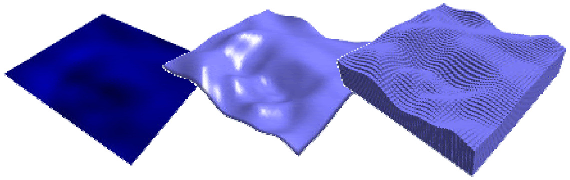
Partial Differential Equations for Computer Animation



Acknowledgement

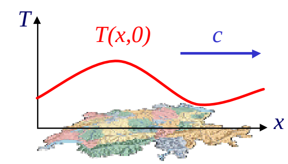
- A major part of this slide set and all accompanying demos are courtesy of Matthias Müller, ETH Zurich, NovodeX AG.

Motivation

- Most dynamic effects and physical processes can be described by partial differential equations PDEs
- We give a short overview of a large research field in mathematics and physics applied to the fields of animation and computer graphics
- Goals
  - You know what a PDE is
  - You know PDEs for specific effects
  - You know how to solve these PDEs numerically
  - You can use your knowledge to implement your own effects and animations

Introductory Example

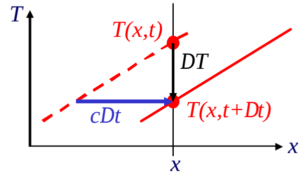
- Simulate temperature evolution
- One space dimension + time



- Given the temperature distribution $T(x,0)$ at time $t=0$ and wind speed c.
- Find PDE for the temperature evolution $T(x,t)$ at time t !

Advection

- How does the temperature change in one time step?



$$\frac{-\Delta T}{c\Delta t} \approx \frac{\partial T}{\partial x} \quad \xrightarrow{\Delta t \to 0} \quad \frac{\partial T}{\partial t} = -c\frac{\partial T}{\partial x} \text{ or } T_t = -cT_x$$

- PDE for $T(x,t)$ is 1-d advection (transport) equation.

Analytical Solution

- Any $T(x,t)$ of the form
  $T(x,t) = f(x-ct)$
  solves $T_t = -cT_x$
- The physical solution also needs to satisfy the initial condition
  $T(x,0) = T_0(x)$
- Thus, the solution is
  $T(x,t) = T_0(x-ct)$
- T is obtained by shifting T_0 through a distance ct without any change of shape of T_0.
- Only simple PDEs can be solved analytically.

Numerical Solution – Finite Differences

- Sample $T(x,t)$: $T^t[i] = T(i \cdot h, t \cdot \Delta t), \quad i \in (1,..,n), t \in (0,1,2..)$
- Discretize the derivatives (e.g. first-order backward scheme)
  $$T_t = -cT_x \quad \Rightarrow \quad \frac{T^{t+1}[i] - T^t[i]}{\Delta t} = -c\frac{T^t[i] - T^t[i-1]}{h}$$
- Solving for $T^{t+1}[i]$ yields the explicit update rule
  $$T^{t+1}[i] = T^t[i] - \Delta t \cdot c\frac{T^t[i] - T^t[i-1]}{h}$$
- Before starting the simulation, initialize $T^0[i]$
- Possible boundary conditions are periodic boundaries
  $T^t[0] = T^t[n], \quad T^t[n+1] = T^t[1]$

Outline

Mathematical Definition of a PDE

- Given a function u of two or more independent variables, e.g. $u(x,y,z,t)$
- A Partial Differential Equation (PDE) is an equation that determines the behavior of u in terms of
  - partial derivatives of u, e.g. u_{xy}, u_{tt},
  - u itself and
  - the independent variables, e.g. x,y,z,t
- Example: $u_{tt} = uu_{xy} + x$
- Notation: $u_{tt} = \frac{\partial^2}{\partial t^2} u(t,..), \quad u_{xy} = \frac{\partial^2}{\partial x \partial y} u(x,y,..),$

PDEs in Physics

- Independent variables are
  - Static Problem: x (1D), x,y (2D), x,y,z (3D)
  - Dynamic Problem: x,t (1D+1) x,y,t (2D+1) x,y,z,t (3D+1)
- Unknown function u can be
  - $u(x,..)$ a scalar, e.g. temperature T, density r
  - $\mathbf{u}(x,..) = [u(x,..), v(x,..), w(x,..)]^T$ a vector, e.g. displacement $\mathbf{u}$, velocity $\mathbf{v}$

PDE Classification

- Order
  - Order of PDE = Order of highest partial derivative
- Linear
  - u and its partial derivatives only occur linearly
  - coefficients may be functions of independent variables
- 2nd order linear PDE of 2 independent variables:

$$f_1(x,y)u_{xx} + f_2(x,y)u_{xy} + f_3(x,y)u_{yy} +$$
$$f_4(x,y)u_x + f_5(x,y)u_y + f_6(x,y)u + f_7(x,y) = 0$$

- Non-linear example

$$u_{xx} \cdot u_{xy} = u$$

PDE Classification

- Classification of 2nd order linear PDEs

$$A(x,y)u_{xx} + 2B(x,y)u_{xy} + C(x,y)u_{yy} = F(x,y,u,u_x,u_y)$$

- Hyperbolic $B^2 - AC > 0$
- Parabolic $B^2 - AC = 0$
- Elliptic $B^2 - AC < 0$

- Geometric motivation
- Different mathematical and physical behavior
- Determines type of boundary conditions needed
- For more than 2 independent variables use generalized criterion (see textbook)

PDE Classification

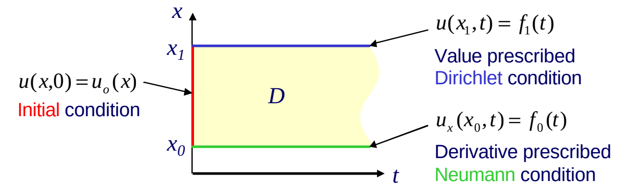
Hyperbolic
- time-dependent processes
- not evolving to a steady state, reversible
- propagate behavior undiminished, e. g. wave motion

Parabolic
- time-dependent processes
- evolving to a steady state, irreversible
- dissipative, e. g. heat diffusion

Elliptic
- time-independent
- already in steady state

Boundary Conditions

- Generally, there are many u which solve the PDE
- In physical applications only one solution is expected
- Typically u is only defined in a region D
- The physical solution is required to satisfy certain conditions on the boundary dD of D



Outline

Analytical Solution Techniques

- Analytical solutions only exist for small and simple problems
  - linear equations, simple geometry, initial and boundary conditions
- Examples for analytical solution techniques (see textbooks):
  - Method of separation of variables
  - Theory of characteristics
  - Green function method
  - Laplace transform
- Real-world problems are typically solved numerically

Numerical Solution - Overview

Governing Equations IC / BC	→	Discretization	→	System of Algebraic Equations	→	Equation (Matrix) Solver	→	Approx. Solution
Continuous Solution		Finite Difference Finite Volume Finite Element Spectral Boundary Element		Discrete Nodal Values		Explicit solution CG		

Discretization

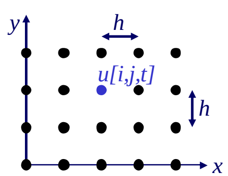
Time derivatives
- Finite-difference methods

Spatial derivatives
- Finite-difference method FDM
- Finite-element method FEM
- Finite-volume method FVM
- Spectral methods
- Boundary element methods
- …

The Finite Difference Method

- Simplest method to understand and implement
- Sample function u on a regular grid, e.g.

$$u[i,j,t] = u(i\cdot h, j\cdot h, t\cdot \Delta t) \qquad i\in(1,..,n),\ j\in(1,..,m),\ t\in(0,1,2,...)$$



- with grid spacing h and time step Δt

The Finite Difference Method

- Approximation of the temporal derivative

$$u_t[i,j,t] = \frac{u[i,j,t+1]-u[i,j,t]}{\Delta t} + O(\Delta t)$$

- Approximation of spatial derivatives (three possibilities)

$$u_x[i,j,t] = \frac{u[i+1,j,t]-u[i,j,t]}{h} + O(h)$$ forward scheme

$$u_x[i,j,t] = \frac{u[i,j,t]-u[i-1,j,t]}{h} + O(h)$$ backward scheme

$$u_x[i,j,t] = \frac{u[i+1,j,t]-u[i-1,j,t]}{2h} + O(h^2)$$ central scheme

- If information moves from left to right (as in $u_t = -cu_x$) backward is upwind (upwind is best choice)

The Finite Difference Method

- Approximation of higher derivatives

$$u_{xx}[i,j,t] = \frac{u_x[i,j,t] - u_x[i-1,j,t]}{h} = \frac{u[i+1,j,t] - 2u[i,j,t] + u[i-1,j,t]}{h^2} + O(h^2)$$

- Higher-order approximations

$$u_x[i,j,t] = \frac{u[i-2,j,t] - 8u[i-1,j,t] + 8u[i+1,j,t] - u[i+2,j,t]}{12h} + O(h^4)$$

$$u_{xx}[i,j,t] = \frac{-u[i-2,j,t] + 16u[i-1,j,t] - 30u[i,j,t] + 16u[i+1,j,t] - u[i+2,j,t]}{12h^2} + O(h^4)$$

The Finite Difference Method

- PDE becomes system of algebraic equations for the grid values $u[i,j,t]$
- Linear PDE becomes system of linear equations for the grid values $u[i,j,t]$
- Simple stability rule: *Information must not travel more than one grid cell in one time step*

$$c\Delta t < h \quad \rightarrow \quad \Delta t < h / c$$

Explicit Schemes – Linear Scalar Advection

- upwind (c>0)

$$\frac{u[t+1,i] - u[t,i]}{\Delta t} + c\frac{u[t,i] - u[t,i-1]}{\Delta x} = 0 \qquad u[t+1,i] = u[t,i] - c\frac{\Delta t}{\Delta x}(u[t,i] - u[t,i-1])$$

- downwind (c>0)

$$\frac{u[t+1,i] - u[t,i]}{\Delta t} + c\frac{u[t,i+1] - u[t,i]}{\Delta x} = 0 \qquad u[t+1,i] = u[t,i] - c\frac{\Delta t}{\Delta x}(u[t,i+1] - u[t,i])$$

- centered (Forward time centered space FTCS)

$$\frac{u[t+1,i] - u[t,i]}{\Delta t} + c\frac{u[t,i+1] - u[t,i-1]}{2\Delta x} = 0 \qquad u[t+1,i] = u[t,i] - c\frac{\Delta t}{2\Delta x}(u[t,i+1] - u[t,i-1])$$

- Leap-frog

$$\frac{u[t+1,i] - u[t-1,i]}{2\Delta t} + c\frac{u[t,i+1] - u[t,i-1]}{2\Delta x} = 0 \qquad u[t+1,i] = u[t-1,i] - c\frac{\Delta t}{\Delta x}(u[t,i+1] - u[t,i-1])$$

Explicit Schemes – Linear Scalar Advection

- Lax-Wendroff

$$u[t+1,i] = u[t,i] - c\frac{\Delta t}{2\Delta x}(u[t,i+1] - u[t,i-1]) + \frac{1}{2}\left(c\frac{\Delta t}{\Delta x}\right)^2 (u[t,i+1] - 2u[t,i] + u[t,i-1])$$

- Beam-Warming (c>0)

$$u[t+1,i] = u[t,i] - c\frac{\Delta t}{2\Delta x}(3u[t,i] - 4u[t,i-1] + u[t,i-2]) + \frac{1}{2}\left(c\frac{\Delta t}{\Delta x}\right)^2 (u[t,i] - 2u[t,i-1] + u[t,i-2])$$

- Lax-Friedrich

$$u[t+1,i] = \frac{1}{2}(u[t,i+1] - u[t,i-1]) - c\frac{\Delta t}{2\Delta x}(u[t,i+1] - u[t,i-1])$$

- CFL number (Courant-Friedrichs-Lewy), important in stability analysis

$$s = c\frac{\Delta t}{\Delta x}$$

Outline

Advection in 2D and 3D

- For 1D advection we had $T_t = -cT_x$
- In 2D and 3D, speed $\mathbf{c}$ is a vector with direction $\mathbf{n}_c$ and length c
- We have the spatial derivative of T in direction $\mathbf{n}_c$

$$\frac{\partial T}{\partial \mathbf{n}_c} = \mathbf{n}_c \cdot \nabla T, \qquad \text{in 2D}: \nabla T = \begin{bmatrix} T_x \\ T_y \end{bmatrix}, \quad \nabla = \begin{bmatrix} \partial / \partial x \\ \partial / \partial y \end{bmatrix}$$

Nabla operator

- Thus, we have $T_t = -c(\mathbf{n}_c \cdot \nabla T)$
- Since $\mathbf{c} = c\mathbf{n}_c$ the general advection PDE reads

$$T_t = -\mathbf{c} \cdot \nabla T$$

Finite Differences Solution

- The update rule in 2D

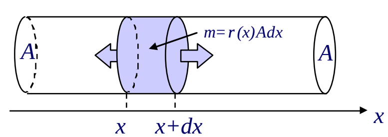
$$T^{t+1}[i,j] = T^t[i,j] - \Delta t \left(c_x \frac{T[i,j]-T[i-1,j]}{h} + c_y \frac{T[i,j]-T[i,j-1]}{h} \right)$$

- Make the wind velocity a function of the location

$$T_t = -\mathbf{c}(x,y) \cdot \nabla T$$

- T is advected by a general wind field
- Use velocity array, replace c_x by $c_x[i,j]$ and c_y by $c_y[i,j]$
- PDE still first order and linear

Diffusion in 1D



- The density $r(x)$ describes the concentration of a substance in a tube with cross section A.
- Conduction law: *The mass flow through an area A (flux) is proportional to the gradient of the density normal to A:*

$$\frac{dm}{dtA} = \frac{dr(x)Adx}{dtA} = k\,r_x\big|_{x+dx} - k\,r_x\big|_x$$

Conductivity

$$r_t = k\frac{r_x\big|_{x+dx} - r_x\big|_x}{dx} = kr_{xx} \quad \Rightarrow \quad r_t = kr_{xx}$$

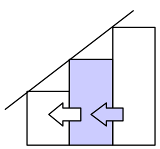
Diffusion in 3D

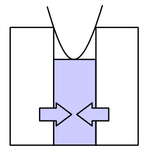
- The diffusion PDE in 3D

Laplace operator

$$r_t = k(r_{xx} + r_{yy} + r_{zz}) = k\nabla^2 r = k\,\Delta r$$

- Intuition for second spatial derivatives
- Constant concentration gradient $r(x)=a+bx$ has no effect


Flux but no change


Flux and positive change

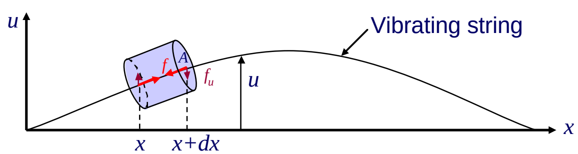
Finite Differences Solution

- The update rule in 2D

$$r^{t+1}[i,j] = r^{t}[i,j] + \Delta t\, k \frac{r[i+1,j]+r[i-1,j]+r[i,j+1]+r[i,j-1]-4r[i,j]}{h^2}$$

- Intuition:
  If the average concentration of the neighbor cells
  is larger than the concentration of the cell,
  its concentration increases and vice versa.

The 1D Wave Equation



- Function $u(x)$ is displacement of the string normal to x-axis
- Assuming small displacements and constant stress s
- Force acting normal to cross section A is $f = sA$
- Component in u-direction $f_u \approx sAu_x$
- Newton's 2nd law for an infinitesimal segment

$$(rAdx)u_{tt} = sAu_x\big|_{x+dx} - sAu_x\big|_x \Rightarrow r\,u_{tt} = s\,u_{xx}$$

Analytical Solution

- For the string we have $r\,u_{tt} = s\,u_{xx}$
- The standard form is $u_{tt} = c^2 u_{xx}$ where $c = \sqrt{\frac{s}{?}}$
- with the analytical solution
  $a \cdot f(x+ct) + b \cdot f(x-ct)$
- so c is the speed with which waves travel

2D Wave Equation

- The wave equation in 2D

$$u_{tt} = c^2(u_{xx} + u_{yy}) = c^2\,\nabla^2 u = c^2\,\Delta u$$

- Waves propagate with velocity c
- For the vibrating string we have $c^2 = s/r$

Finite Differences Solution

- The update rule in 2D

$$v^{t+1}[i,j] = v^t[i,j] + \Delta t\, c^2 \frac{u[i+1,j]+u[i-1,j]+u[i,j+1]+u[i,j-1]-4u[i,j]}{h^2}$$

$$u^{t+1}[i,j] = u^t[i,j] + \Delta t\, v^{t+1}[i,j]$$

- Very simple to implement
- Yields cool water surface animation
- Boundary conditions assuming $i \in (1,..,n)$
  - Periodic: $u[0,j] = u[n,j], \quad u[n+1,j] = u[1,j]$
  - Mirror: $u[0,j] = u[1,j], \quad u[n+1,j] = u[n,j]$
- Analog for j

Demo

- Combination of wave equation, diffusion, advection (1D)

$u_{tt} = -c_w^2 u_{xx}$ wave equation

$u_t = c_d u_{xx}$ diffusion

$u_t = -c_a u_x$ advection

- Explicit Leap-frog integration

$$v^{t+1} = v^t + \Delta t c_w^2 \nabla^2 u^t$$

$$u^{t+1} = u^t + \Delta t\left(v^{t+1} + c_d \nabla^2 u^t - c_a \nabla u^t\right)$$

Literature

- Alan Jeffrey, „Applied Partial Differential Equations – An Introduction," Academic Press, Amsterdam, ISBN 0-12-382252-1
- John D. Anderson, „Computational Fluid Dynamics – The Basics with Applications," McGraw-Hill Inc., New York, ISBN 0-07-001685-2