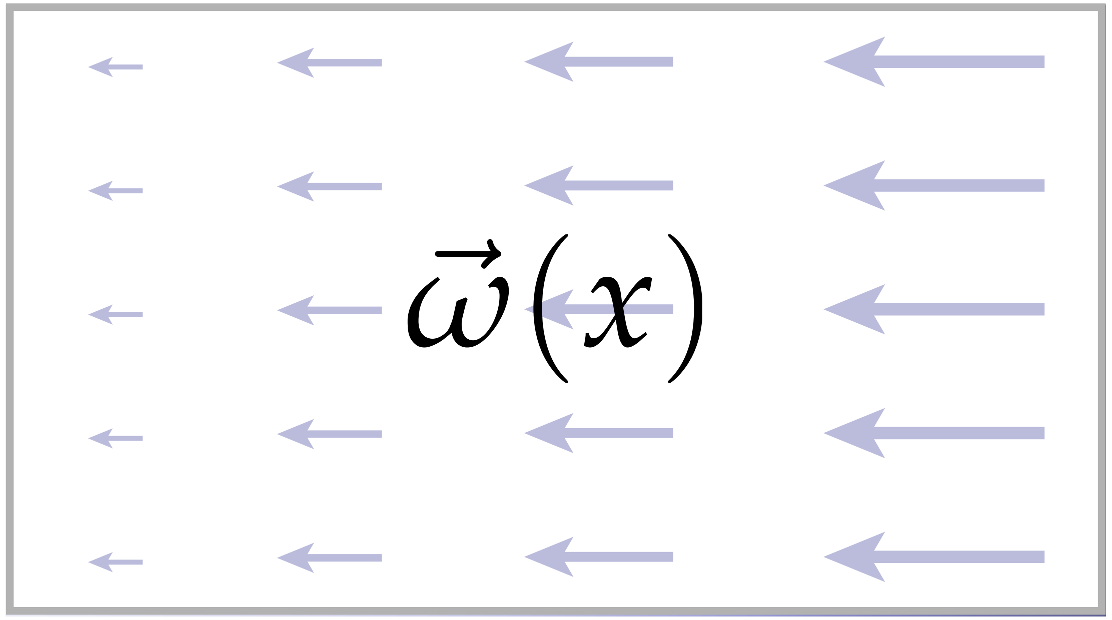


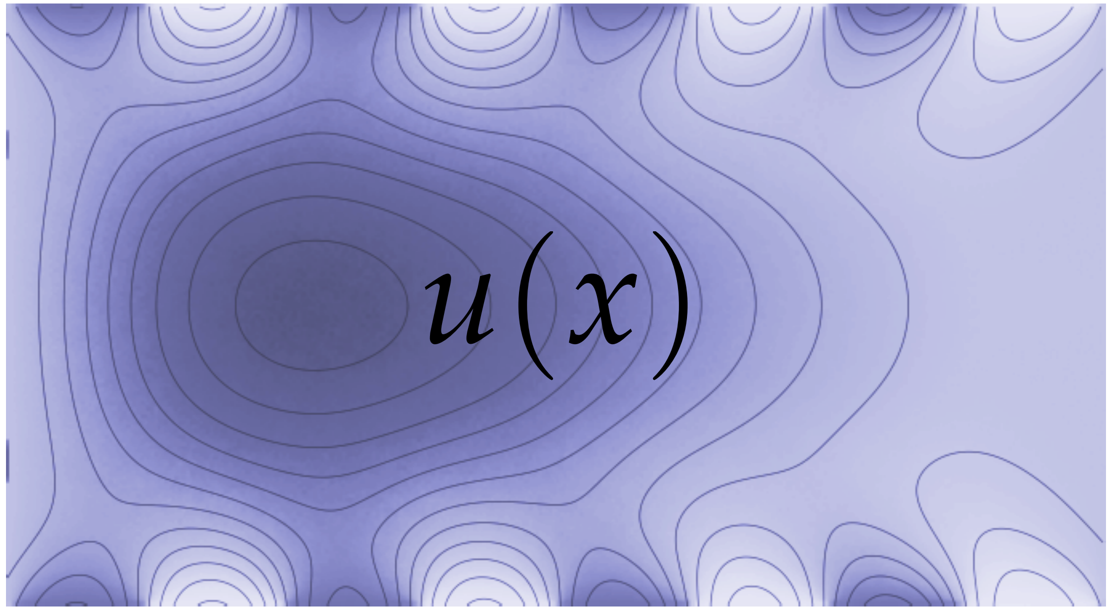


Intuition: “cooling” due to absorption into background medium.

Convection-Diffusion Equation

$$\nabla \cdot (\alpha \nabla u) + \overset{\text{DRIFT}}{\boxed{\vec{\omega} \cdot \nabla u}} - \lambda u = f \text{ on } \Omega$$
$$u = g \text{ on } \partial\Omega$$



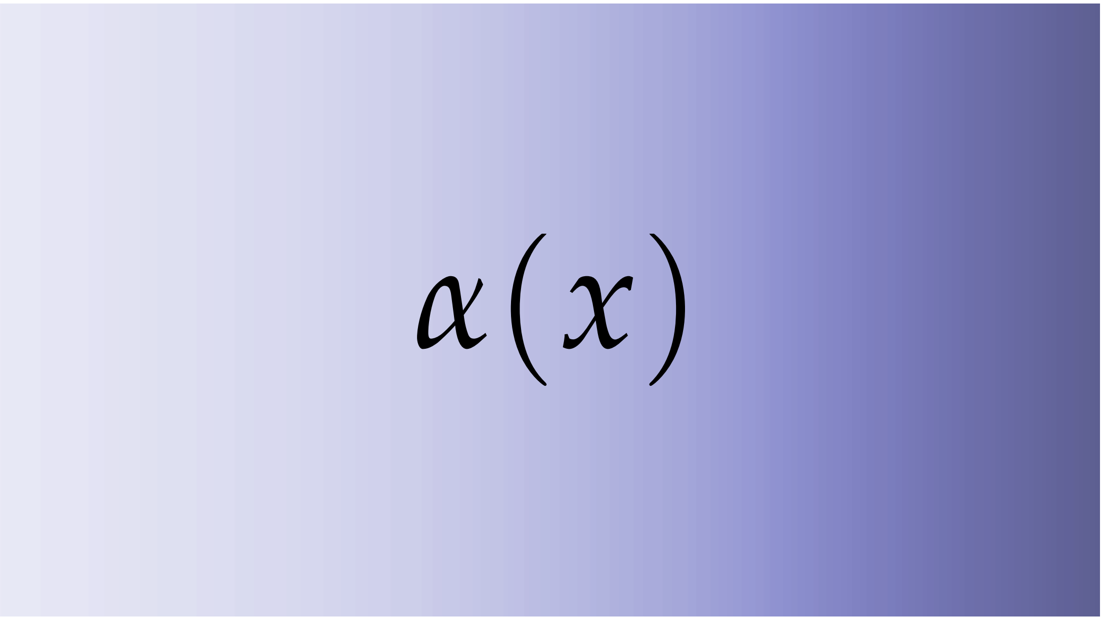


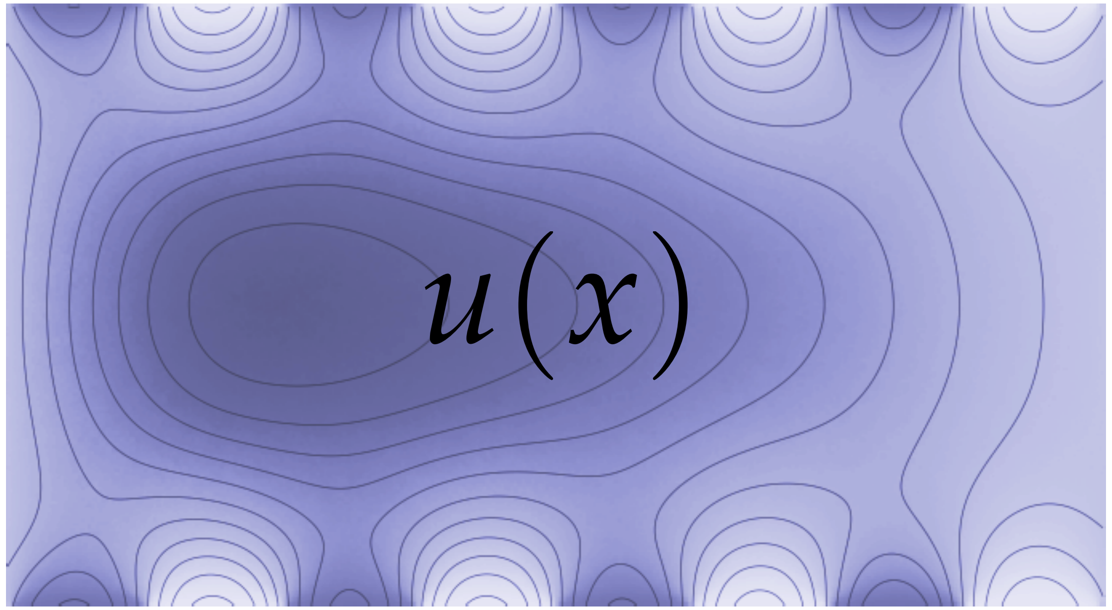
Intuition: heat is dragged along with a flowing river.

Convection-Diffusion Equation

DIFFUSION

$$\begin{aligned} \nabla \cdot (\alpha \nabla u) + \vec{\omega} \cdot \nabla u - \lambda u &= f \quad \text{on } \Omega \\ u &= g \quad \text{on } \partial\Omega \end{aligned}$$





Intuition: how fast does heat "spread out"?

Convection-Diffusion Equation—Coefficients

In general, coefficient functions α, ω, λ can vary over space:

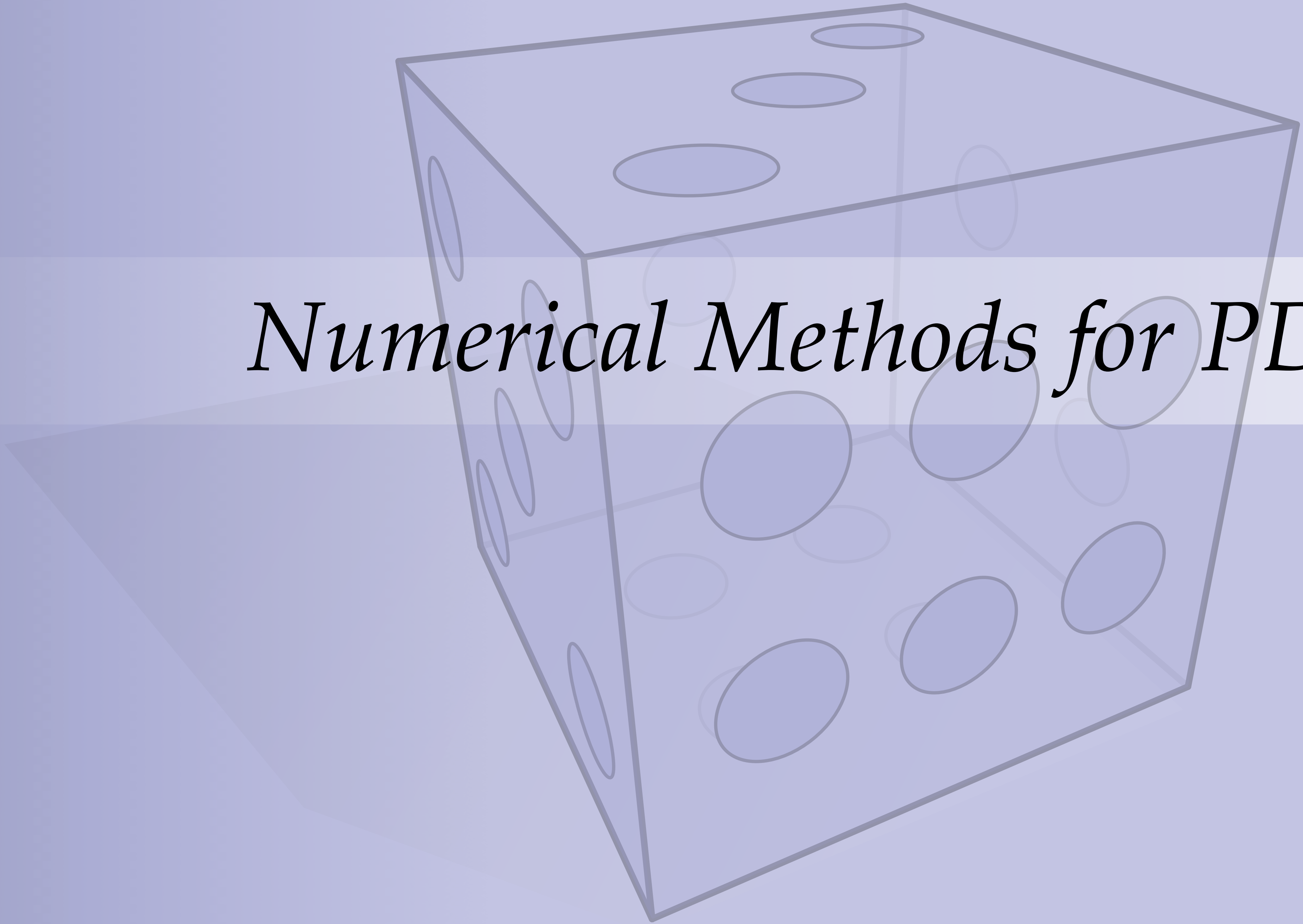
$$\begin{aligned}\underbrace{\nabla\cdot(\alpha(x)\nabla u)}_{\textsc{Diffusion}} + \underbrace{\vec{\omega}(x)\cdot\nabla u}_{\textsc{Drift}} - \overbrace{\lambda(x)u}^{\textsc{Absorption}} &= \overbrace{f}^{\textsc{Source}} \text{ on } \Omega\\ u &= \underbrace{g}_{\textsc{Boundary}} \text{ on } \partial\Omega\end{aligned}$$

Convection-Diffusion Equation—Coefficients

In general, coefficient functions α, ω, λ can vary over space:

$$\begin{aligned} \underbrace{\Delta u}_{\textsc{Diffusion}} + \underbrace{\vec{\omega}\cdot\nabla u}_{\textsc{Drift}} - \overbrace{\lambda u}^{\textsc{Absorption}} &= \overbrace{f}^{\textsc{Source}} \quad \text{on } \Omega \\ u &= \underbrace{g}_{\textsc{Boundary}} \quad \text{on } \partial\Omega \end{aligned}$$

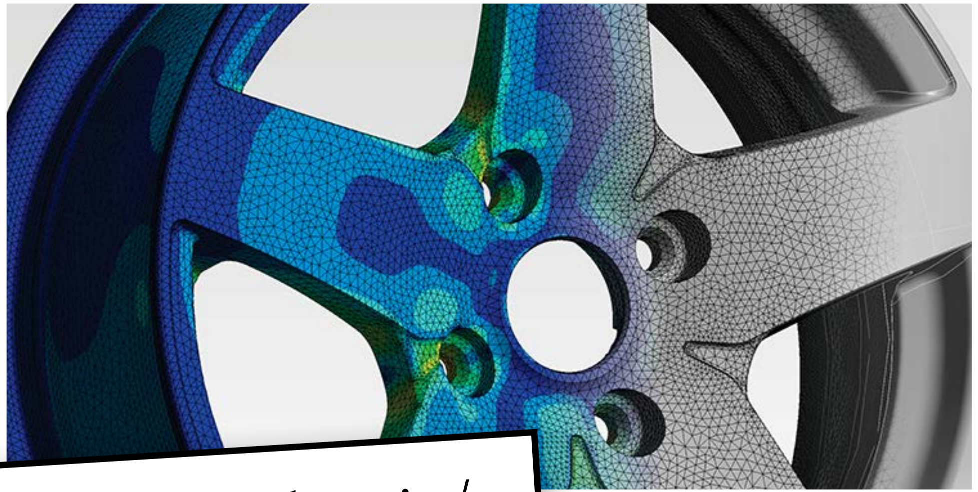
Notation: will often drop arguments for brevity.



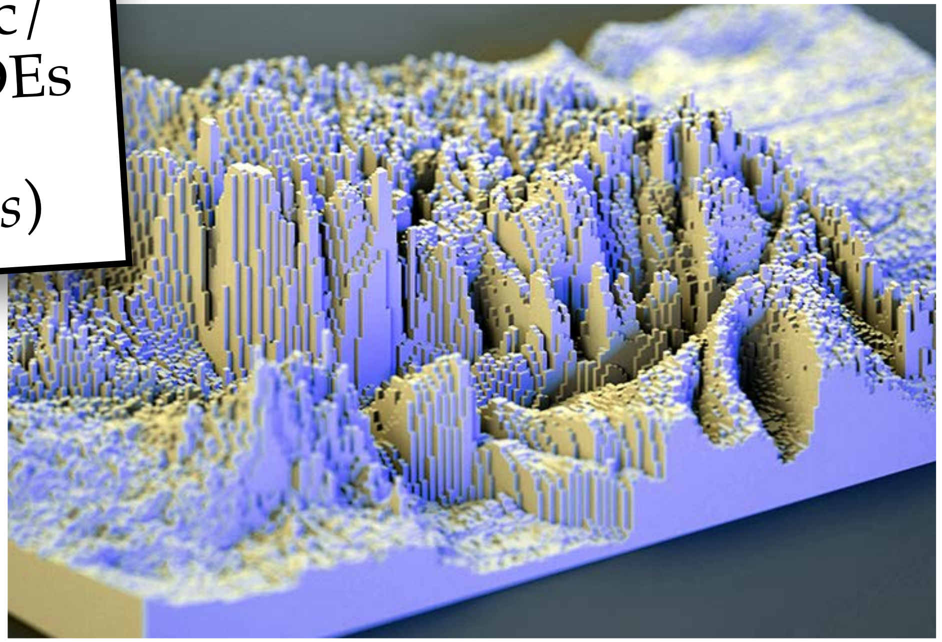
Numerical Methods for PDEs

Numerical Methods for PDEs—Overview

- Numerical methods for ODEs were reasonably straightforward: start with some initial conditions and "march through time"; some choice of time stepping scheme
- Much bigger "zoo" of approaches for PDEs:
  - finite difference methods
  - finite element methods
  - finite volume methods
  - boundary integral methods
  - spectral methods
  - …
- **Common thread:** all use *finite-dimensional approximation*
  - e.g., finite system of equations, discrete set of unknowns

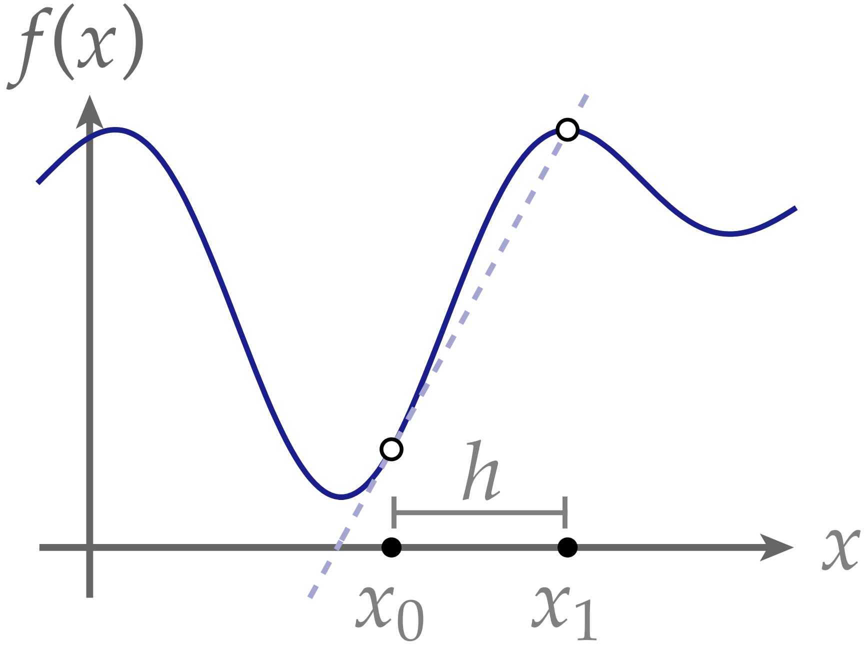


Later lecture: will use stochastic/Monte Carlo picture to solve PDEs without finite-dimensional approximation! (*walk on spheres*)



Finite Differences

- **Finite difference method** approximates differential equations via *"pointwise"* perspective
  - Imagine I have values of a function $f : \mathbb{R} \to \mathbb{R}$ only at a discrete set of sample points
  - *How can I approximate derivatives?*
- **Example.** Approximate 1st-order derivative via first couple terms of Taylor series.



sample points

$$f_i := f(x_i),\ x_1, x_2, \ldots \in \mathbb{R}$$

Taylor series

$$f(x+h) = \sum_{k=0}^{\infty} \frac{f^{(k)}(x)}{k!} h^k$$

$$f(x+h) = f(x) + f'(x)h + O(h^2)$$

$$\iff \frac{f(x+h) - f(x)}{h} = f'(x) + O(h^2)$$

"first-order accurate"

forward difference

$$f'(x_0) \approx \frac{f_1 - f_0}{h}$$

Finite Differences—Second Order

Example. Take "difference of differences" to approximate 2nd-order derivative:

$$\hat{f}'_0 := \frac{f_1 - f_0}{h} \qquad \hat{f}'_1 := \frac{f_2 - f_1}{h}$$

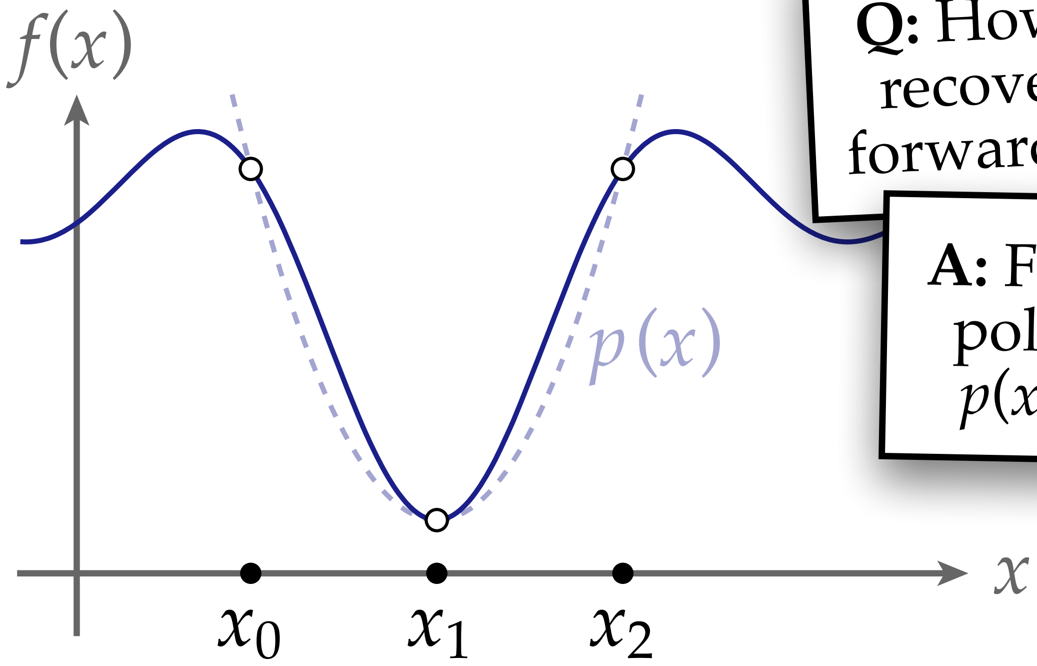
$$\hat{f}''_1 := \frac{\hat{f}'_1 - \hat{f}'_0}{h} = \frac{\frac{f_2 - f_1}{h} - \frac{f_1 - f_0}{h}}{h} = \frac{f_0 - 2f_1 + f_2}{h^2}$$

Q: Is this a reasonable approximation?

Feels like we're piling approximation on top of approximation…

Finite Differences—Polynomial Fitting

- Similar perspective: fit polynomial to sample points; derivatives of polynomial then approximate derivatives of original function.
- **Example.** Fit quadratic polynomial $p(x)$ to three points. Then $p'(x)$ gives *centered* difference approximation for 1st derivative of $f(x)$ at x_0; $p''(x)$ gives finite difference formula for 2nd derivative.



Q: How would we recover 1st-order forward difference?

A: Fit a linear polynomial $p(x) = ax+b$

sample points

$$x_0, x_1, x_2$$

$-h \quad 0 \quad +h$

sample values

$$f_i = f(x_i), \ \ i = 1,2,3$$

quadratic fit

$$p(x) = ax^2 + bx + c$$

linear system for a, b, c

$$p(x_i) = f_i, \ i = 1,2,3$$

$$\frac{f_0 x(x-h) + 2f_1(h-x)(h+x) + f_2 x(h+x)}{2h^2}$$

1st derivative *(centered)*

$$p'(x)|_{x=0} = \frac{f_2 - f_0}{2h}$$

O(h^2) error

2nd derivative

$$p''(x)|_{x=0} = \frac{f_0 - 2f_1 + f_2}{h^2}$$

Finite Differences - Laplacian

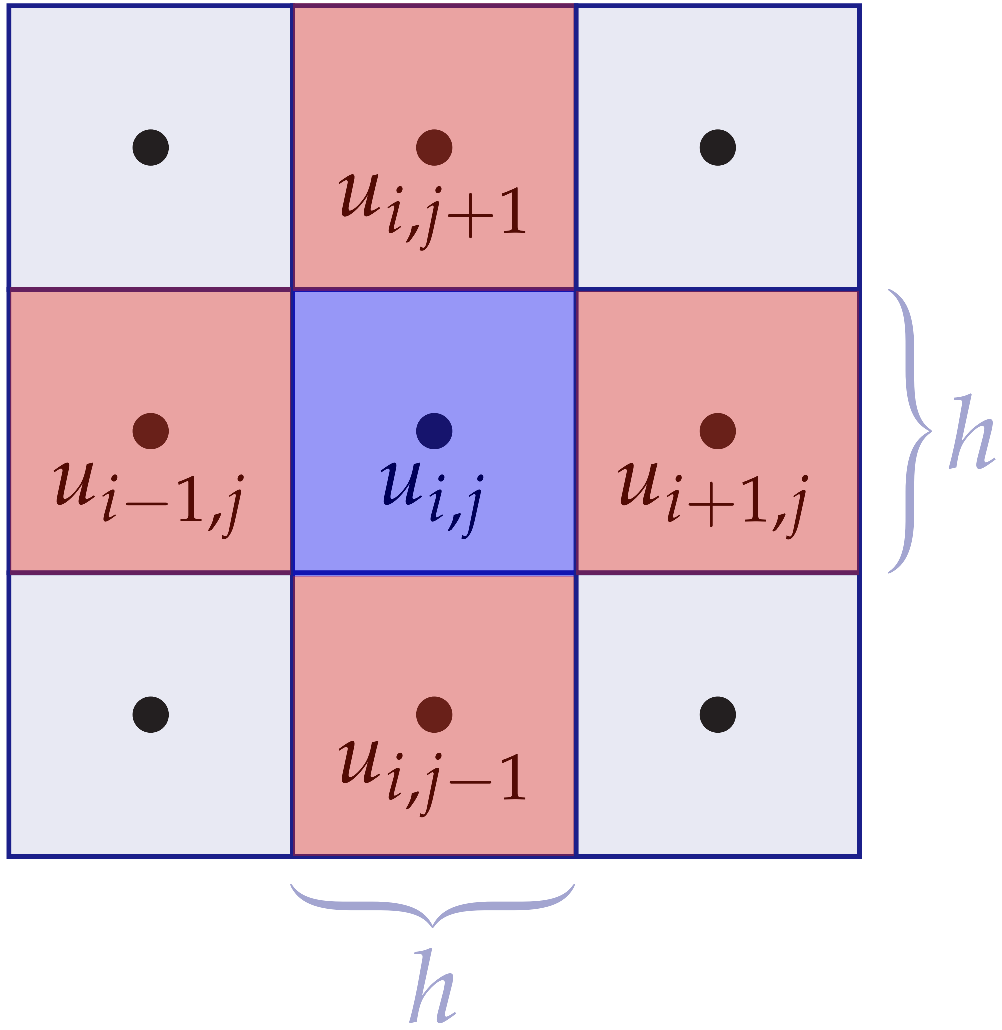
Notice: approximates deviation from local average!

- To approximate differential operators in multiple dimensions, can just apply 1D finite difference approximations along each coordinate direction
- **Example.** Laplacian in 2D

$$\Delta u = \frac{\partial^2}{\partial x^2}u + \frac{\partial^2}{\partial y^2}u$$

$$\frac{u_{i-1,j} - 2u_{i,j} + u_{i+1,j}}{h^2} \qquad \frac{u_{i,j-1} - 2u_{i,j} + u_{i,j+1}}{h^2}$$

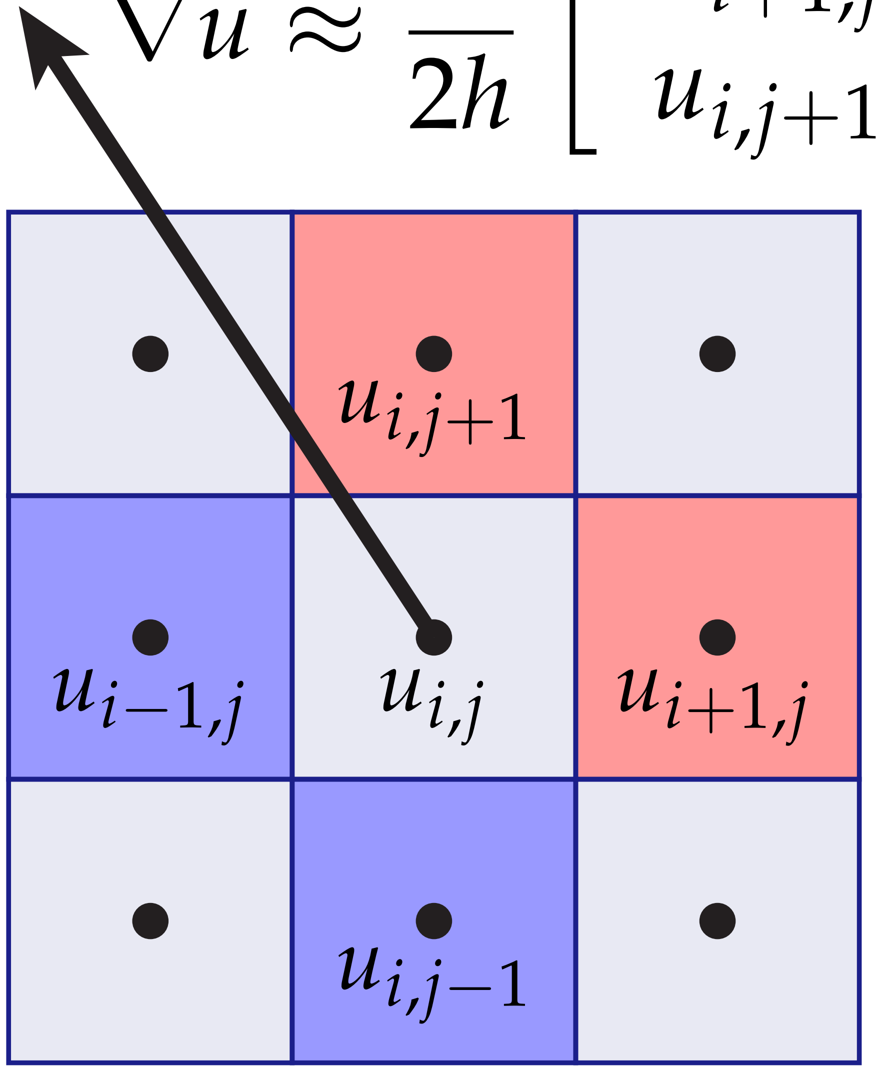
$$\boxed{\frac{u_{i-1,j} + u_{i+1,j} + u_{i,j-1} + u_{i,j+1} - 4u_{i,j}}{h^2}}$$



Finite Differences - Gradient (Centered)

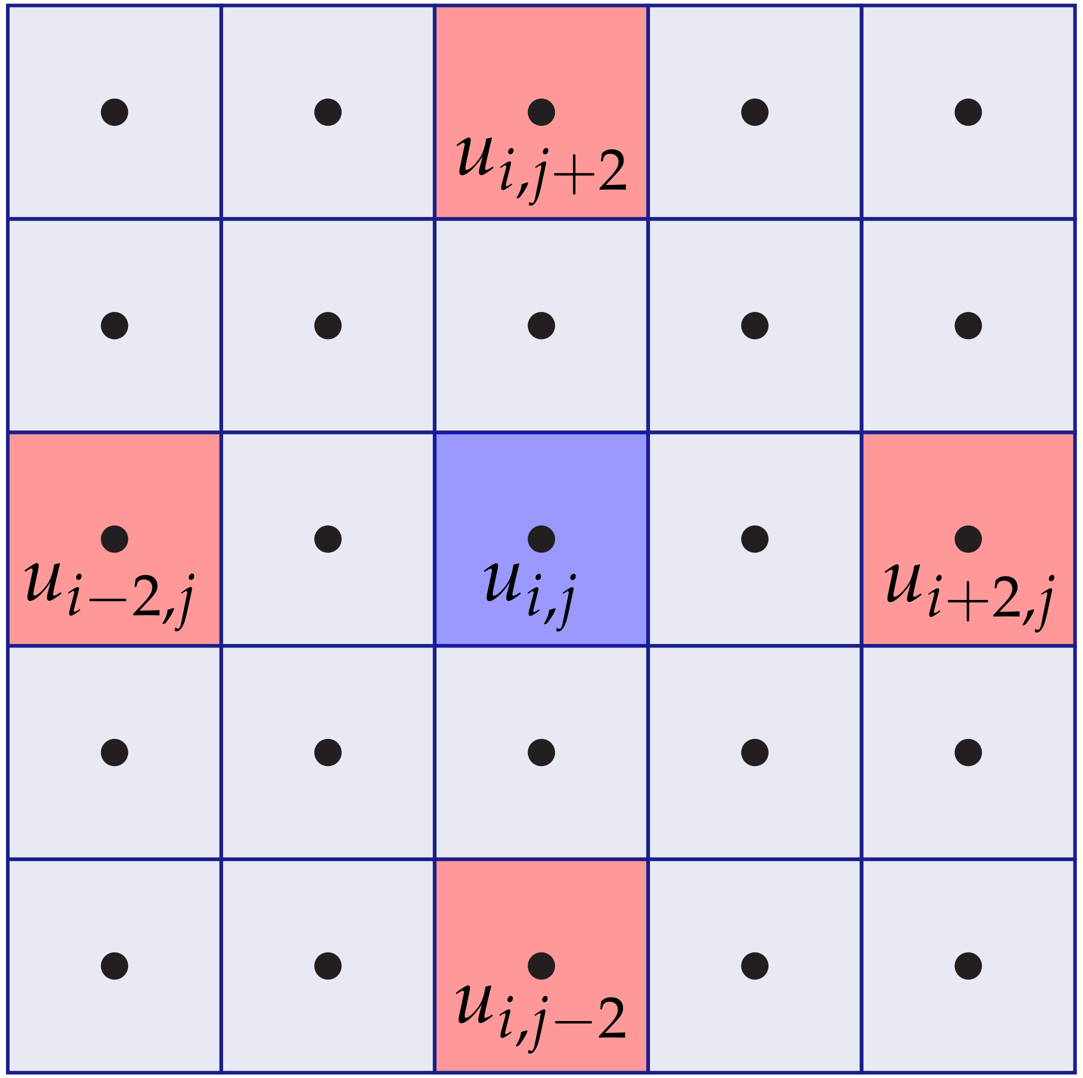
- One way to define gradient: take *centered differences* along each axis

$$\nabla u \approx \frac{1}{2h} \begin{bmatrix} u_{i+1,j} - u_{i-1,j} \\ u_{i,j+1} - u_{i,j-1} \end{bmatrix}$$



Finite Differences - Gradient (Centered)

- One way to define gradient: take *centered differences* along each axis
- Unfortunately doesn't play well with subsequent operations
  - *e.g.*, suppose we again use centered difference to approximate divergence of gradient (Laplacian)
  - resulting approximation has large "checkerboard" stencil, ignores local information
  - leads to bizarre numerical behavior (e.g., functions that don't belong in null space of Laplacian)

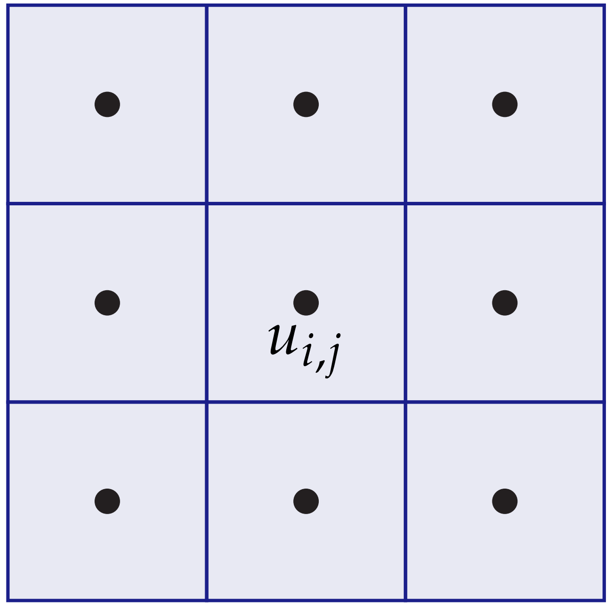


$$\Delta u = \nabla \cdot \nabla u \approx \frac{u_{i+2,j} + u_{i-2,j} + u_{i,j+2} + u_{i,j-2} - 4u_{i,j}}{4h^2}$$

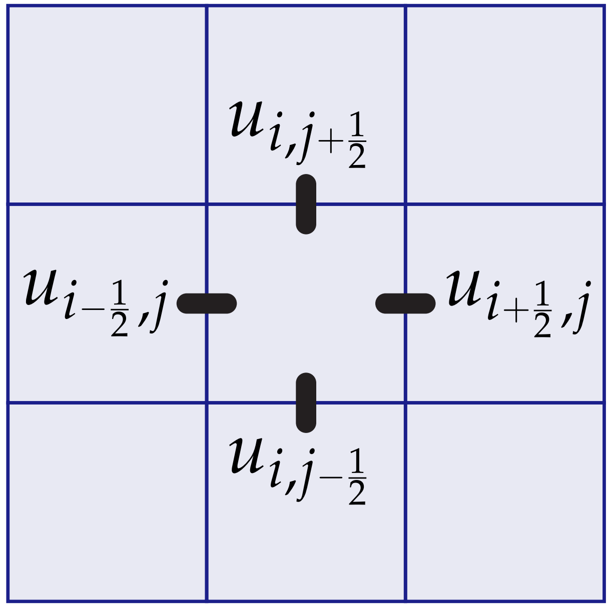
Finite Differences—Staggered Grid

- Instead, quite common to store quantities on a **staggered grid***
  - **scalar functions** (e.g., u) stored at cell *centers*
  - **vector fields** (e.g., ∇u) stored at cell *faces*
    - *e.g.*, in 2D store x-component on vertical faces, y-component on horizontal faces
    - use *half-index* notation
  - Finite difference approximations are otherwise identical (in each dimension)

scalar



vector

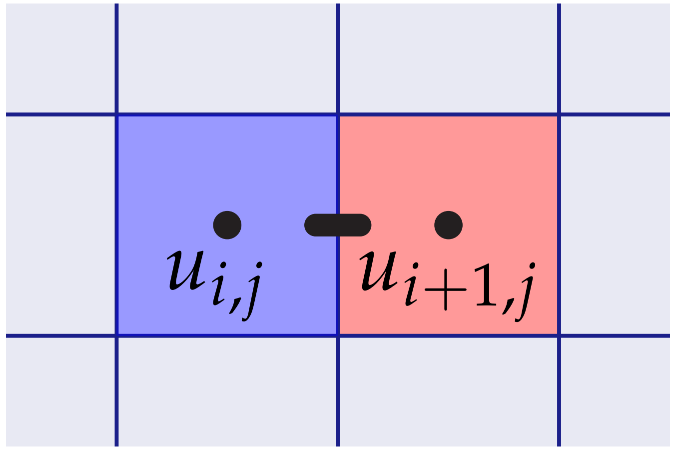


---

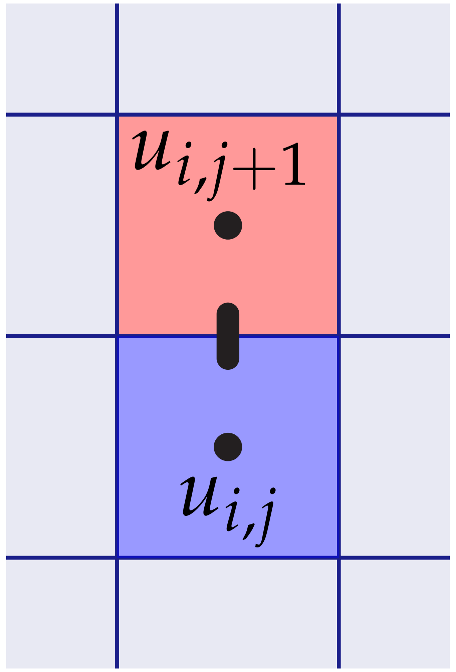
*Arises naturally from several perspectives: finite volumes, discrete exterior calculus, …

Gradient—Staggered Grid

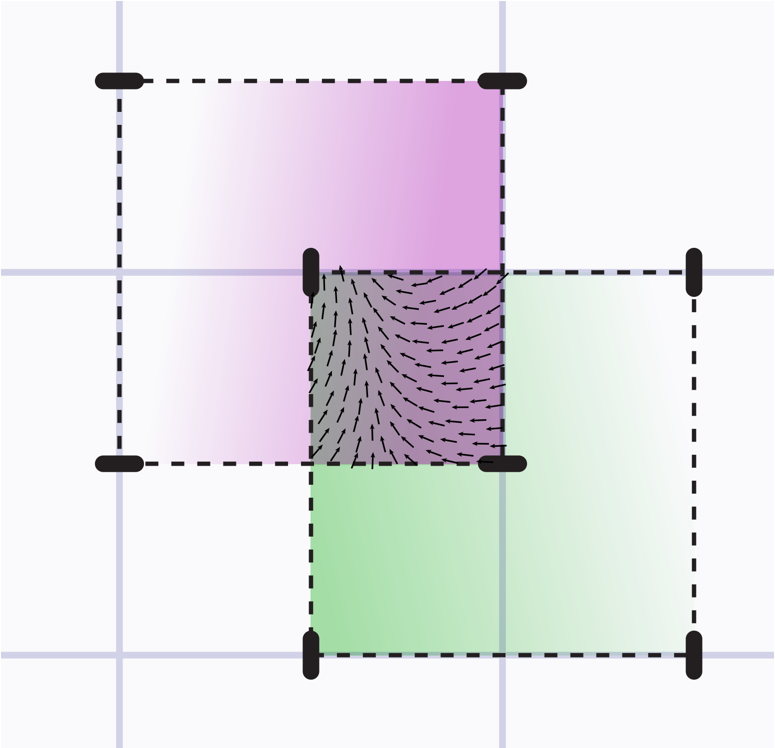
- **Example.** To approximate the gradient on a 2D staggered grid take a “horizontal” difference across each vertical edge, and “vertical” difference across each horizontal edge.
  - Note: there is no complete vector defined at any grid node!
  - If necessary (e.g., for visualization), can still *(bi-)linearly interpolate* nodal values to get continuous vector field in space



$$(\nabla u)_{i+\frac{1}{2},j} = u_{i+1,j} - u_{i,j}$$

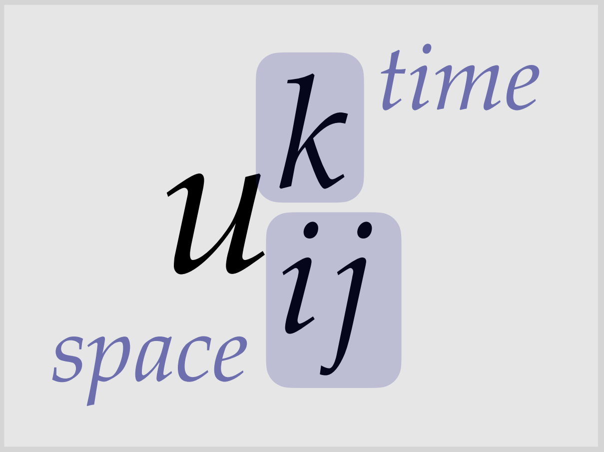


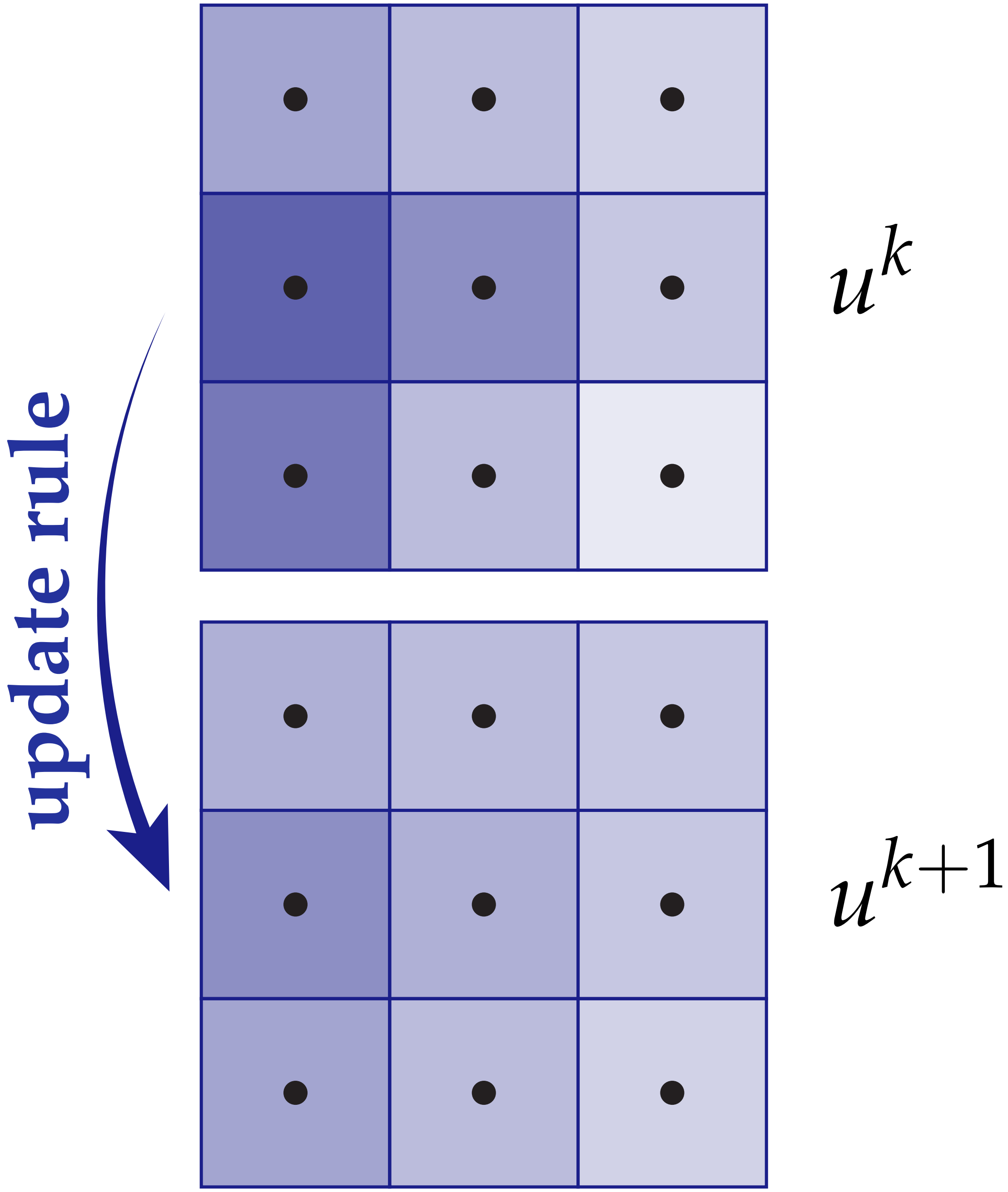
$$(\nabla u)_{i,j+\frac{1}{2}} = u_{i,j+1} - u_{i,j}$$



Finite Difference Solver

- Store values on a regular grid
- Use finite differences to approximate spatial derivatives
- Use same strategies as with ODEs to approximate temporal derivatives
- Now "march forward in time" by repeatedly solving for u^{k+1} (update rule)
- Notation:
  - subscript for space
  - superscript for time





Example — Heat Equation

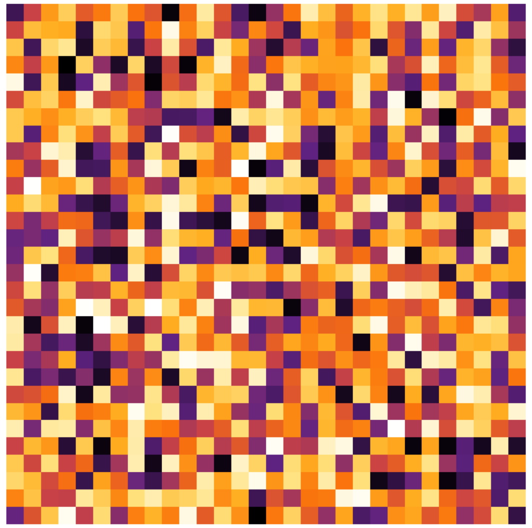
$$\frac{\partial}{\partial t}u = \Delta u$$ **Update rule:** forward Euler, finite difference Laplacian

$$u\big|_{t=0} = u_0$$ **Initial conditions:** all values given at initial time step (k=0)

$$u\big|_{(x,y)\in\partial\Omega} = g$$ **Boundary conditions:** values given at boundary nodes for all steps k

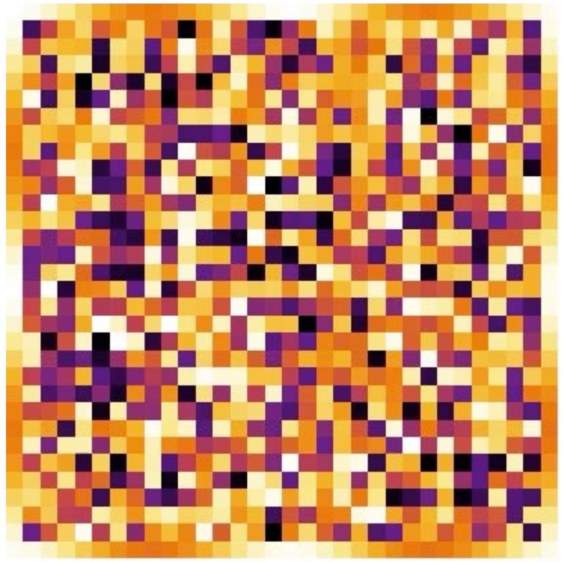
	1	
1	-4	1
	1	

$$\underset{\text{function at next moment in time}}{u_{ij}^{k+1}} = \overset{\text{function at previous time}}{u_{ij}^{k}} + \underset{\text{time step}}{\tau}\left(\frac{u_{i+1,j}+u_{i-1,j}+u_{i,j+1}+u_{i,j-1}-4u_{ij}}{\underset{\text{node spacing}}{h^2}}\right)$$



initial conditions u_0

boundary conditions g



solution $u(t)$

Heat Equation — Stability

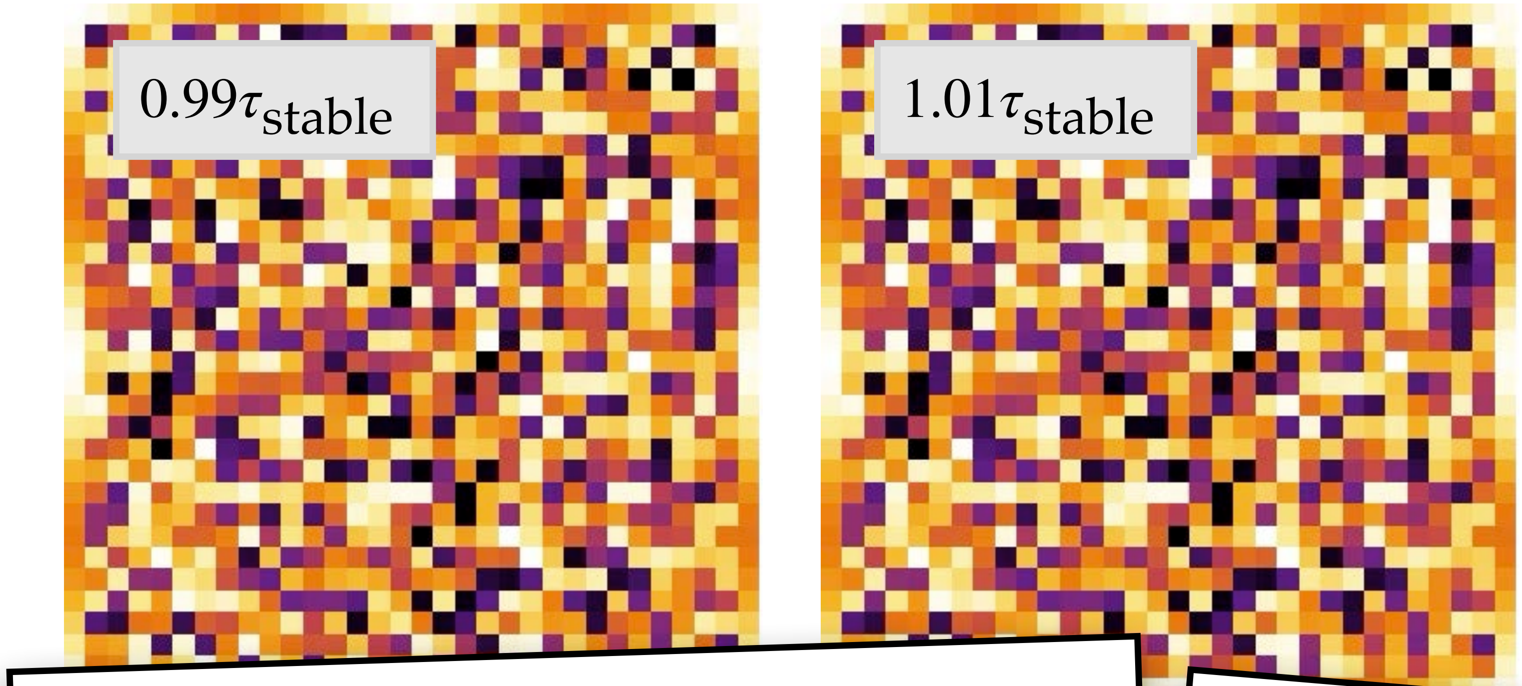
- **Q:** How do we pick maximum stable time step τ for solving heat equation?
- **A:** Same basic analysis as for ODEs, since a discretized PDE is just a *system of ODEs!*

forward Euler
(heat equation)

$$u^{k+1} = u^k + \tau \mathsf{L} u^k$$
$$u^k = (\mathsf{I} + \tau \mathsf{L})^k u^0$$

eigenvectors/ eigenvalues
(Laplacian)

$$\mathsf{L} e_i = \lambda_i e_i$$



$$u = \sum_i c_i e_i, \quad c_i := u^{\mathsf{T}} e_i$$

$$\mathsf{L}u = \sum_i c_i \mathsf{L} e_i = \sum_i \lambda_i c_i e_i$$

$$(\mathsf{I} + \tau \mathsf{L})^k u = \sum_i (1 + \tau\lambda_i)^k c_i e_i$$

$$|1 + \tau\lambda_{\text{max}}| < 1$$

As domain geometry becomes more complicated ($h \to 0$), cost grows **quadratically** in space *and* time ($O(h^4)$)

Many ways to do better…

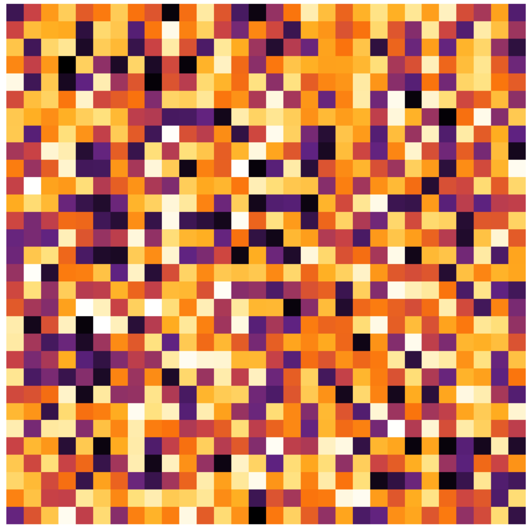
Rule of thumb:
$\tau_{\text{stable}} \approx \frac{1}{4}h^2$

Example — Heat Equation with Source

Easy enough to add other terms, like a source term f:

$$\frac{\partial}{\partial t} u = \Delta u + f$$

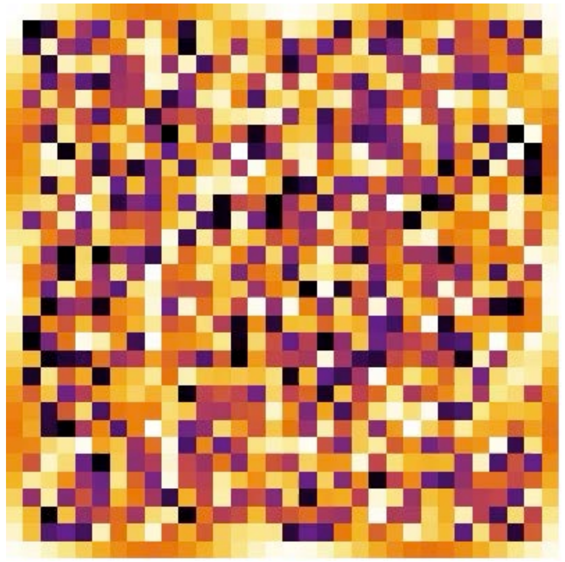
$$u_{ij}^{k+1} = u_{ij}^{k} + \tau \left(\frac{u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{ij}}{h^2} \right) + \tau f_{ij}$$



initial conditions u_0

source term f

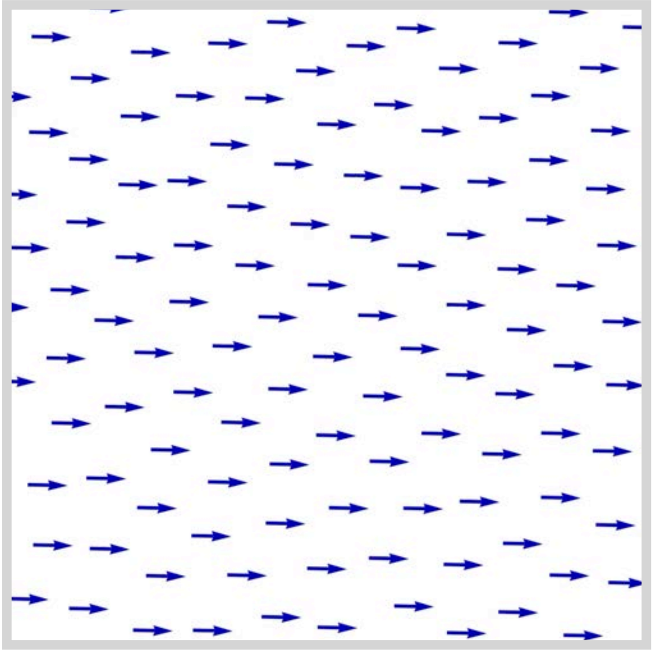
boundary conditions g



solution $u(t)$

Example — Convection-Diffusion Equation

- Convective term not as straightforward
  - central differences + forward Euler unstable for reasonable time steps
  - finite volume approach (integrate over cell) doesn't really lead anywhere…
- Principled approach: **upwind differencing**
  - 1D: *forward* difference when velocity *negative* & vice-versa
  - nD: apply 1D scheme in each dimension



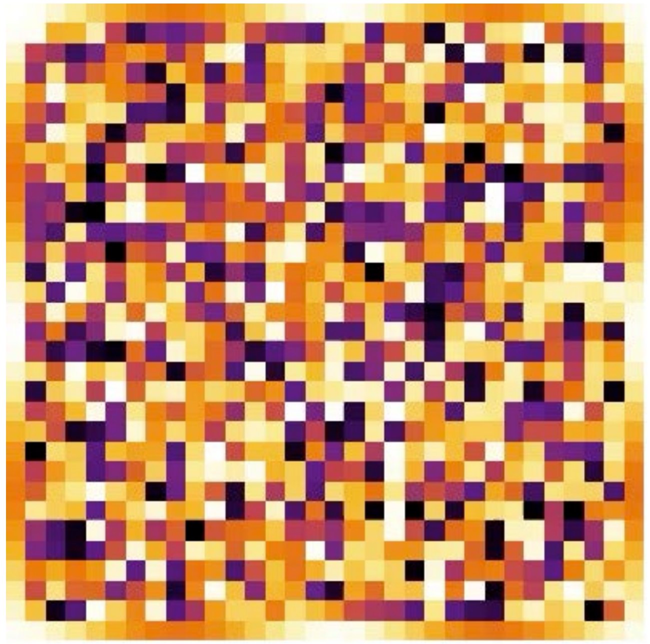
$$(D^+u)_i := \frac{u_{i+1} - u_i}{h} \qquad (D^-u)_i := \frac{u_i - u_{i-1}}{h}$$

forward difference *backward difference*

upwind (1D)

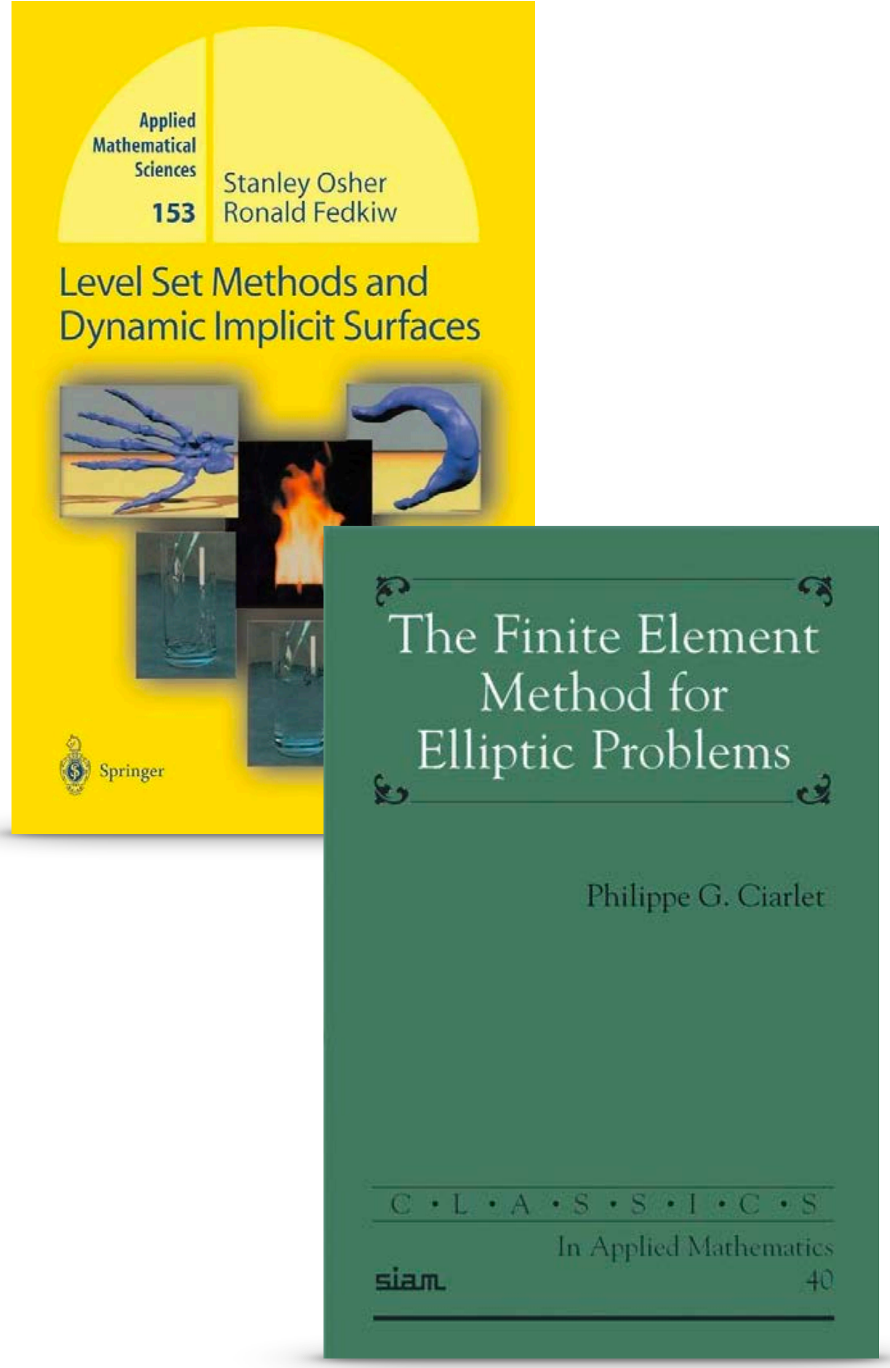
$$(\omega \cdot \nabla u)_i = \begin{cases} \omega_i (D^+u)_i, & \omega_i < 0, \\ \omega_i (D^-u)_i, & \omega_i > 0. \end{cases}$$

see: Osher & Fedkiw, *"Level Set Methods and Dynamic Implicit Surfaces"* §3.2



Numerical Methods for PDEs—Summary

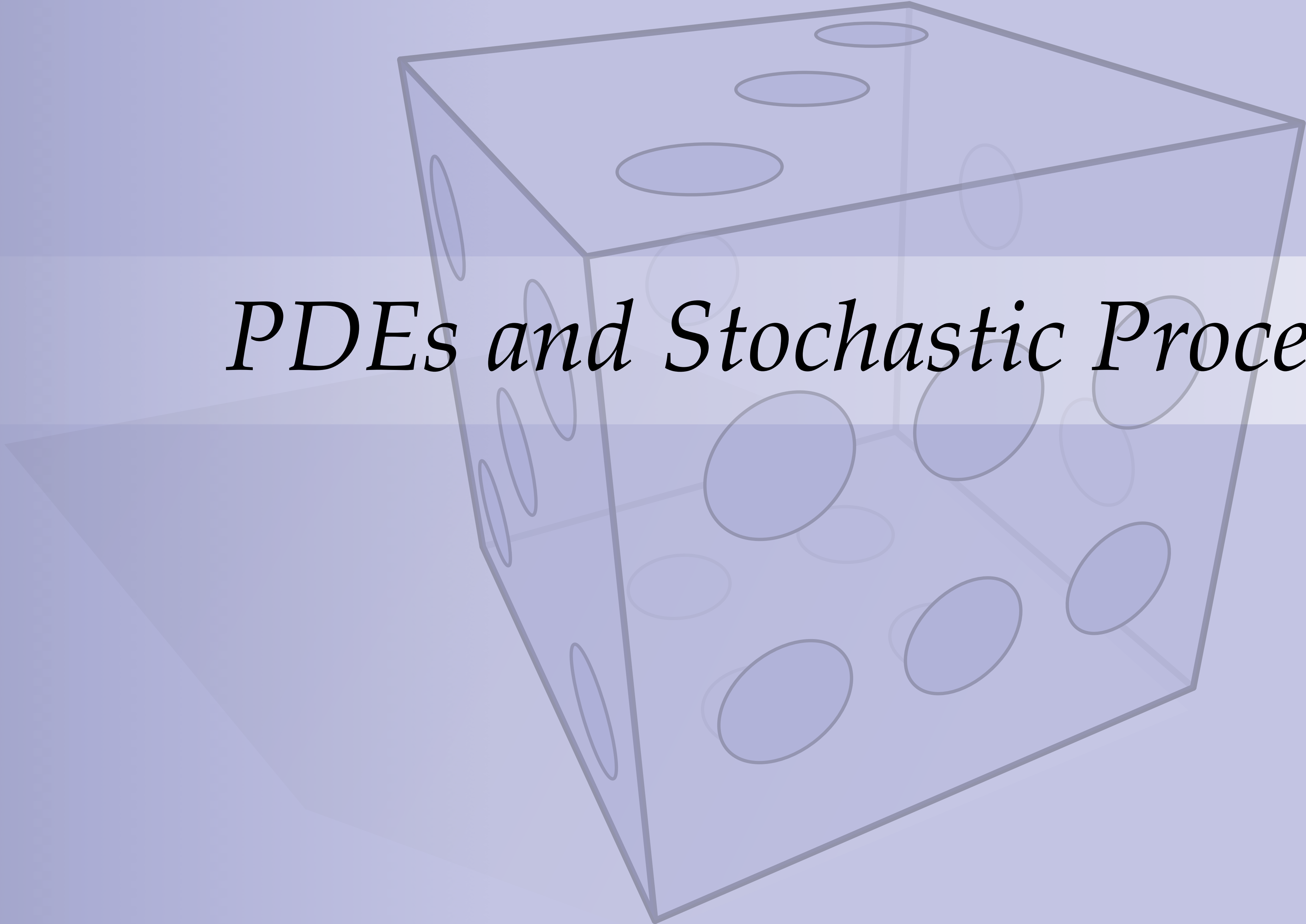
- Only scratched the surface
- *Many* ways to think about approximating solution, space & time derivatives
- Many criteria for "good" solver
  - **stability**: how big of a time step can I take?
  - **accuracy**: how fast do I converge under refinement?
- Just comparing stability & accuracy is **shallow** way of judging method—in practice have to deal with:
  - complex geometry
  - complex boundary conditions
  - singular or discontinuous functions
  - …
- No universally "best" method—*pick the right tool for the job!*



Applied Mathematical Sciences 153
Stanley Osher
Ronald Fedkiw
Level Set Methods and Dynamic Implicit Surfaces
Springer

The Finite Element Method for Elliptic Problems
Philippe G. Ciarlet
CLASSICS In Applied Mathematics 40
siam

PDEs and Stochastic Processes



Feynman-Kac Formula — Overview

- **Main question:** *what's the connection between PDEs and random walks?*
- **One direction:** solution to PDE given by average behavior of many random walks
  - encapsulated by *Feynman-Kac formula*
- **Another direction:** behavior of stochastic process (e.g., probability distribution) can be analyzed by solving a PDE
  - encapsulated by *Fokker-Planck equation*
- *Prelude:* connection between Brownian motion & heat equation

Review: Donsker's Theorem

- Consider a sequence of i.i.d. random variables $X_1, \ldots, X_n$
- Can associate these discrete steps with a time-continuous function

$$\widehat{W}_n(t) := \frac{1}{\sqrt{n}} \sum_{i=1}^{\lfloor tn \rfloor} X_i, \quad t \in [0,1]$$

spatial step size h

time step size $\tau = 1/n$

Ratio τ/h^2 is a fixed constant.

- **Donsker's theorem.**

  As $n \to \infty$, $\widehat{W}_n(t)$ converges to a standard Brownian motion W_t over $t \in [0,1]$

Review: Discrete Heat Equation

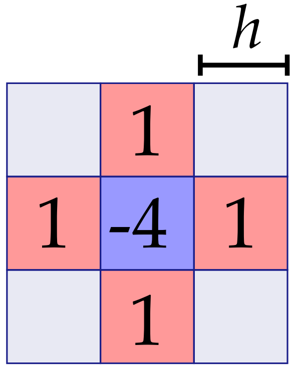
function at previous time

$$\boxed{u_{ij}^{k+1}} = \boxed{u_{ij}^{k}} + \boxed{\tau}\left(\frac{u_{i+1,j}+u_{i-1,j}+u_{i,j+1}+u_{i,j-1}-4u_{ij}}{\boxed{h^2}}\right)$$

solution value at next time

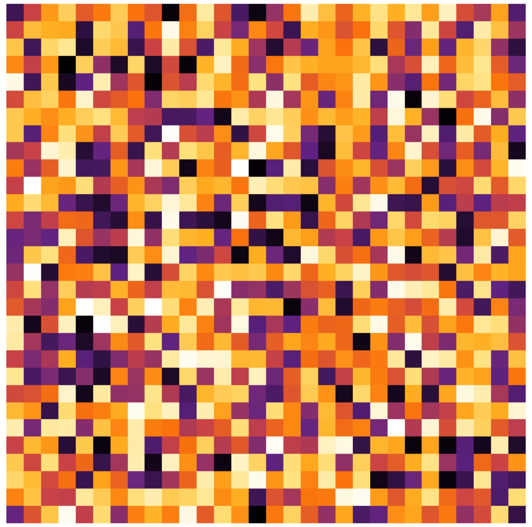
time step

node spacing



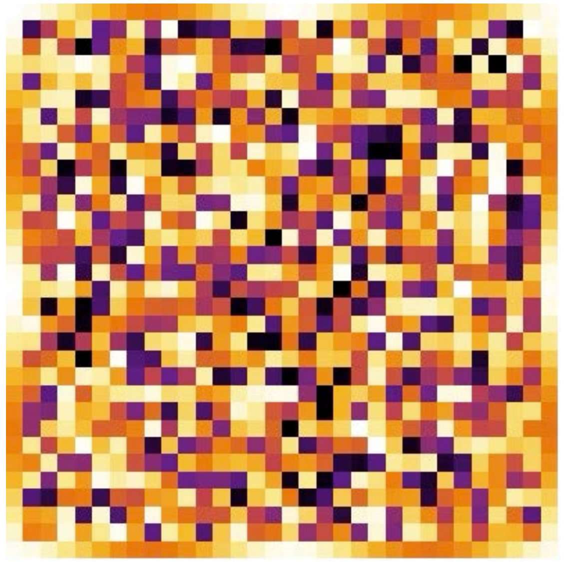
	1	
1	-4	1
	1	

$$\boxed{\tau_{\text{stable}} = \frac{1}{4}h^2}$$

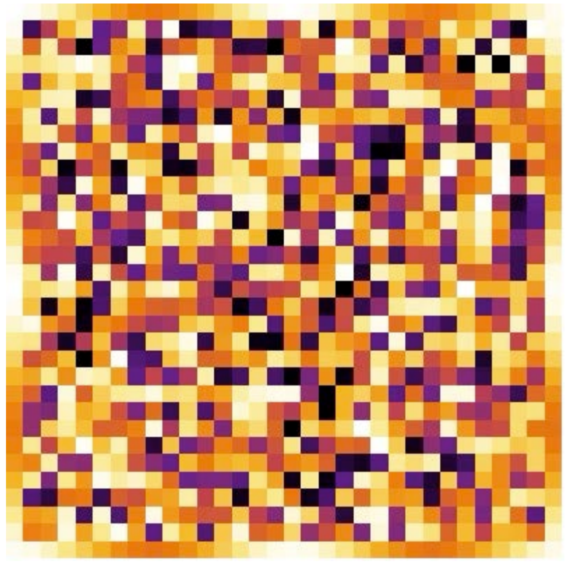


initial conditions u_0

boundary conditions g



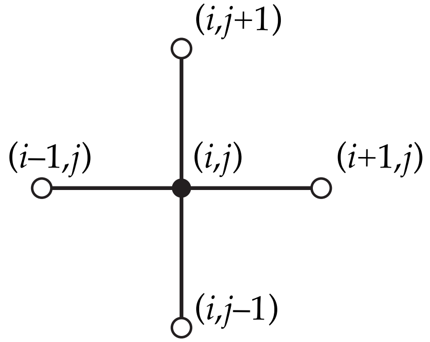
solution ($0.99\tau_{\text{stable}}$)



solution ($1.01\tau_{\text{stable}}$)

Prelude: Heat Equation & Random Walks

Discrete Random Walk



Consider a random walk on $\mathbb{Z}^2$

$$P(X_{k+1} = x | X_k = (i,j)) = \begin{cases} 1/4, & x = (i+1,j), (i,j+1), (i-1,j), (i,j+1), \\ 0, & \text{otherwise} \end{cases}$$

If P is the transition matrix and the row vector μ encodes a probability distribution, then

$$\mu^{k+1} = \mu^k P$$

$$\mu_{ij}^{k+1} = (\mu_{i+1,j}^k + \mu_{i,j+1}^k + \mu_{i-1,j}^k + \mu_{i,j-1}^k)/4$$

I.e., just repeatedly average the four neighbors.

Discrete Heat Equation

One step of 2D heat equation, via forward Euler:

$$u^{k+1} = (I + \tau L)u^k$$

Assume node spacing $h = 1$. Then

$$(Lu)_{ij} = u_{i+1,j} + u_{i,j+1} + u_{i-1,j} + u_{i,j-1} - 4u_{ij}$$

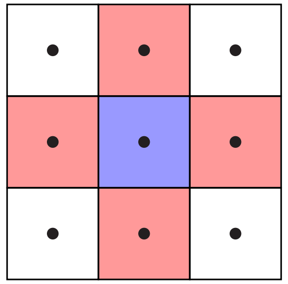
Also let $\tau = \frac{1}{4}h^2$ to ensure stability. Then

$$u_{ij}^{k+1} = (u_{i+1,j}^k + u_{i,j+1}^k + u_{i-1,j}^k + u_{i,j-1}^k)/4$$

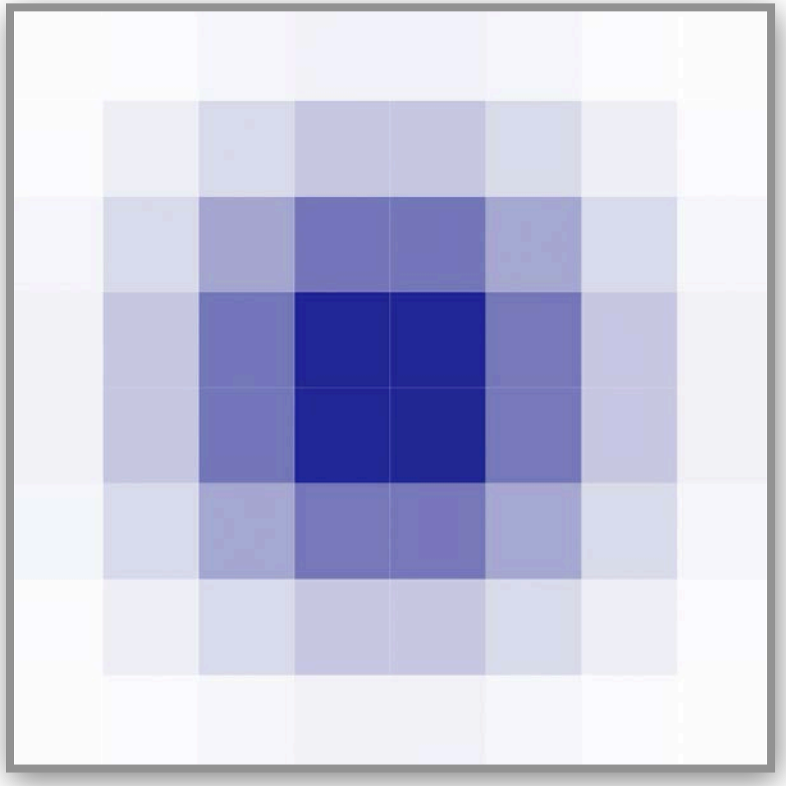
I.e., just repeatedly average the four neighbors.

Discrete heat equation models probability distribution of **discrete** random walk.

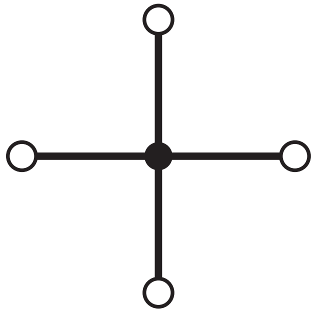
Prelude: Heat Equation & Random Walks



keeping fixed ratio
$\tau/h^2 = \frac{1}{4}$

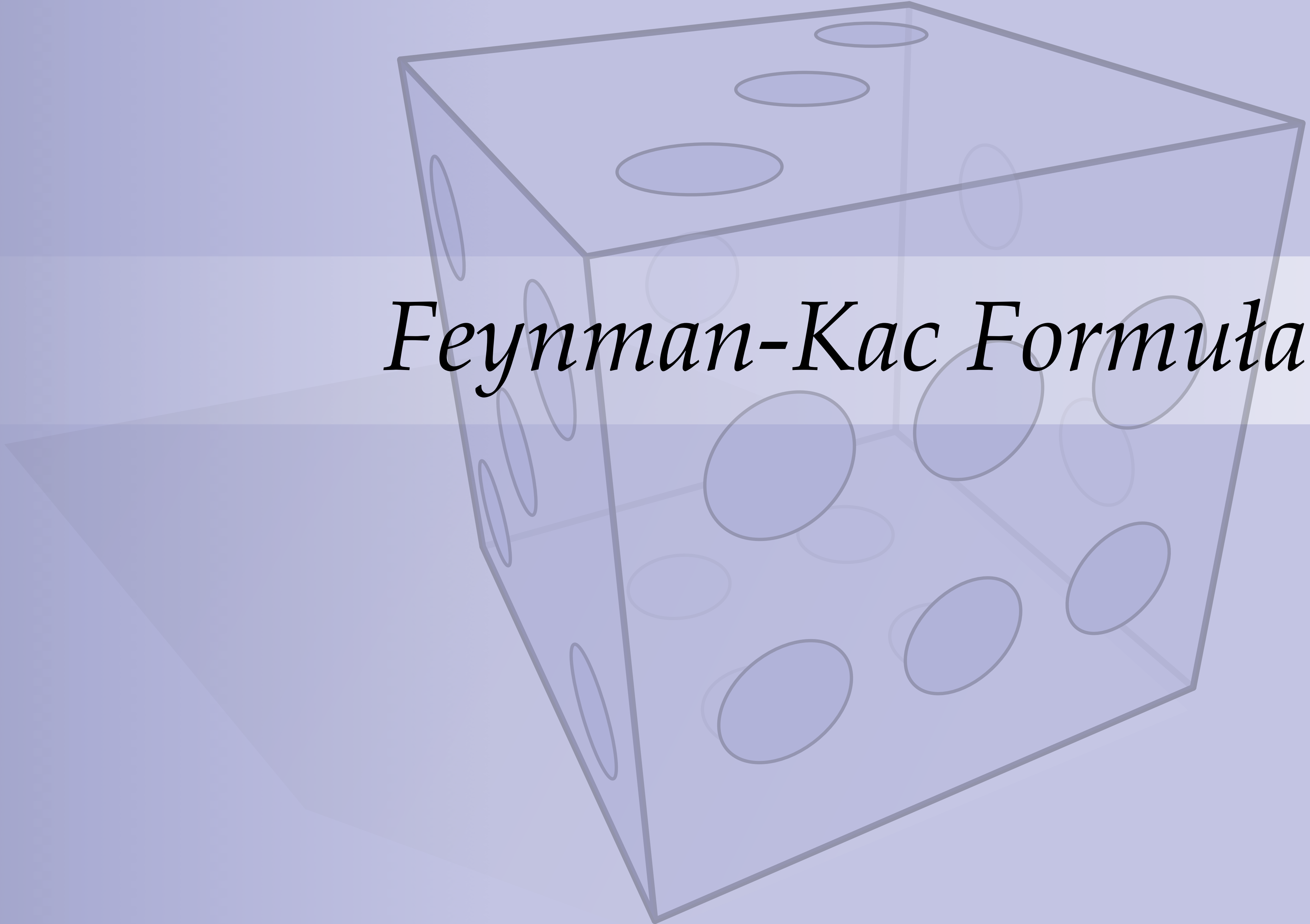


discrete heat equation converges
to continuous solution
(Taylor's theorem)



discrete random walk
converges to Brownian motion
(Donsker's theorem)

Heat equation models probability distribution of random walk.



Feynman-Kac Formuła

Feynman-Kac Formula — Overview

Partial Differential Equation (Convection-Diffusion)

$$\begin{aligned} \nabla \cdot (\alpha^2 \nabla u) + \vec{\omega} \cdot \nabla u - \lambda u &= f \text{ on } \Omega \\ u &= g \text{ on } \partial\Omega \end{aligned}$$

$$\downarrow$$

Stochastic Representation of Solution (Feynman-Kac)

$$dX_t = \vec{\omega}(X_t)dt + \alpha(X_t)dW_t$$

$$u(x) = \mathbb{E}\left[\int_0^T e^{-\int_0^t \lambda(X_s)ds} f(X_t)dt + e^{-\int_0^T \lambda(X_t)dt} g(X_t)\right]$$

Boundary Term

$$\begin{aligned} \Delta u &= 0 \text{ on } \Omega \\ u &= \underbrace{g}_{\text{boundary}} \text{ on } \partial\Omega \end{aligned}$$

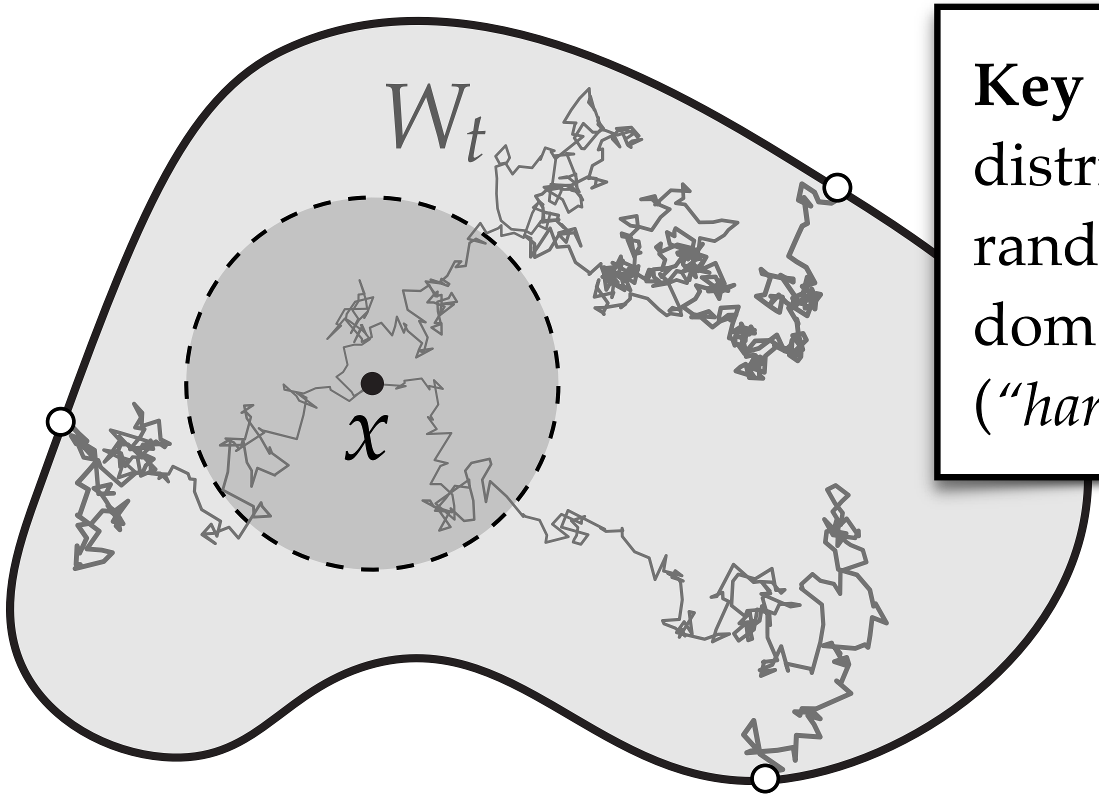
differential equation
(Laplace)

→

KAKUTANI'S PRINCIPLE

$$\mathbb{E}\left[g(\underbrace{W_T}_{\text{Brownian process}}) \,|\, W_0 = x\right]$$

stochastic representation



Key idea: track distribution of where random walkers first hit domain boundary ∂Ω (*"harmonic measure"*).

Source Term

SOURCE

$$\begin{aligned}\Delta u &= f \text{ on } \Omega\\ u &= g \text{ on } \partial\Omega\end{aligned}$$

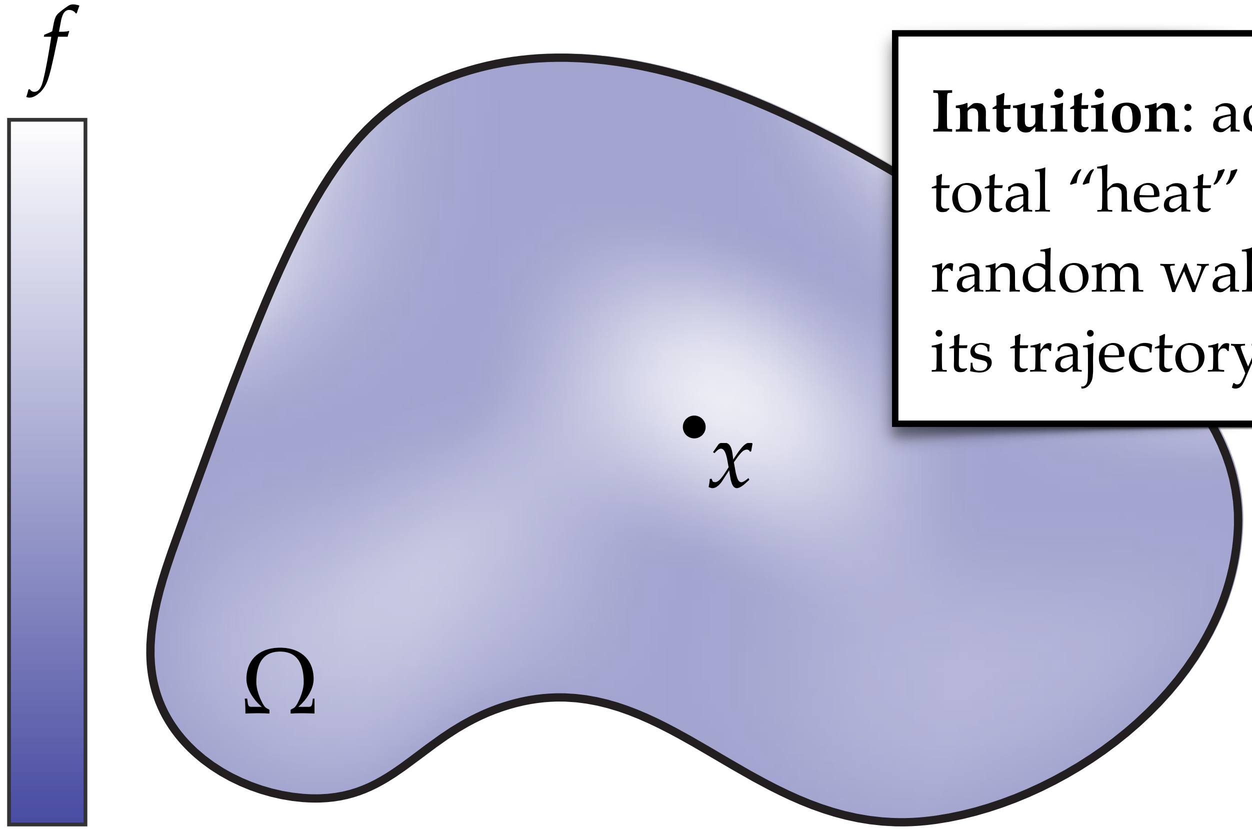
differential equation
(Poisson)

$$\longrightarrow$$

TIME TO REACH BOUNDARY

$$\mathbb{E}\left[\int_0^T f(W_t)\,dt \,\middle|\, W_0 = x\right]$$

stochastic representation



Intuition: accumulate total “heat” felt by random walker along its trajectory.

Absorption Term

ABSORPTION

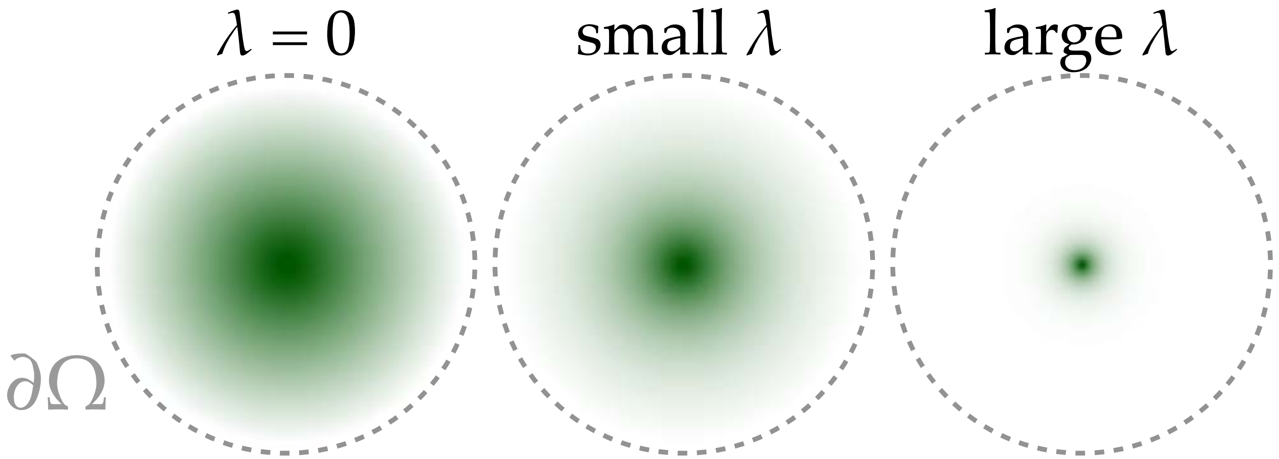
$$\begin{aligned} \Delta u + \lambda u &= f \quad \text{on } \Omega \\ u &= g \quad \text{on } \partial\Omega \end{aligned}$$

differential equation
(screened Poisson)

EXPONENTIAL DECAY

$$\mathbb{E}\left[\int_0^T \underbrace{e^{-\lambda t} f(W_t)\,dt}_{\text{source term}} + \underbrace{e^{-\lambda T} g(W_T)}_{\text{boundary term}}\right]$$

stochastic representation



some walkers get "absorbed"
before reaching boundary

Key idea: absorption of random walkers reduces contribution of "heat" from source & boundary

Drift Term

$$\begin{aligned}\Delta u + \overset{\text{DRIFT}}{\vec{\omega}\cdot\nabla u} &= f \quad \text{on } \Omega \\ u &= g \quad \text{on } \partial\Omega\end{aligned}$$

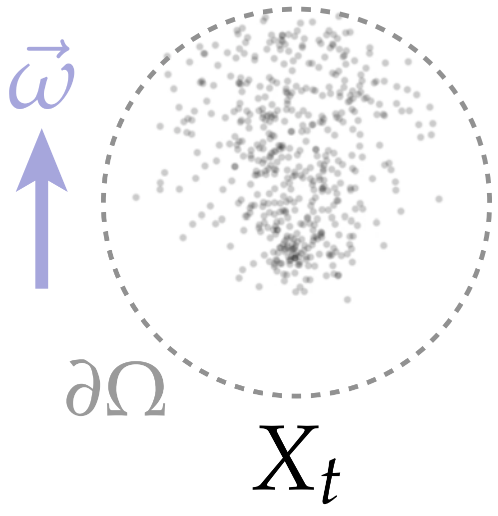
differential equation
(convection-diffusion)

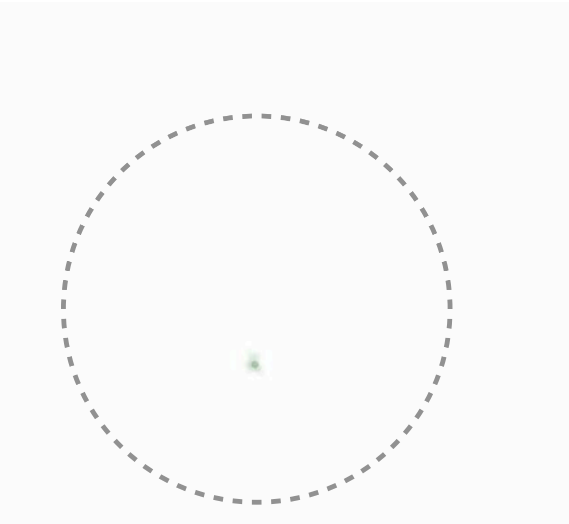
DIFFUSION PROCESS WITH DRIFT

$$dX_t = dW_t + \vec{\omega}(X_t)dt$$

$$\mathbb{E}\left[\int_0^T f(X_t)dt + g(X_t)\right]$$

stochastic representation





distribution of walks & exit points

Key idea: adding drift to random walk will change distribution of *where* we feel heat from the source & boundary

Feynman-Kac Formula

Summary: we can express solution to any convection-diffusion equation (PDE) via the average behavior of a diffusion process (SDE).

PARTIAL DIFFERENTIAL EQUATION (CONVECTION-DIFFUSION)

$$\overset{\textit{diffusion}}{\nabla \cdot (\alpha^2 \nabla u)} + \overset{\textit{drift}}{\vec{\omega} \cdot \nabla u} - \overset{\textit{absorption}}{\lambda u} = \overset{\textit{source}}{f} \text{ on } \Omega$$

$$u = \underset{\textit{boundary}}{g} \text{ on } \partial\Omega$$

STOCHASTIC REPRESENTATION OF SOLUTION (FEYNMAN-KAC FORMULA)

$$dX_t = \vec{\omega}\, dt + \alpha\, dW_t$$

$$u(x) = \mathbb{E}\left[\int_0^T e^{-\int_0^t \lambda(X_s) ds} f(X_t) dt + e^{-\int_0^T \lambda(X_t) dt} g(X_T) \right]$$

Feynman-Kac Formula

Summary: we can express solution to any convection-diffusion equation (PDE) via the average behavior of a diffusion process (SDE).

Partial Differential Equation (convection-diffusion)

$$\begin{aligned} \nabla\cdot(\alpha^2\nabla u)+\vec{\omega}\cdot\nabla u-\lambda u &= f \text{ on } \Omega \\ u &= g \text{ on } \partial\Omega \end{aligned}$$

Stochastic Representation of Solution (Feynman-Kac formula)

$$dX_t = \overset{\textit{drift}}{\vec{\omega}\,dt} + \overset{\textit{diffusion}}{\alpha\,dW_t}$$

$$u(x) = \mathbb{E}\left[\int_0^T e^{-\int_0^t \overset{\textit{absorption}}{\lambda(X_s)}ds}\,\overset{\textit{source}}{f(X_t)}\,dt + e^{-\int_0^T \overset{\textit{absorption}}{\lambda(X_t)}dt}\,\overset{\textit{boundary}}{g(X_T)}\right]$$

Feynman-Kac—Numerical Approximation

- **Q:** How could you <u>numerically</u> approximate u at some point x?
- **A:** *Expected value is an integral, so let's use Monte Carlo integration!*
- **Q:** Ok, but how do you evaluate samples Y_i?
- **A:** *We have an SDE, so let's use an SDE integrator! (e.g., Euler-Maruyama)*

"*Tour de force*" of numerical methods we've learned so far…

$$dX_t = \vec{\omega}\,dt + \alpha\,dW_t$$

① Simulate <u>many</u> random walks, using SDE integrator

$$Y := e^{-\int_0^T \lambda(X_t)dt} g(X_T) + \int_0^T e^{-\int_0^T \lambda(X_s)ds} f(X_t)dt$$

② For each walk, evaluate 1D time integrals with <u>deterministic</u> quadrature

$$\mathbb{E}[Y] = \frac{1}{N}\sum_{i=1}^{N} Y_i$$

③ To get final value, just take an average (Monte Carlo)

$$u(x) = \mathbb{E}\left[e^{-\int_0^T \lambda(X_t)dt} g(X_T) + \int_0^T e^{-\int_0^t \lambda(X_s)ds} f(X_t)dt\right]$$

Feynman-Kac—Numerical Example (Laplace)

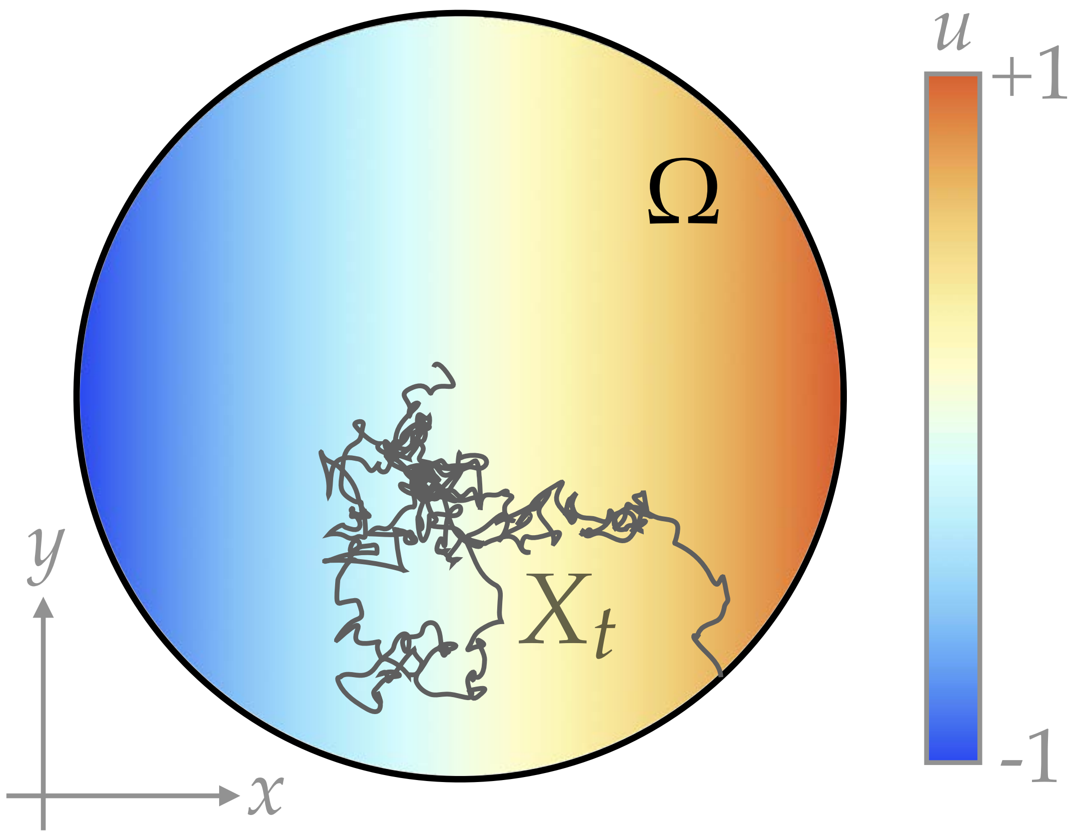
Example. Basic Laplace problem:

PDE

$$\Delta u(x,y) = 0 \quad \text{on } \Omega$$
$$u(x,y) = x \quad \text{on } \partial\Omega$$
$$\Omega := D^2 \subset \mathbb{R}^2$$

Q: What's the solution?

A: Just a linear function: $u(x,y) = x$



Q: What's the corresponding SDE?

A: Just plain Brownian motion

SDE $dX_t = dW_t$

Q: What's the Feynman-Kac representation of the solution?

A: Just the expected boundary value at the exit time T.

Solution

$$u(x) = \mathbb{E}[g(X_T)]$$

So, no quadrature in time to worry about (for now).

Feynman-Kac—Numerical Example (Laplace)

① Simulate random walk using Euler-Maruyama

$$dX_t = dW_t$$

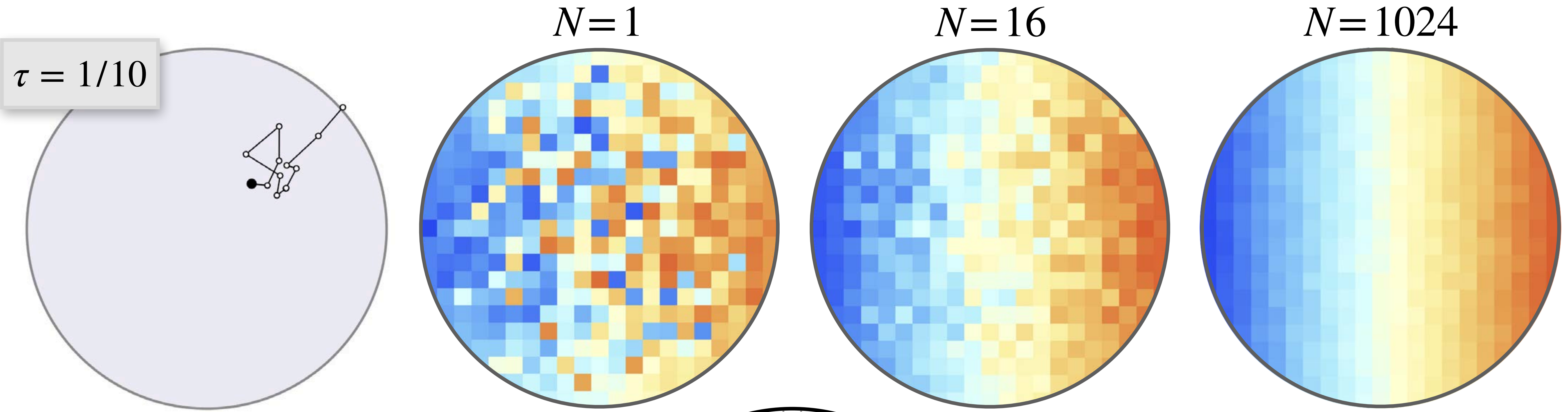
```
$\hat{u} \leftarrow 0$   initialize estimate
for $i = 1, \ldots, N$:
    $x \leftarrow x_0$   start at evaluation point
    until $\|x\| > 1$:   while still in the disk
        $\xi \sim \mathcal{N}^{2D}(0,1)$   sample direction
        $x \leftarrow x + \tau\xi$   take a step
    $\hat{u} \leftarrow \hat{u} + g(x)$   add boundary value
return $\hat{u}/N$   return the average
```

③ Overall estimate is average of boundary values at exit points

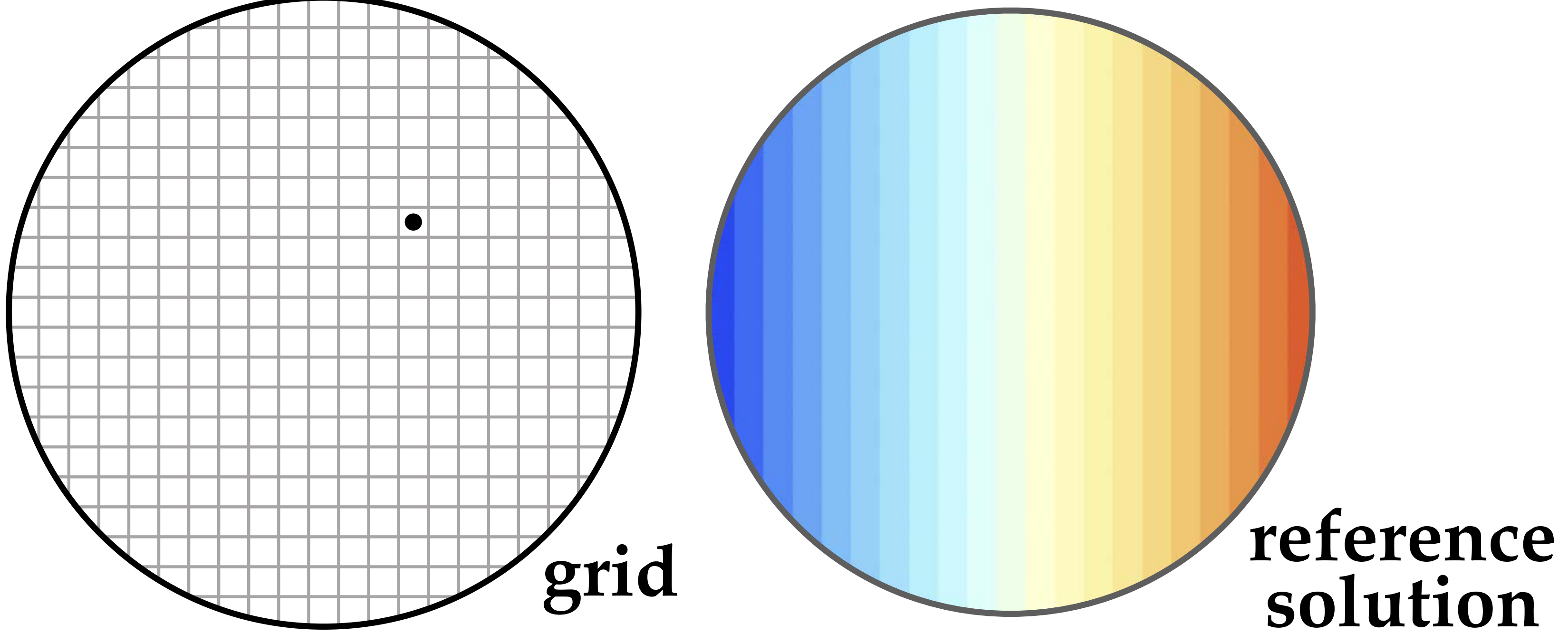
$$u(x) = \mathbb{E}[g(X_T)]$$

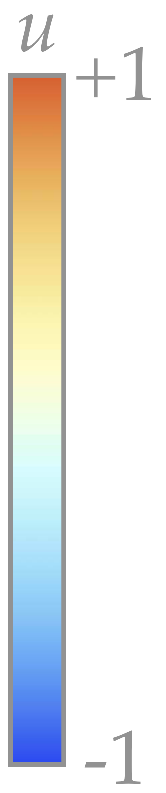
② Nothing to integrate in time

Feynman-Kac—Numerical Example (Laplace)



Can visualize solution by, e.g., evaluating at each point of a regular grid





Feynman-Kac—Numerical Example (Poisson)

Example. Let's add a source term:

PDE

$$\Delta u(x,y) = f \quad \text{on } \Omega$$

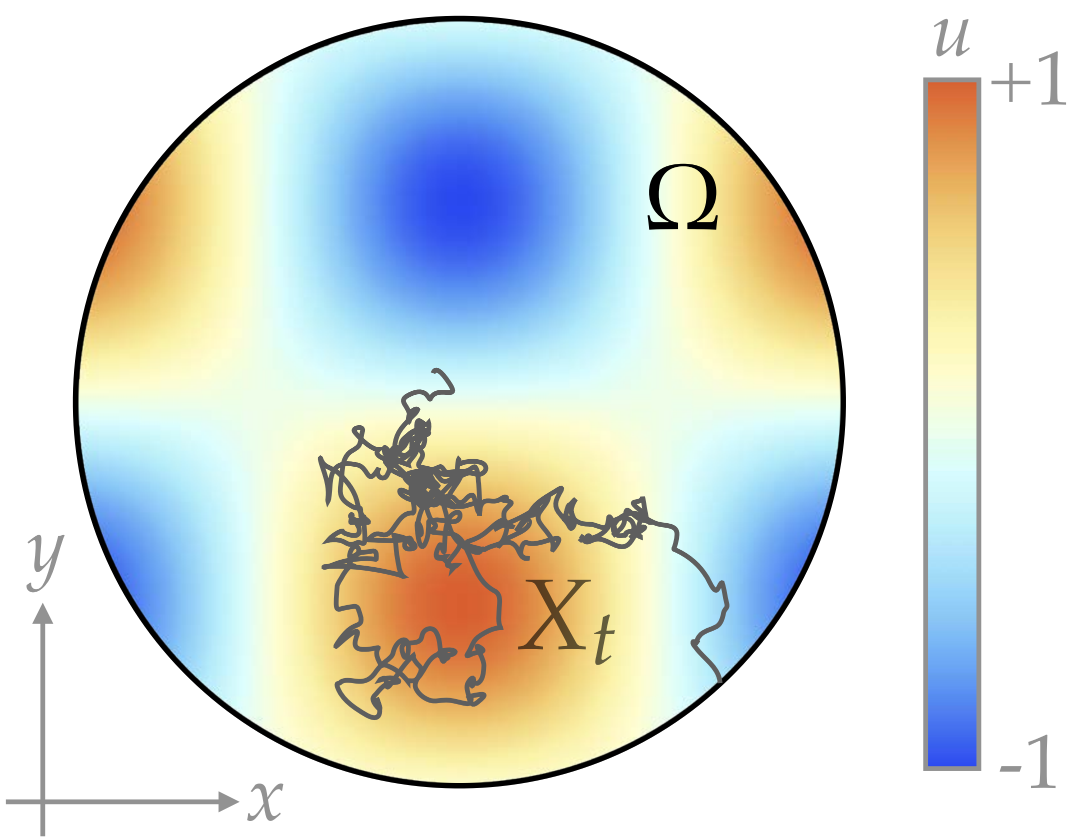
$$u(x,y) = u_0 \quad \text{on } \partial\Omega$$

$$\Omega := D^2 \subset \mathbb{R}^2$$

For testing purposes, let's define *f* such that we already know the solution:

$$f := \Delta u_0$$

$$u_0(x,y) := \cos(3x)\sin(3y)$$



Q: What's the SDE now?

A: Still just Brownian motion—we didn't change the differential operator (Δ)

SDE $dX_t = dW_t$

Q: How does the Feynman-Kac representation change?

A: We now have to integrate the source term along each walk

Solution

$$u(x) = \mathbb{E}[g(X_T)] + \mathbb{E}\left[\int_0^T f(X_t)\,dt\right]$$

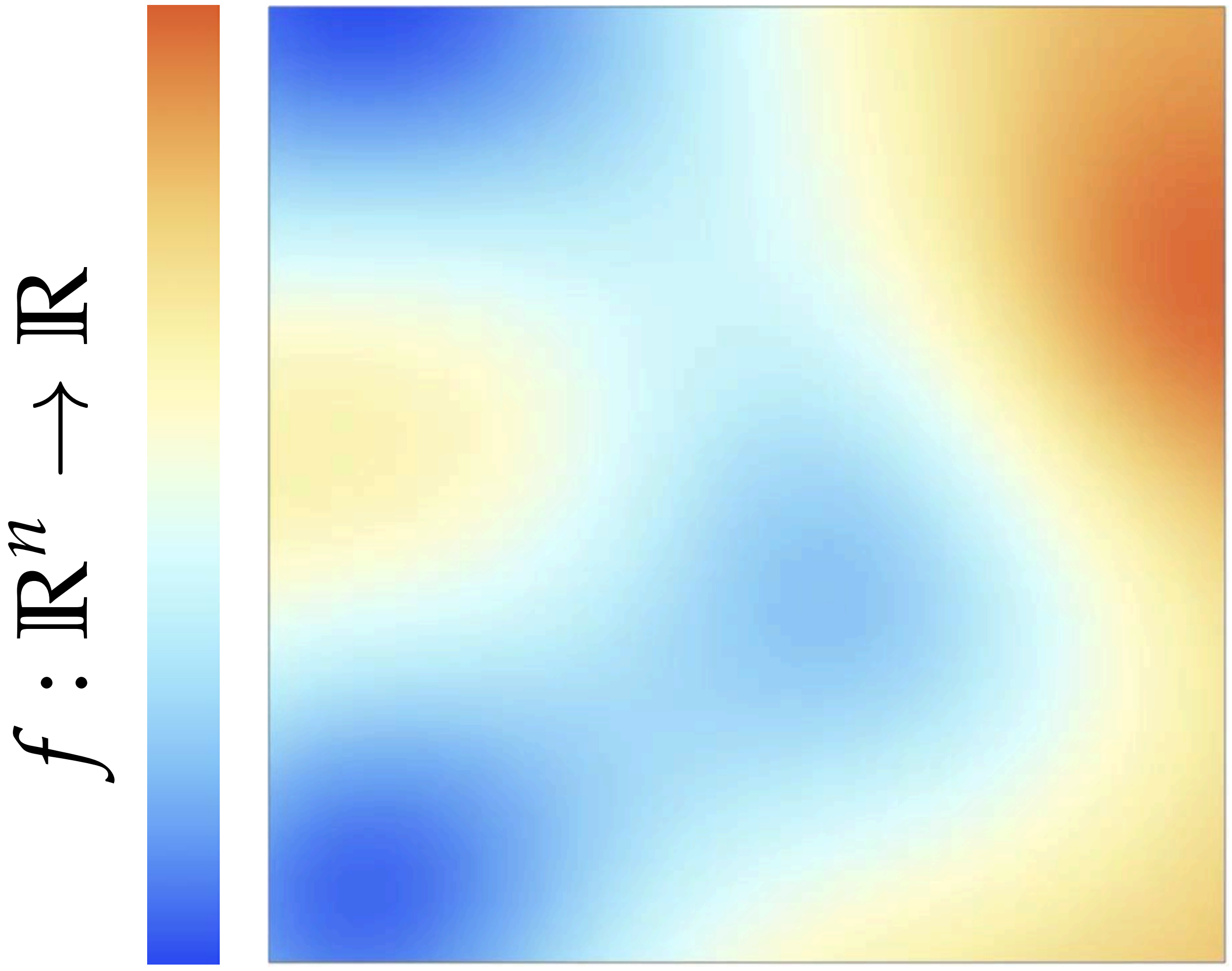
Integrating the Source Term

Q: Given that we've already generated a sequence of points $x_0, x_1, x_2 \ldots, x_K \in \mathbb{R}^2$ at equispaced time intervals $0, \tau, 2\tau, \ldots, K\tau$, what is a reasonable way to numerically approximate the source integral?

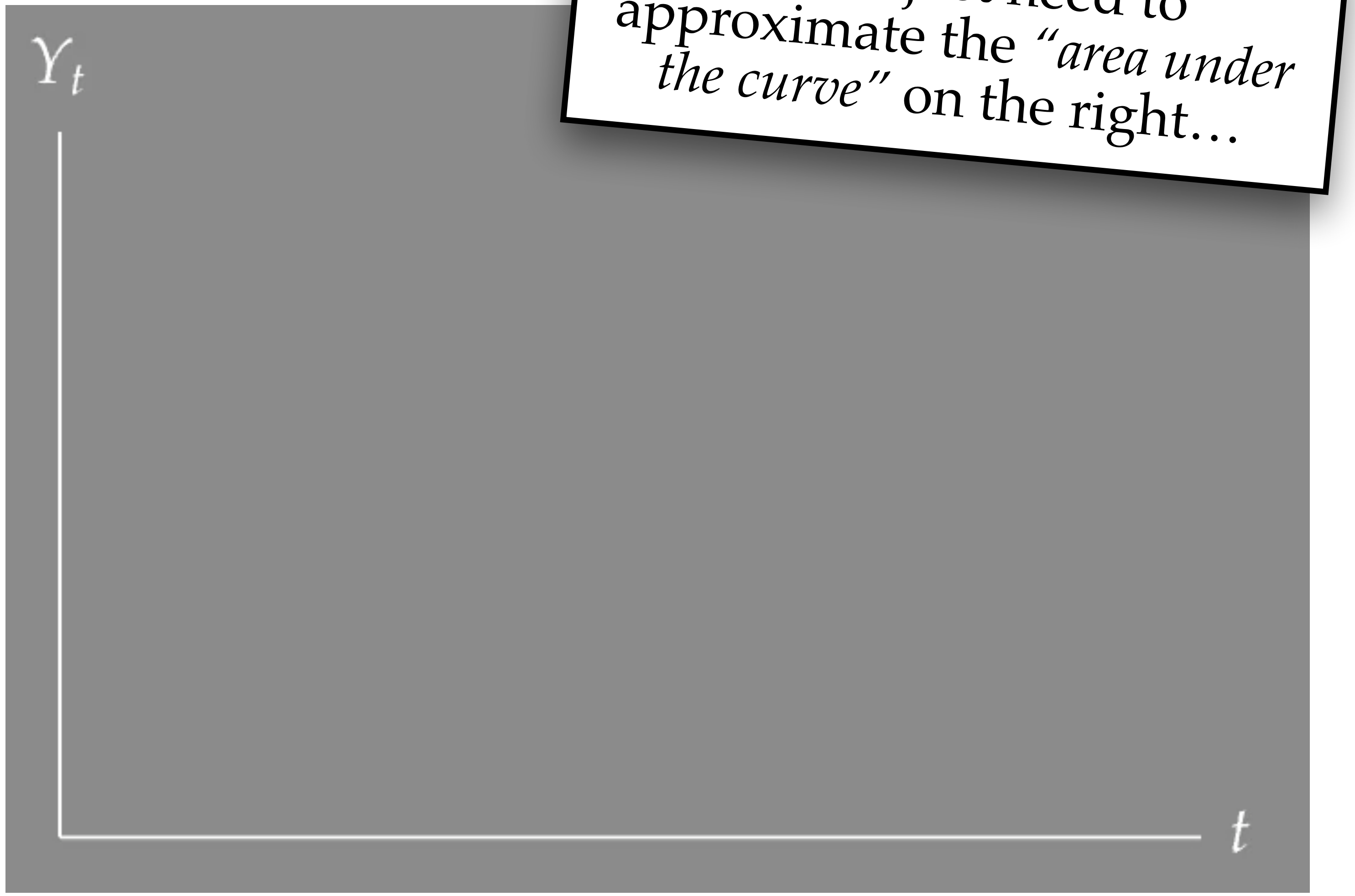
$$\int_0^T f(X_t)\,dt \approx \; ?$$

Integrating the Source Term

Recall: a function of stochastic process is a stochastic process



$$dX_t = \vec{\omega}(X_t)dt + \sigma dW_t$$



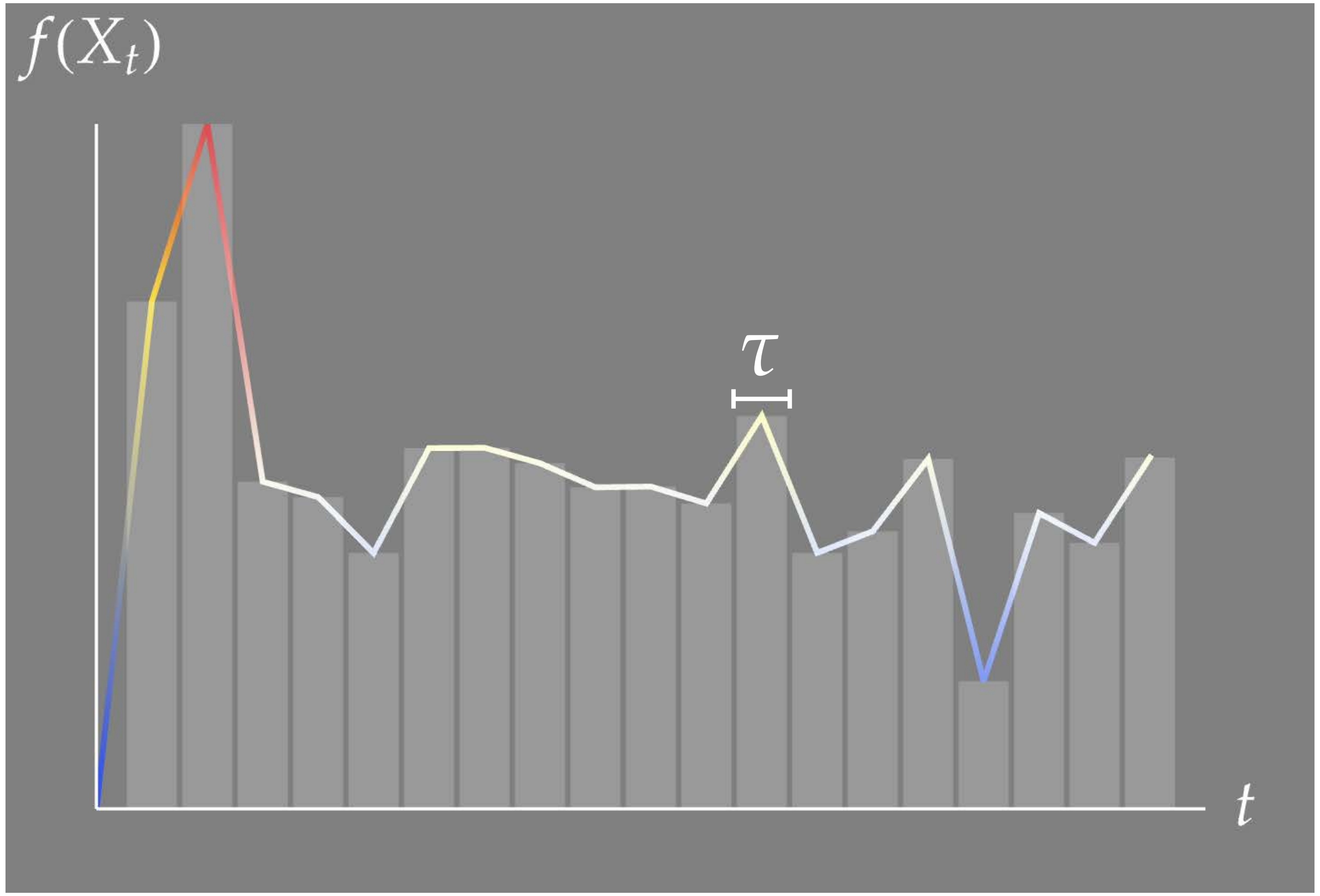
So, we just need to approximate the *"area under the curve"* on the right…

$$Y_t := f(X_t)$$

Integrating the Source Term

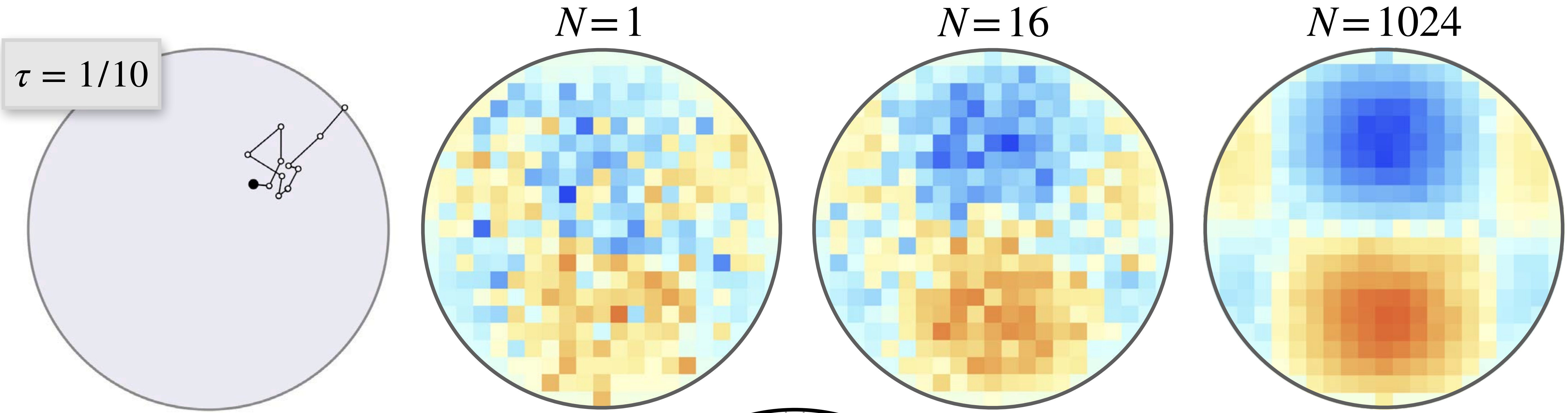
Q: Given that we've already generated a sequence of points $x_0, x_1, x_2 \ldots, x_K \in \mathbb{R}^2$ at equispaced time intervals $0, \tau, 2\tau, \ldots, K\tau,$ what is a reasonable way to numerically approximate the source integral?

$$\int_0^T f(X_t)\,dt \;\approx\; \sum_{i=1}^{K} \tau f(x_k)$$

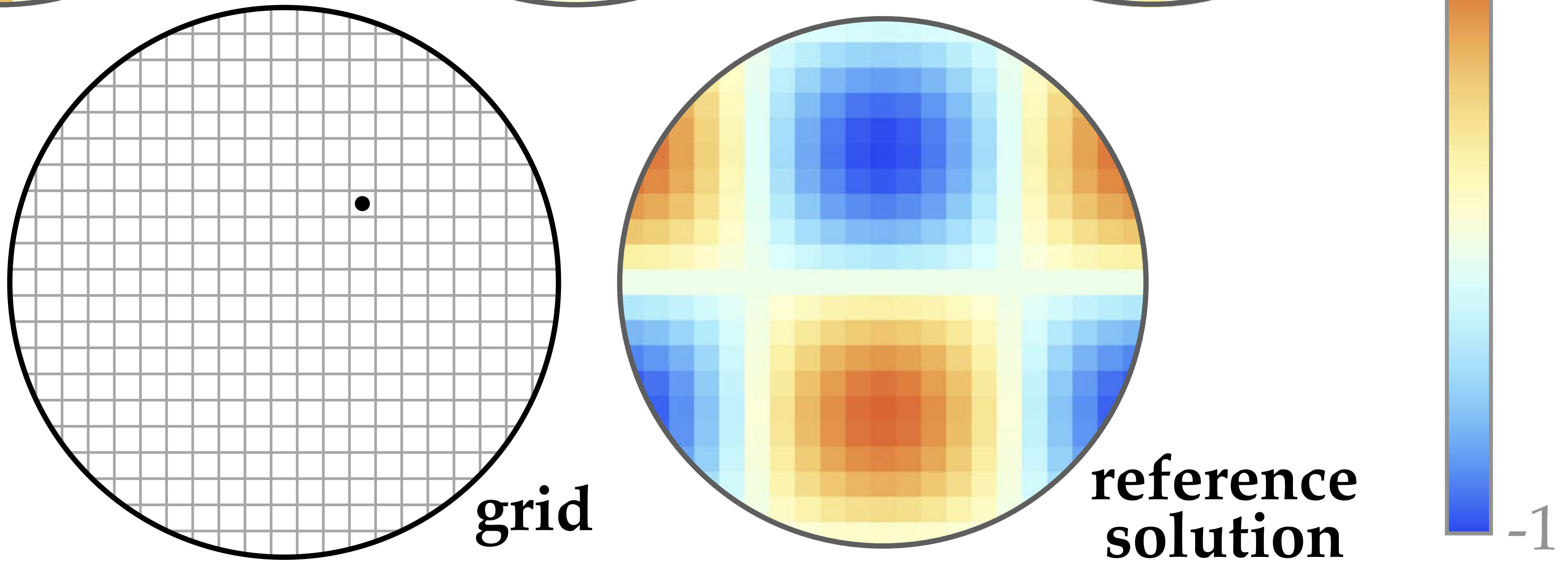


A: Can just use midpoint rule.

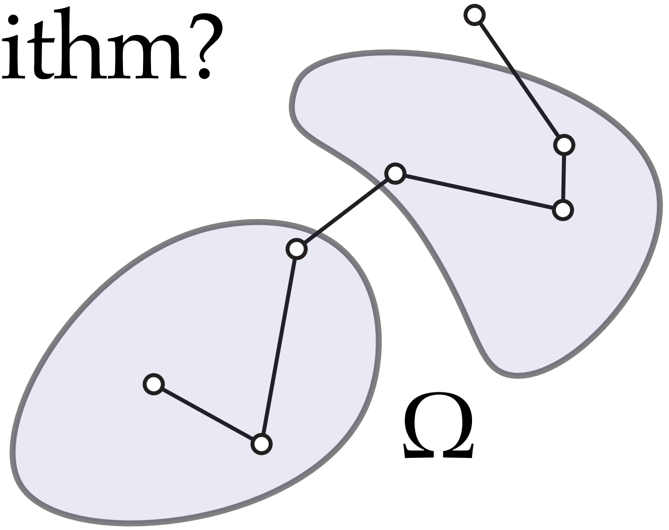
Feynman-Kac—Numerical Example (Poisson)



Converging… but not clearly to the correct solution!

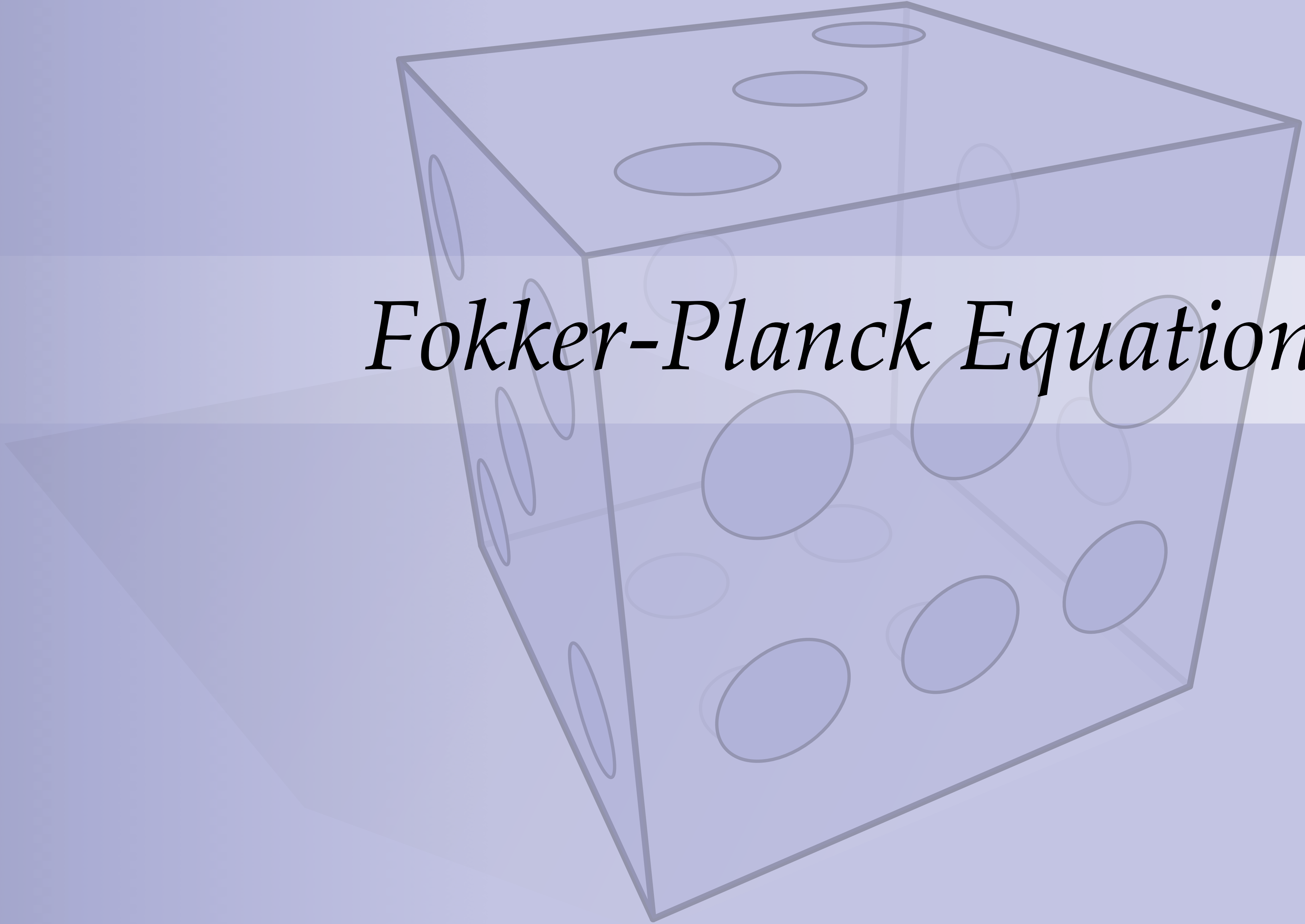


What Can Go Wrong?



- **Q:** Are there any sources of numerical error in our algorithm?
- **A:** Yes, many!
  - **number of samples** N (Monte Carlo estimate)
  - **step size** τ (Euler Maruyama)
    - crude approximation of time integrals (even if N is large!)
    - final step may be far from boundary
    - could even "jump" from one part of the domain to another
  - …

Later: can nicely address some of these problems with *walk on spheres*.



Fokker-Planck Equation

Fokker-Planck Equation — Overview

- **Main question:** *what's the connection between PDEs and random walks?*
- **One direction:** solution to PDE given by average behavior of many random walks
  - encapsulated by *Feynman-Kac formula*
- **Another direction:** behavior of many random walks can be described by PDE
  - encapsulated by *Fokker-Planck equation* (a.k.a. *Kolmogorov forward equation*)
- Not one single equation! Rather, for any SDE, can ask for its Fokker-Planck equation
  - often yields a familiar PDE (heat equation, convection-diffusion equation, etc.)
  - solution is probability distribution $p(x, t)$ describing location of X_t in space & time

Translating SDE to Fokker-Planck Equation

- Suppose we have an SDE, and want corresponding Fokker-Planck PDE
- We already know how to do this for many SDEs…
  - *just opposite direction of what we did for Feynman-Kac!*
- **Q:** For instance, what's the PDE for pure Brownian motion?
  - **A:** Heat equation
- **Q:** What's the PDE for a general diffusion process?
  - **A:** (Screened) convection-diffusion equation
- For general SDE, there may or may not be an established PDE…

Example—Langevin Equation

More generally: have to solve numerically! (*e.g., finite differences*)

- Let's return to Langevin equation from beginning (*a.k.a. Ornstein-Uhlenbeck process*)
- **Rare case:** simple enough to solve for probability $p(v,t)$ in closed form!

SDE

$$\overset{\textit{time evolution of velocity}}{dv_t} = -\overset{\textit{damping force}}{\gamma v_t} + \overset{\textit{Langevin force}}{\alpha dW_t}$$

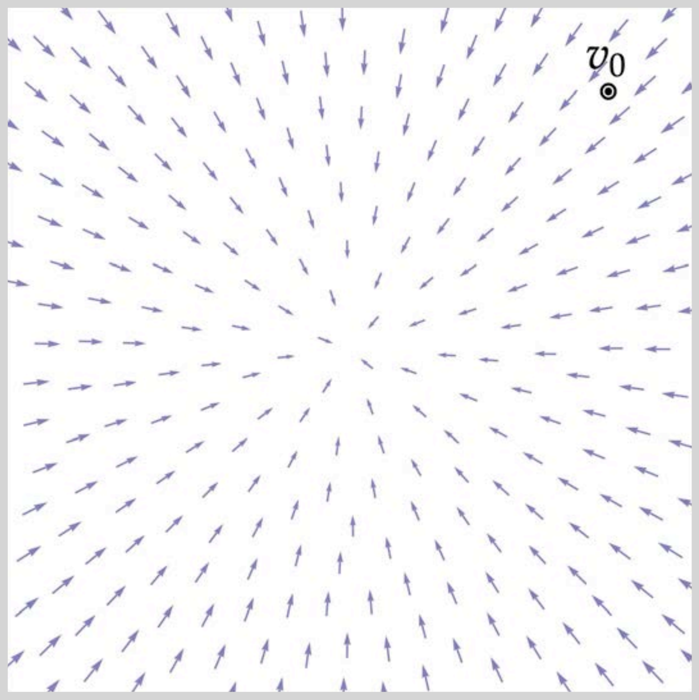
PDE

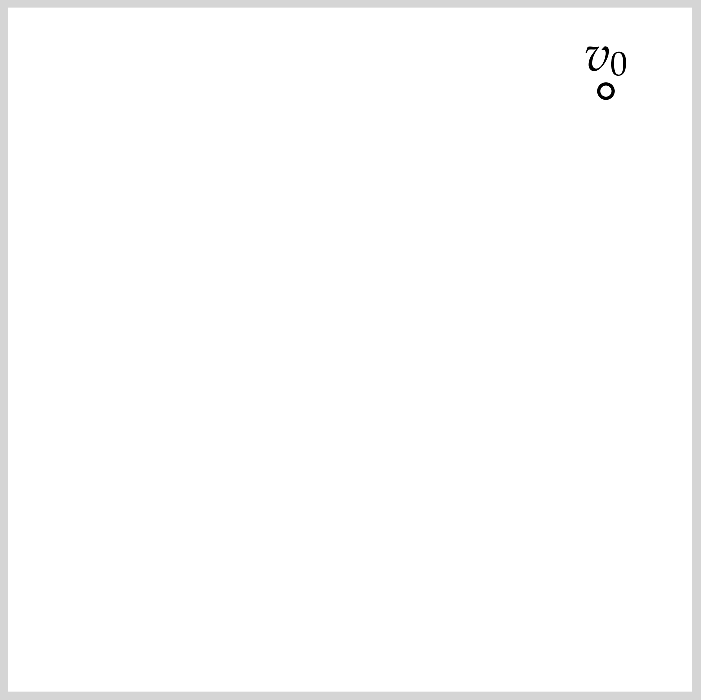
$$\frac{\partial}{\partial t} p = \overset{\textit{diffusion}}{\tfrac{1}{2}\alpha^2 \Delta^2 p} - \overset{\textit{convection}}{\gamma v \cdot \nabla p}$$

$$p(0) = \underset{\textit{starts at } v_0}{\delta(v - v_0)}$$

$$p(v,t) = \mathcal{N}(\mu(t), \Sigma(t))$$

$$\mu(t) = \overset{\textit{just like linear ODE...}}{e^{-\gamma t} v_0} \qquad \Sigma(t) = \overset{\textit{approaches } \alpha^2/2\gamma}{\frac{1-e^{-2\gamma t}}{2\gamma}\alpha^2 I}$$





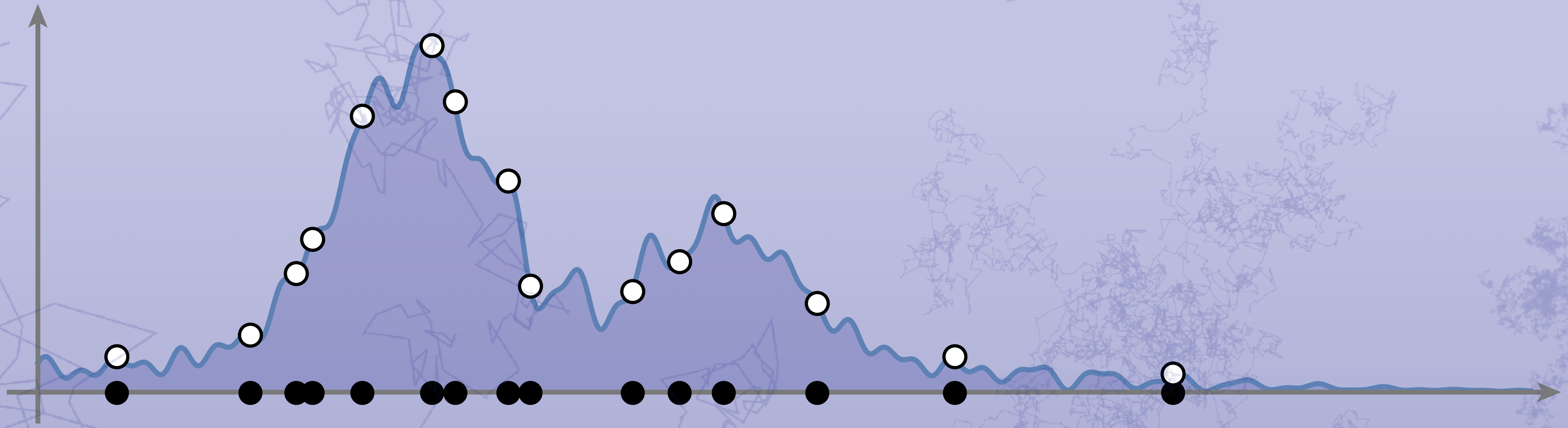
Using the Solution of the Fokker-Planck Equation

- Suppose we've solved for the distribution $p(x)$
- Can now compute, e.g., the expected value of any other quantity $a(x)$ by integration
  - e.g., suppose we want the average temperature of one of our "pollen grains" at some stopping time T

$$\mathbb{E}[a(X_T)] = \int_\Omega a(x) p(x,T)\, dx$$

Will see a lot more of this kind of thing as we develop MCMC algorithms!

Thanks!



MONTE CARLO METHODS
AND APPLICATIONS

Carnegie Mellon University | 21-387 / 15-327 / 15-627 / 15-860 | Fall 2023