高等数值算法与应用（十二）

Advanced Numerical Algorithms & Applications

计算机科学与技术系 喻文健

Today

- Summary of Lecture 11
- Introduction to Partial Differential Equation
  - Preliminaries of Partial Differential Equation
  - Time-Dependent Problems
  - Time-Independent Problems
- Matlab Topics

Lecture 11 – ODE-BVP

Boundary Value Problem

$$y' = f(t, y), \quad a < t < b,$$
$$g(y(a), y(b)) = o,$$

- Separated, linear boundary condition

component of g involves solution values only at a or at b

$$B_a y(a) + B_b y(b) = c$$

Linear BVP

解的存在、唯一性

一般地，不一定成立

- 线性BVP问题解的存在、唯一性
- Mode solution: $y' = A(t)y$ with initial condition $y(a) = e_i$
- 充分必要条件：$Q \equiv B_a Y(a) + B_b Y(b)$ 非奇异
- 敏感性、稳定性　　解的不稳定性受边界条件限制

线性**BVP**问题可推导出条件数，表明解的敏感程度

Numerical methods

- Shooting　　多次**time integration**；结合非线性方程求解方法
- Multiple shooting　　分多个子区间，增加问题维度，提高稳定性
- Finite difference　　差分近似微分、得到离散点处的近似解

Lecture 11 – ODE-BVP

Numerical methods (cont'd)

- 函数空间逼近 $u(t) \approx v(t, x) = \sum_{i=1}^{n} x_i \phi_i(t)$ （x_i：确定系数；$\phi_i(t)$：选基函数）
- Collocation：在若干配置点处满足**ODE**和边界条件
- 最小二乘：最小化剩余函数的范数 $\int_a^b r(t,x)^2\,dt$　　$u'' = f(t)$

$$r(t,x) = \sum_{i=1}^{n} x_i \phi_i''(t) - f(t) \Rightarrow Ax = b \qquad a_{ij} = \int_a^b \phi_j''(t)\phi_i''(t)\,dt$$

- 加权余量法 $\int_a^b r(t,x) w_i(t)\,dt = 0$　　and $b_i = \int_a^b f(t)\phi_i''(t)\,dt$

$$a_{ij} = \int_a^b \phi_j''(t) w_i(t)\,dt \quad \text{and} \quad b_i = \int_a^b f(t) w_i(t)\,dt.$$

- Galerkin: $w_i = \phi_i$, 分部积分变换降低求导阶数　　$a_{ij} = -\int_a^b \phi_j'(t)\phi_i'(t)\,dt$
- 基函数的选择: 谱方法, **FEM**

global support（谱方法）　**localized support**（FEM）

相关的特征值问题

$$u'' = \lambda f(t, u, u'), \quad a < t < b, \qquad u(a) = \alpha, \quad u(b) = \beta.$$

- 将**ODE**离散成代数方程，转化为代数方程特征值问题

Preliminaries of Partial Differential Equation

Wenjian Yu

Partial Differential Equations

Partial differential equations (PDEs) involve partial derivatives with respect to more than one independent variable

Independent variables typically include one or more space dimensions and possibly time dimension as well

More dimensions complicate problem formulation: can have pure initial value problem, pure boundary value problem, or mixture

Equation and boundary data may be defined over irregular domain in space

两方面的复杂性

我们就一些简单问题讨论基本概念和方法

Continuous phenomena modeled by PDE

- Maxwell's equations in electro-magnetics
  - 静电场的高斯定律

$$\oiint_S \varepsilon \boldsymbol{E} \cdot d\boldsymbol{s} = \iiint_V \rho dv \Rightarrow \nabla \cdot (\varepsilon \boldsymbol{E}) = \rho$$

$$\begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{bmatrix} \cdot \begin{bmatrix} E_x \\ E_y \\ E_z \end{bmatrix} = \frac{\rho}{\varepsilon}$$

  - 安培定律 (电流、变化的电场产生磁场)

$$\nabla \times \boldsymbol{H} = \varepsilon \partial \boldsymbol{E} / \partial t + \boldsymbol{J}$$

  - 法拉弟电磁感应定律

$$\oint_l \boldsymbol{E} \cdot dl = -\frac{\partial(\iint_S \mu \boldsymbol{H} \cdot ds)}{\partial t} \Rightarrow \nabla \times \boldsymbol{E} = -\mu \frac{\partial \boldsymbol{H}}{\partial t}$$

$$\begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{bmatrix} \times \begin{bmatrix} E_x \\ E_y \\ E_z \end{bmatrix} = -\mu \begin{bmatrix} \frac{\partial H_x}{\partial t} \\ \frac{\partial H_y}{\partial t} \\ \frac{\partial H_z}{\partial t} \end{bmatrix}$$

  - 磁场是无源场

$$\nabla \cdot (\mu \boldsymbol{H}) = 0$$

均为$\boldsymbol{x}$, $\boldsymbol{y}$, $\boldsymbol{z}$, $\boldsymbol{t}$的函数

- 其他：流体的Navier-Stokes、弹性力学中linear elasticity、量子力学中Schrodinger's、广义相对论中Einstein's

Partial Differential Equations, cont.

For simplicity, we will deal only with single PDEs (as opposed to systems of several PDEs) with only two independent variables, either

- two space variables, denoted by x and y, or
- one space variable and one time variable, denoted by x and t, respectively

类似于**ODE**
边值问题

类似于**ODE**
初值问题

Partial derivatives with respect to independent variables denoted by subscripts:

记号说明：

- $u_t = \partial u / \partial t$
- $u_{xy} = \partial^2 u / \partial x \partial y$, etc.

求解未知函数 $\boldsymbol{u}$
在定义域范围内满足给定**PDE**
同时满足初始、和边界条件

Example: Advection Equation

对流方程，或单向波方程

Advection equation:

$$u_t = -c\,u_x,$$

where c is nonzero constant

Unique solution determined by initial condition

$$u(0, x) = u_0(x), \qquad -\infty < x < \infty,$$

纯初值问题

where u_0 is given function defined on $\mathbb{R}$

We seek solution $u(t, x)$ for $t \geq 0$ and all $x \in \mathbb{R}$

From chain rule, solution given by **验证?**

$$u(t, x) = u_0(x - c\,t),$$

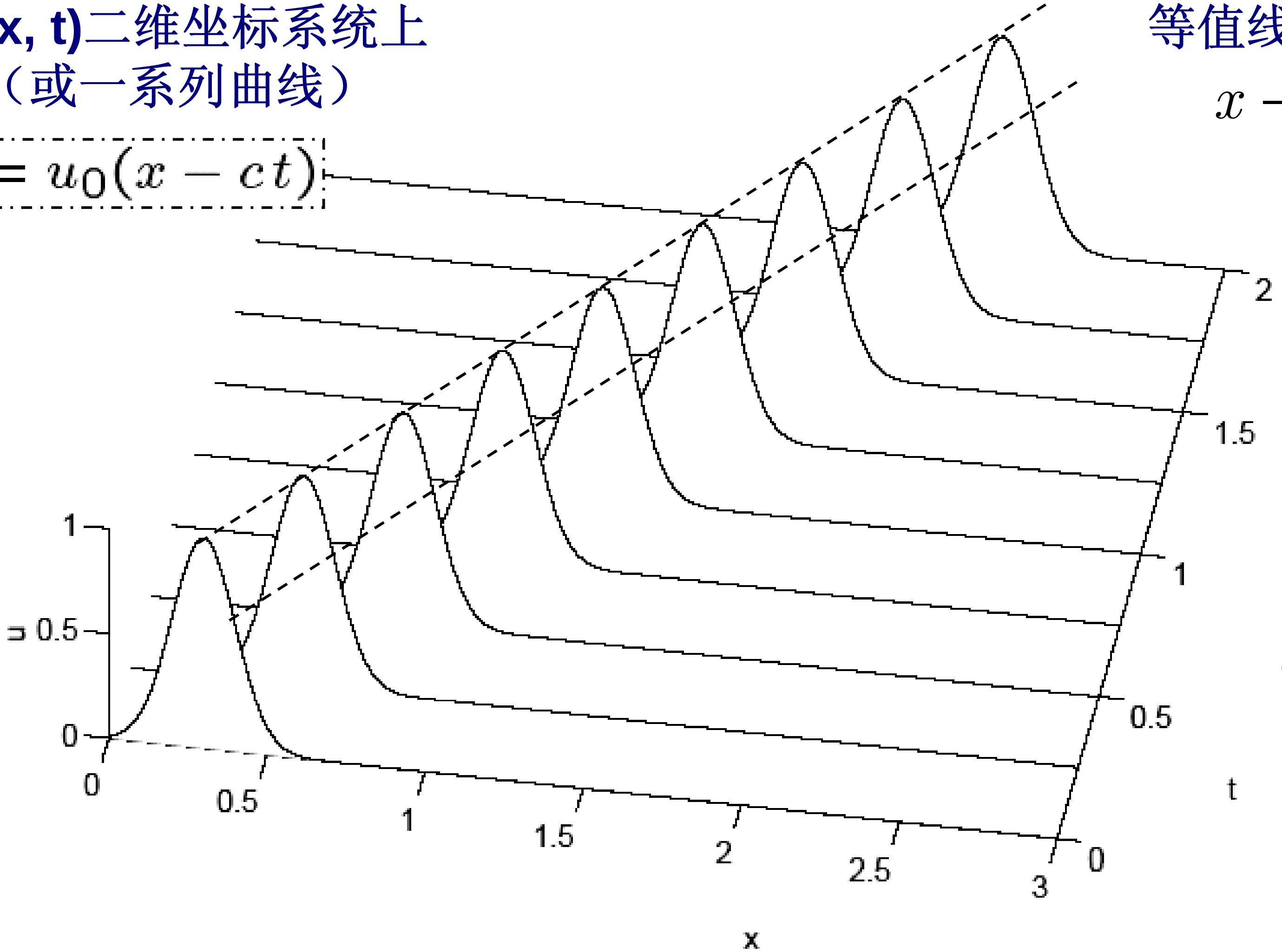
i.e., solution is initial function u_0 shifted by $c\,t$ to right if $c > 0$, or to left if $c < 0$

定义在**(x, t)**二维坐标系统上的曲面（或一系列曲线）

$$u(t,x) = u_0(x - ct)$$

等值线，等高线

$$x - ct = x_0$$



***c* =1**

Typical solution of advection equation $u_t = -c\,u_x$

Initial function u_0 is propagated to the right (or left) with velocity *c*

Characteristics

等值线

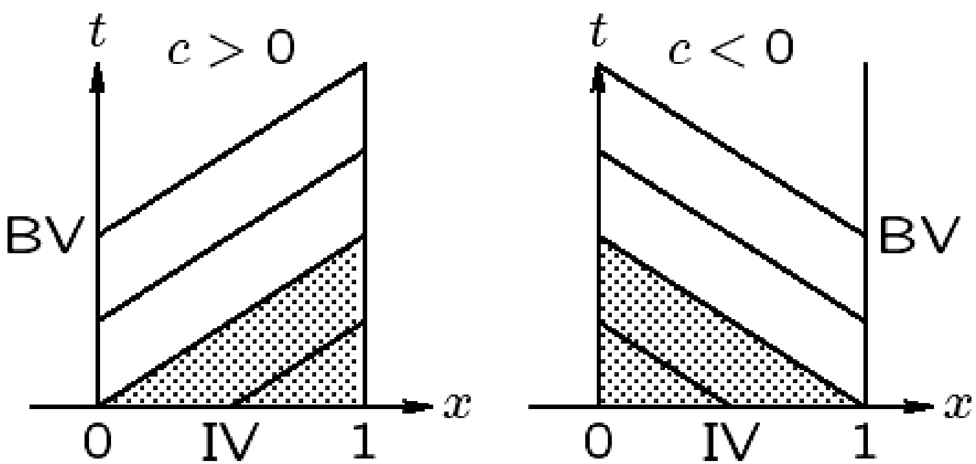
<u>Level curves</u> of solution to PDE are called *characteristics* 特征线

刻画依赖关系

Characteristics for advection equation, for example, are <u>straight lines of slope c</u>

理论上很重要，确定问题的可解条件

Characteristics determine where <u>boundary conditions</u> can or must be imposed for problem to be <u>well-posed</u> 例如，定义域为**0≤x ≤1, *t* ≥0** 的常系数对流方程



阴影区域的解由初值u_0决定，还需加边界条件确定其他区域的解

Classification of PDEs

Second-order linear PDEs of form (阶数为最高阶偏导数)

$$a u_{xx} + b u_{xy} + c u_{yy} + d u_x + e u_y + f u + g = 0$$ (考虑两个自变量)

are classified by value of *discriminant*, $b^2 - 4ac$,
判别式

通过变量代换转化为

双曲	$b^2 - 4ac > 0$:	*hyperbolic* (e.g., wave eqn)	$u_{tt} - u_{xx} + \cdots = 0$
抛物	$b^2 - 4ac = 0$:	*parabolic* (e.g., heat eqn)	$u_t - u_{xx} + \cdots = 0$
椭圆	$b^2 - 4ac < 0$:	*elliptic* (e.g., Laplace eqn)	$u_{xx} + u_{yy} + \cdots = 0$

回忆平面二次曲线方程:

$$ax^2 + bxy + cy^2 + dx + ey + f = 0$$

通过变量代换

$$(x + \frac{b}{2a} y)^2 + \frac{4ac - b^2}{4a^2} y^2 + \frac{d}{a}(x + \frac{b}{2a} y) + \alpha y + \beta = 0$$

$$\frac{(x - x_0)^2}{a^2} - \frac{(y - y_0)^2}{b^2} = 1.$$

$$(y - y_0)^2 = 4a(x - x_0).$$

$$\frac{(x - x_0)^2}{a^2} + \frac{(y - y_0)^2}{b^2} = 1.$$

Classification of PDEs, cont.

Classification of more general PDEs not so clean and simple, but roughly speaking:

特征线的概念对
前两种仍适用

双曲

- *Hyperbolic* PDEs describe time-dependent, conservative physical processes, such as convection, that *are not* evolving toward steady state

对流, 波动

抛物

耗散

- *Parabolic* PDEs describe time-dependent, dissipative physical processes, such as diffusion, that *are* evolving toward steady state

扩散

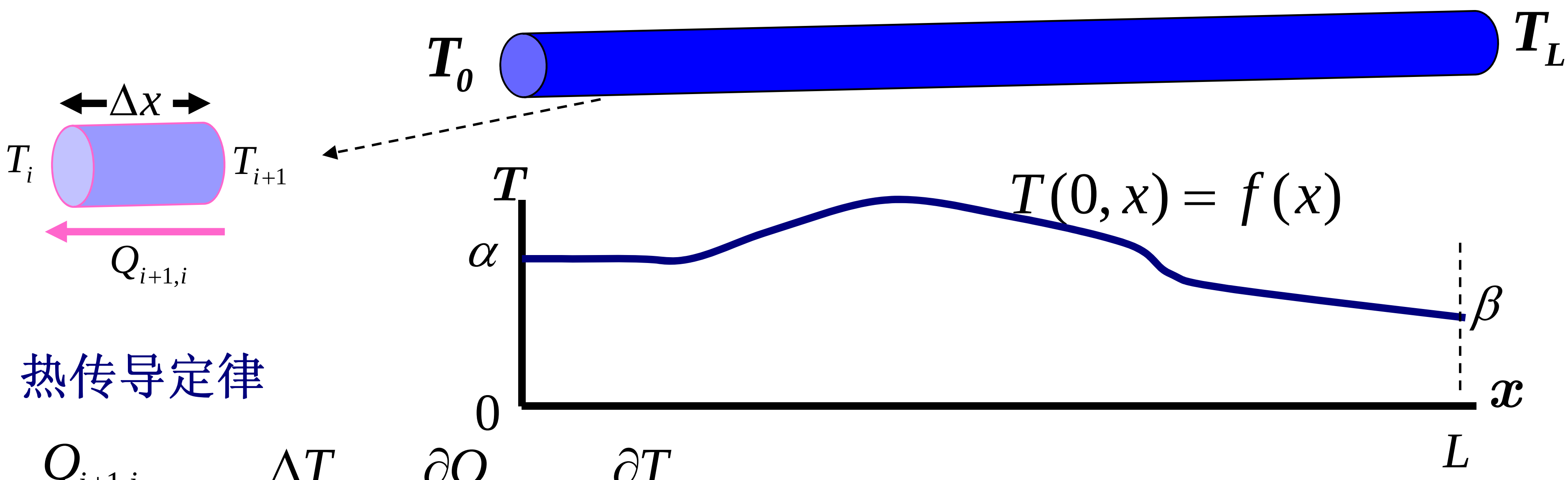
椭圆

- *Elliptic* PDEs describe processes that have already reached steady state, and hence are time-independent

更多与物理问题的联系，见**p. 387**

Example - Heat equation (抛物型)

假设不从空气散热，求时间t的温度分布



热传导定律

$$\frac{Q_{i+1,i}}{\Delta t} = kA\frac{\Delta T}{\Delta x} \Rightarrow \frac{\partial Q}{\partial t} = kA\frac{\partial T}{\partial x}$$

能量守恒定律

$$\frac{Q_{i+1,i} - Q_{i,i-1}}{\Delta t} - c\rho A\Delta x\frac{\Delta T}{\Delta t} = 0 \Rightarrow \frac{\partial}{\partial x}\left(\frac{\partial Q}{\partial t}\right)dx = c\rho A\frac{\partial T}{\partial t}dx$$

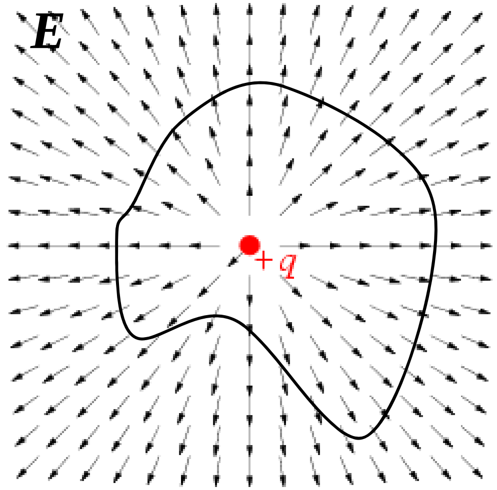
$$k\frac{\partial^2 T}{\partial x^2} = c\rho\frac{\partial T}{\partial t}$$

给定了初值、以及两端的边界条件

Example - Laplace equation (椭圆型)

静电场中的高斯定理：

$$\oiint_S \varepsilon \boldsymbol{E} \cdot d\boldsymbol{s} = \iiint_V \rho dv \Rightarrow \nabla \cdot (\varepsilon \boldsymbol{E}) = \rho$$



其中**S**是封闭曲面。由于静电场**E**是保守场，即沿任意闭合曲线积分为**0**，则可引入标量电位***u***，使得：

$$\boldsymbol{E} = -\nabla u$$

注意**Laplace**算符的不同用法和意义

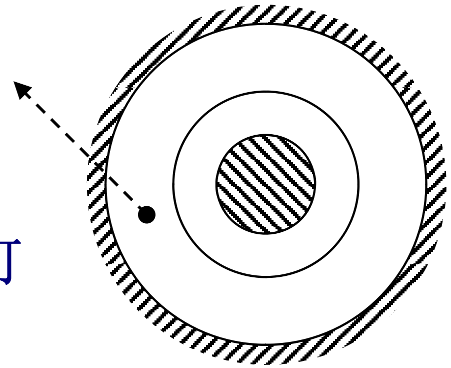
$$\nabla^2 u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} = -\frac{\rho}{\varepsilon}$$

ρ是空间点**(x,y,z)**处的电荷密度

Poisson方程，若右端为零则称***Laplace***方程。

2D example：

$$u_{xx} + u_{yy} = 0$$



给定区域的边界条件后，可求电势***u***的分布

总结——PDE基本概念

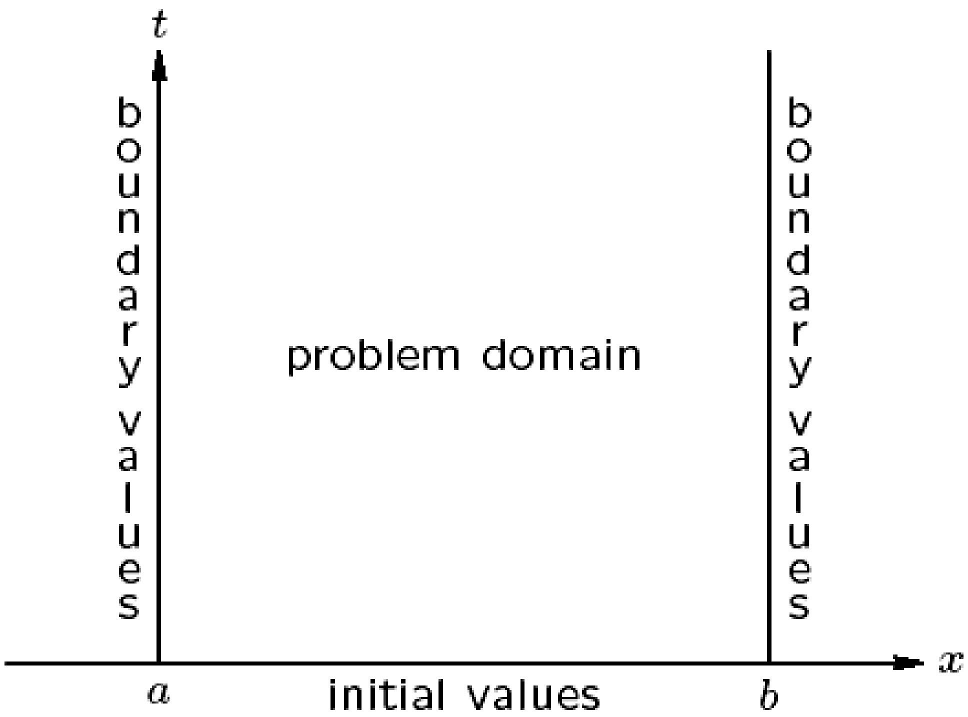
- PDE问题求解的复杂性
  - 变量多导致问题类型多：初值、边值、混合
  - 高维的复杂定义域形状
- PDE问题描述各种自然现象
- 仅讨论含两独立变量的PDE
  - 纯边值问题，不含时间变量
  - 初边值问题，含时间变量
- 单向波方程——特征线的概念　如何加边界条件使问题可解
- PDE的阶数、二阶PDE分类（双曲、抛物、椭圆）
- 二阶PDE举例——热方程、静电场方程

Time-Dependent Problems

Wenjian Yu

Time-Dependent Problems

Time-dependent PDEs usually involve both initial values and boundary values



抛物型**(heat)**
双曲型**(wave)**

数值求解方法可分为两类：

- 只对空间进行离散的半离散方法（有限差分、**collocation**方法）
- 全离散方法（显格式有限差分、隐格式有限差分）

Semidiscrete Methods

One way to solve time-dependent PDE numerically is to discretize in space, but leave time variable continuous

Result is system of ODEs that can then be solved by methods previously discussed

ODE初值问题

For example, consider heat equation

$$u_t = c\, u_{xx}, \qquad 0 \le x \le 1, \qquad t \ge 0,$$

with initial condition

$$u(0, x) = f(x), \qquad 0 \le x \le 1,$$

and boundary conditions

$$u(t, 0) = 0, \qquad u(t, 1) = 0, \qquad t \ge 0$$

Semidiscrete Finite Difference Method

If we introduce spatial mesh points $x_i = i\Delta x$, $i = 0, \ldots, n+1$, where $\Delta x = 1/(n+1)$ and replace derivative u_{xx} by finite difference approximation

$$u_{xx}(t, x_i) \approx \frac{u(t, x_{i+1}) - 2u(t, x_i) + u(t, x_{i-1})}{(\Delta x)^2},$$

then we get system of ODEs

$$y_i'(t) = \frac{c}{(\Delta x)^2}\left(y_{i+1}(t) - 2y_i(t) + y_{i-1}(t)\right),$$

$i = 1, \ldots, n$, where $y_i(t) \approx u(t, x_i)$

From boundary conditions, $y_0(t)$ and $y_{n+1}(t)$ are identically zero, and from initial conditions, $y_i(0) = f(x_i)$, $i = 1, \ldots, n$

因此，得到齐次线性**ODE**

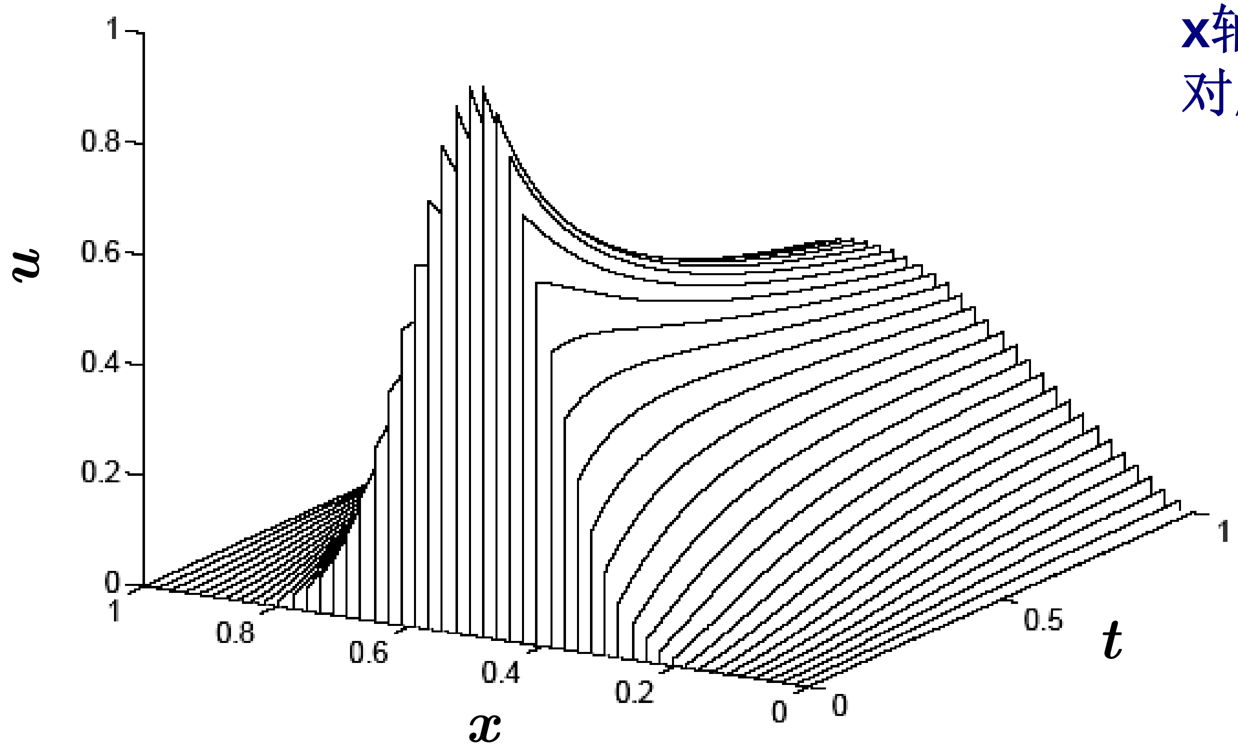
$$\boldsymbol{y}' = \boldsymbol{A}\boldsymbol{y}$$

Can therefore use ODE method to solve initial value problem for this system

Method of Lines

Approach just described is called *method of lines*

MOL computes cross-sections of solution surface over space-time plane along series of lines, each parallel to time axis and corresponding to one of discrete spatial mesh points



x轴上的离散点，对应的时变曲线

Stiffness

关于Gerischgorin圆盘定理：

设矩阵 $\boldsymbol{A} = \left(a_{ij}\right)_{n\times n}$

，定义复数平面的n个圆盘区域

$$D_i = \left|z - a_{ii}\right| \le \sum_{j=1, j\ne i}^{n} \left|a_{ij}\right|$$

则矩阵A的特征值

$$\lambda_i \in \mathrm{U} D_i$$

Semidiscrete system of ODEs just derived can be written in matrix form

$$y' = \frac{c}{(\Delta x)^2} \begin{bmatrix} -2 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & \cdots & 0 \\ 0 & 1 & -2 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & -2 \end{bmatrix} y = \boldsymbol{A} y$$

Jacobian matrix $\boldsymbol{A}$ of this system has eigenvalues between $-4c/(\Delta x)^2$ and 0, which makes ODE very stiff as spatial mesh size Δx becomes small

This stiffness, which is typical of ODEs derived from PDEs in this manner, must be taken into account in choosing ODE method for solving semidiscrete system

选择解ODE算法时要考虑这种stiffness

Spatial discretization to convert PDE into system of ODEs can also be done by spectral or finite element approach

Approximate solution is linear combination of basis functions, but now coefficients are time dependent

对给定的时间t, u(t, x)变成关于x的函数，再将其表示为一组基函数的线性组合

Thus, we seek solution of form

$$u(t, x) \approx v(t, x, \boldsymbol{\alpha}(t)) = \sum_{j=1}^{n} \alpha_j(t)\phi_j(x),$$

where $\phi_j(x)$ are suitably chosen basis functions

If we use collocation, then we substitute this approximation into PDE and require that equation be satisfied exactly at discrete set of points x_i

Semidiscrete Collocation, continued

$u_t = c\,u_{xx}$

For heat equation, this yields system of ODEs

$$\sum_{j=1}^{n} \alpha_j'(t)\phi_j(x_i) = c\sum_{j=1}^{n} \alpha_j(t)\phi_j''(x_i),$$

whose solution is set of coefficient functions $\alpha_i(t)$ that determine approximate solution to PDE

Implicit form of this system is not explicit form required by standard ODE methods, so we define $n \times n$ matrices $\boldsymbol{M}$ and $\boldsymbol{N}$ by

并非通常讨论ODE问题时用的显格式

$$m_{ij} = \phi_j(x_i), \qquad n_{ij} = \phi_j''(x_i)$$

$$\boldsymbol{M}\boldsymbol{\alpha}'(t) = c\boldsymbol{N}\boldsymbol{\alpha}(t)$$

Semidiscrete Collocation, continued

Assuming M is nonsingular, we then obtain system of ODEs

$$\alpha'(t) = c M^{-1} N \alpha(t),$$

which is in form suitable for solution with standard ODE software (as usual, M need not be inverted explicitly, but merely used to solve linear systems)

Initial condition for ODE can be obtained by requiring that solution satisfy given initial condition for PDE at points x_i

Matrices involved in this method will be sparse if basis functions are "local," such as B-splines

Unlike finite difference method, spectral or finite element method does not produce approximate values of solution u directly, but rather it generates representation of approximate solution as linear combination of basis functions

并非直接得到u的近似值，而得到近似函数

Basis functions depend only on spatial variable, but coefficients of linear combination (given by solution to system of ODEs) are time dependent

Thus, for any given time t, corresponding linear combination of basis functions generates cross section of solution surface parallel to spatial axis

而不是平行于时间轴

As with finite difference methods, systems of ODEs arising from semidiscretization of PDE by spectral or finite element methods tend to be stiff

仍有stiff问题

Fully Discrete Methods

Fully discrete methods for PDEs discretize in both time and space dimensions 并非仅在空间上离散

In fully discrete finite difference method, we

- Replace continuous domain of equation by discrete mesh of points
- Replace derivatives in PDE by finite difference approximations 有限差分近似替代求导
- Seek numerical solution that is table of approximate values at selected points in space and time

In two dimensions (one space and one time), resulting approximate solution values represent points on solution surface over problem domain in space-time plane 求一系列离散点上的近似解

Fully Discrete Methods, continued

Accuracy of approximate solution depends on stepsizes in both space and time

Replacement of all partial derivatives by finite differences results in system of algebraic equations for unknown solution at discrete set of sample points

代数方程系统

System may be linear or nonlinear, depending on underlying PDE

线性或非线性

With initial-value problem, solution is obtained by starting with initial values along boundary of problem domain and marching forward in time step by step, generating successive rows in solution table

从初值函数，沿时间轴一步步得到后续的解

Time-stepping procedure may be explicit or implicit, depending on whether formula for solution values at next time step involves only past information

沿时间轴“推进”的公式可能是显格式或隐格式

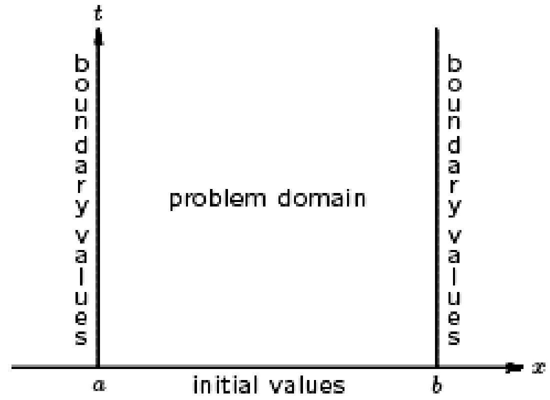
Example: Heat Equation

Consider heat equation

$$u_t = c\, u_{xx}, \qquad 0 \le x \le 1, \qquad t \ge 0,$$

with initial and boundary conditions

$$u(0,x) = f(x), \qquad u(t,0) = \alpha, \qquad u(t,1) = \beta$$



已知(定解)条件

Define spatial mesh points $x_i = i\Delta x$, $i = 0, 1, \ldots, n+1$, where $\Delta x = 1/(n+1)$, and temporal mesh points $t_k = k\Delta t$, for suitably chosen Δt

Let u_i^k denote approximate solution at (t_k, x_i)

If we replace u_t by forward difference in time and u_{xx} by centered difference in space, we get

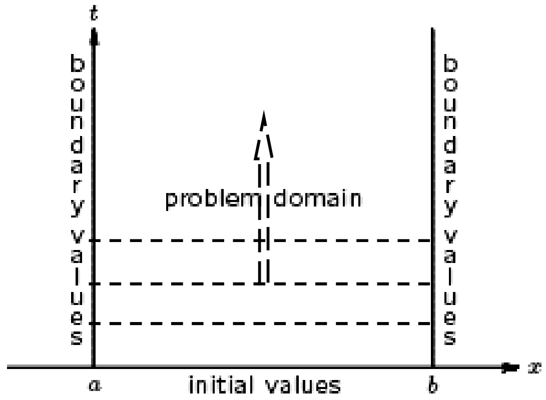
向前差分
中心差分

$$\frac{u_i^{k+1} - u_i^k}{\Delta t} = c\,\frac{u_{i+1}^k - 2u_i^k + u_{i-1}^k}{(\Delta x)^2},$$

or t_{k+1}时间点的函数值

$$u_i^{k+1} = u_i^k + c\,\frac{\Delta t}{(\Delta x)^2}\left(u_{i+1}^k - 2u_i^k + u_{i-1}^k\right), \; i = 1, \ldots, n$$

Heat Equation, continued



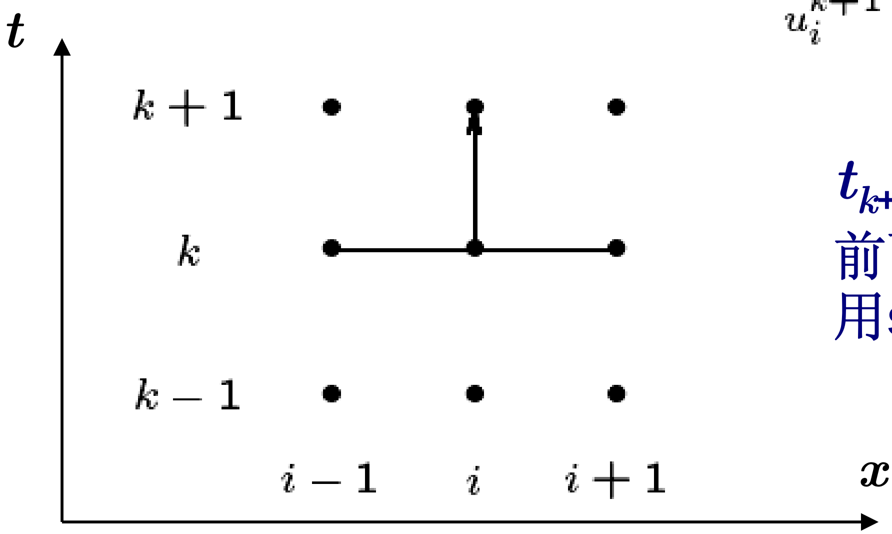
Boundary conditions give us $u_0^k = \alpha$ and $u_{n+1}^k = \beta$ for all k, and initial conditions provide starting values $u_i^0 = f(x_i)$, $i = 1, \ldots, n$

So we can march numerical solution forward in time using this *explicit* difference scheme

$$u_i^{k+1} = u_i^k + c\,\frac{\Delta t}{(\Delta x)^2}\left(u_{i+1}^k - 2u_i^k + u_{i-1}^k\right)$$

蜡纸标记

Pattern of mesh points, or *stencil*, involved at each level is shown below



t_{k+1}时间点的函数值如何通过前面时间点的值来计算？通常用**stencil**图直观表示

Local truncation error is $\mathcal{O}(\Delta t) + \mathcal{O}((\Delta x)^2)$, so scheme is first-order accurate in time and second-order accurate in space

不同于**ODE**中的定义，但也反映当前计算步的误差

Local truncation error

定义：将准确解带入差分方程，
计算方程的残差

$$\frac{u_i^{k+1}-u_i^k}{\Delta t}=c\frac{u_{i+1}^k-2u_i^k+u_{i-1}^k}{(\Delta x)^2},$$

$$\frac{u(t_{k+1},x_i)-u(t_k,x_i)}{\Delta t}-c\frac{u(t_k,x_{i+1})-2u(t_k,x_i)+u(t_k,x_{i-1})}{(\Delta x)^2}=?$$

$$\frac{u(t_{k+1},x_i)-u(t_k,x_i)}{\Delta t}=\frac{\partial u(t_k,x_i)}{\partial t}+O(\Delta t)$$

$$\frac{u(t_k,x_{i+1})-2u(t_k,x_i)+u(t_k,x_{i-1})}{(\Delta x)^2}=\frac{\partial^2 u(t_k,x_i)}{\partial x^2}+O((\Delta x)^2)$$

$$\frac{\partial u}{\partial t}=c\frac{\partial^2 u}{\partial x^2}$$

相减

$$\frac{u(t_{k+1},x_i)-u(t_k,x_i)}{\Delta t}-c\frac{u(t_k,x_{i+1})-2u(t_k,x_i)+u(t_k,x_{i-1})}{(\Delta x)^2}=\underline{O(\Delta t)+O((\Delta x)^2)}$$

Example: Wave Equation

Consider wave equation

$$u_{tt} = c\, u_{xx}, \qquad 0 \le x \le 1, \qquad t \ge 0,$$

with initial and boundary conditions

由于有对t的二阶偏导，
增加一个初始条件

$$u(0,x) = f(x), \qquad u_t(0,x) = g(x),$$

$$u(t,0) = \alpha, \qquad u(t,1) = \beta$$

With mesh points defined as before, using centered difference formulas for both u_{tt} and u_{xx} gives finite difference scheme

$$\frac{u_i^{k+1} - 2u_i^k + u_i^{k-1}}{(\Delta t)^2} = c\,\frac{u_{i+1}^k - 2u_i^k + u_{i-1}^k}{(\Delta x)^2},$$

局部截断误差

$$O((\Delta t)^2) + O((\Delta x)^2)$$

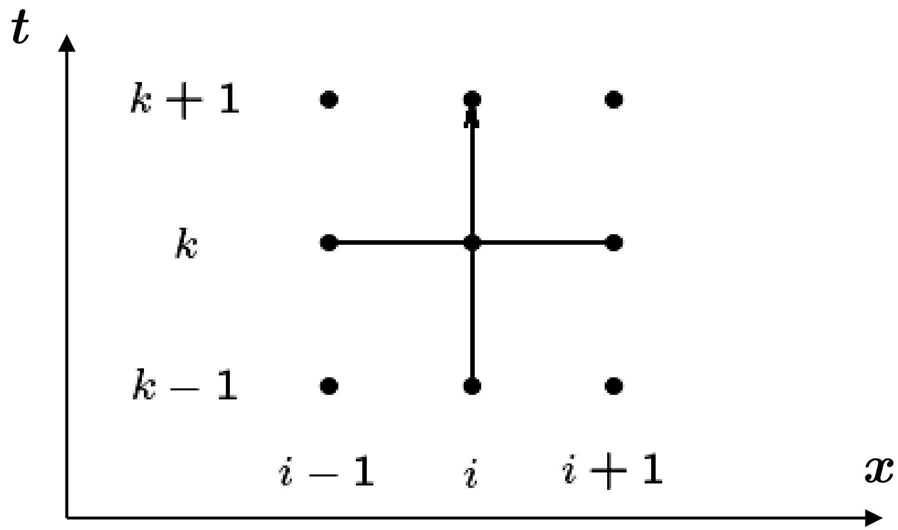
or

$$u_i^{k+1} = 2u_i^k - u_i^{k-1} + c\left(\frac{\Delta t}{\Delta x}\right)^2 \left(u_{i+1}^k - 2u_i^k + u_{i-1}^k\right),$$

$$i = 1, \ldots, n$$

Wave Equation, continued

Stencil for this scheme is shown below



Using data at two levels in time requires additional storage

Also need u_i^0 and u_i^1 to get started, which can be obtained from initial conditions

$$u_i^0 = f(x_i), \qquad u_i^1 = f(x_i) + (\Delta t)g(x_i),$$ 用向前差分近似

where latter uses forward difference approximation to initial condition $u_t(0, x) = g(x)$

Stability

Unlike Method of Lines, where time step is chosen automatically by ODE solver, user must choose time step Δt in fully discrete method, taking into account both accuracy and stability requirements

Semidiscrete法中，时间步长选择交给ODE solver去做

时间上用向前差分格式 $u_t = cu_{xx}$

For example, fully discrete scheme for heat equation is simply Euler's method applied to semidiscrete system of ODEs for heat equation given previously

完全等价于**MOL**得到的方程用前向**Euler**解

$$y' = \frac{c}{(\Delta x)^2}\begin{bmatrix} -2 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & \cdots & 0 \\ 0 & 1 & -2 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & -2 \end{bmatrix} y$$

We saw that Jacobian matrix of semidiscrete system has eigenvalues between $-4c/(\Delta x)^2$ and 0, so stability region for Euler's method requires time step to satisfy

$$\Delta t \le \frac{2}{|\lambda|}$$

$$\Delta t \le \frac{(\Delta x)^2}{2\,c}$$

This severe restriction on time step makes this explicit method relatively inefficient compared to implicit methods we will see next

For ODEs we saw that implicit methods are stable for much greater range of stepsizes, and same is true of implicit methods for PDEs

时间上用向后差分格式

Applying backward Euler method to semidiscrete system for heat equation gives *implicit* finite difference scheme

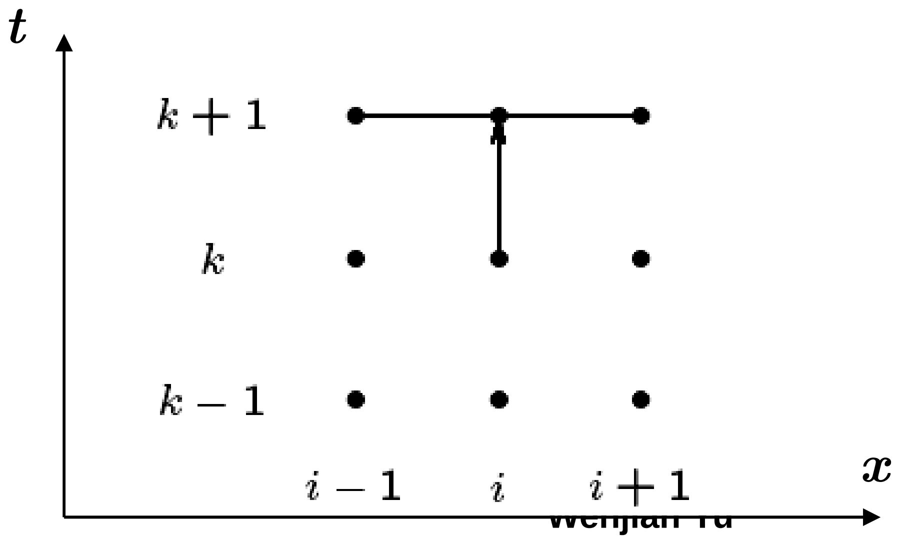
完全等价于**MOL**得到的方程用向后**Euler**解

$$u_i^{k+1} = u_i^k + c\frac{\Delta t}{(\Delta x)^2}\left(u_{i+1}^{k+1} - 2u_i^{k+1} + u_{i-1}^{k+1}\right),$$

$i = 1, \ldots, n$

无条件稳定

Stencil for this scheme is shown below



Implicit Finite Difference Methods, cont.

This scheme inherits unconditional stability of backward Euler method, which means there is no stability restriction on relative sizes of Δt and Δx

However, first-order accuracy in time still limits time step severely

时间离散：
梯形公式

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

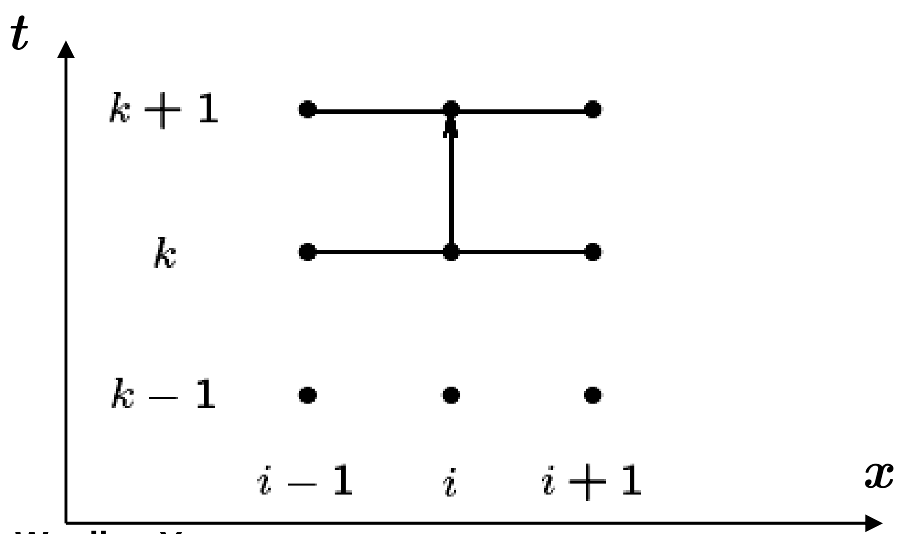
Applying trapezoid method to semidiscrete system of ODEs for heat equation yields implicit *Crank-Nicolson* method

$$u_i^{k+1}=u_i^k+c\frac{\Delta t}{2(\Delta x)^2}\left(u_{i+1}^{k+1}-2u_i^{k+1}+u_{i-1}^{k+1}+u_{i+1}^k-2u_i^k+u_{i-1}^k\right),\quad i=1,\ldots,n,$$

which is unconditionally stable and second-order accurate in time

二阶准确度：
2阶截断误差

Stencil for this scheme is shown below



Implicit Finite Difference Methods, cont.

Much greater stability of implicit finite difference methods enables them to take much larger time steps than explicit methods, but they require more work per step, since system of equations must be solved at each step

For both backward Euler and Crank-Nicolson methods for heat equation in one space dimension, this linear system is tridiagonal, and thus both work and storage required are modest

适度的

In higher dimensions, matrix of linear system does not have such simple form, but it is still very sparse, with nonzeros in regular pattern

例:半离散方法,向后Euler

$$y' = Ay$$

$$y^{k+1} = y^k + \Delta t \cdot A y^{k+1}$$

⇩

$$(I - \Delta t \cdot A) y^{k+1} = y^k$$

稀疏矩阵

二维空间：五对角（二阶差分公式中涉及**5**个点）
三维空间：七对角（二阶差分公式中涉及**7**个点）

Convergence

In order for approximate solution to converge to true solution of PDE as stepsizes in time and space jointly go to zero, two conditions must hold:

收敛性：
时间、空间步长→ 0，则数值解→ 准确解

- *Consistency*: local truncation error goes to zero
- *Stability*: approximate solution at any fixed time t remains bounded

一致性, 相容性：局部截断误差→ 0

稳定性：同解ODE-IVP问题方法的稳定性

Lax Equivalence Theorem says that for well-posed linear PDE, consistency and stability are together necessary and sufficient for convergence

充分必要条件

Stability

Consistency is usually fairly easy verified using Taylor series expansion

分析截断误差，阶数p≥1即保证相容

Stability is more challenging, and several methods are available:

稳定性分析比较困难

- *Matrix method*, based on location of eigenvalues of matrix representation of difference scheme, as we saw with Euler's method

$$y' = Ay$$

$$y^k = (I + \Delta t \cdot A)^k \cdot y^0$$

$$\frac{-4c}{(\Delta x)^2} \le Eig(A) \le 0$$

$$1 - \frac{4c \cdot \Delta t}{(\Delta x)^2} \le Eig(I + \Delta t \cdot A) \le 1$$

$$\Delta t \le \frac{(\Delta x)^2}{2c}$$

- *Fourier method*, in which complex exponential representation of solution error is substituted into difference equation and analyzed for growth or decay

- *Domains of dependence*, in which domains of dependence of PDE and difference scheme are compared

依赖区域

显格式、双曲方程

CFL Condition

依赖区域

Domain of dependence of PDE is portion of problem domain that influences solution at given point, which depends on characteristics of PDE

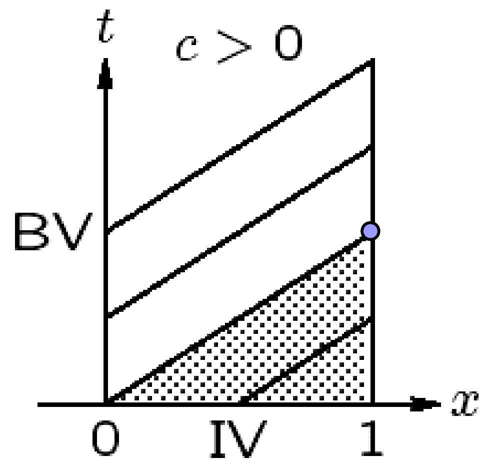
特征线

差分格式

Domain of dependence of difference scheme is set of all other mesh points that affect approximate solution at given mesh point

CFL Condition: necessary condition for explicit finite difference scheme for hyperbolic PDE to be stable is that for each mesh point domain of dependence of PDE must lie *within* domain of dependence of finite difference scheme

$u_t = -c\,u_x$



显格式双曲型
PDE的依赖域

<

有限差分格式的
依赖域

显格式！

Example: Wave Equation

$$u_{tt} = c\,u_{xx}$$

Consider explicit finite difference scheme for wave equation given previously

基本解函数

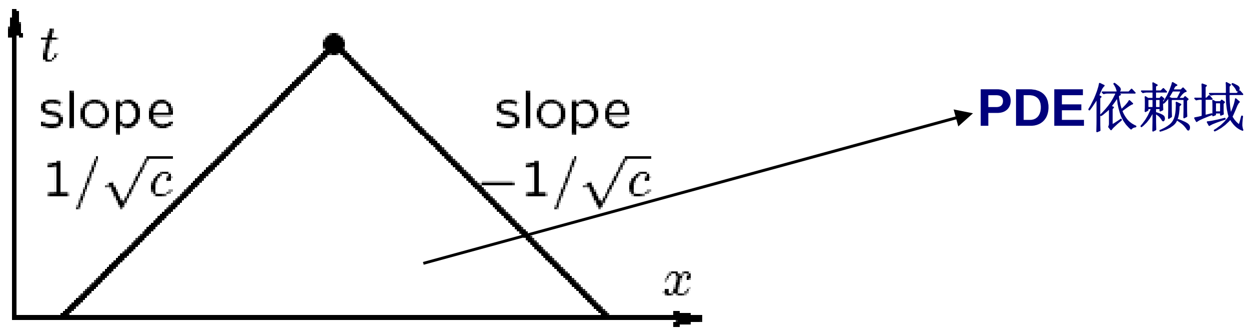
$\psi(x+\sqrt{c}t)$

$\psi(x-\sqrt{c}t)$

Characteristics of wave equation are straight lines in (t, x) plane along which either $x + \sqrt{c}\,t$ or $x - \sqrt{c}\,t$ is constant

两个特征线

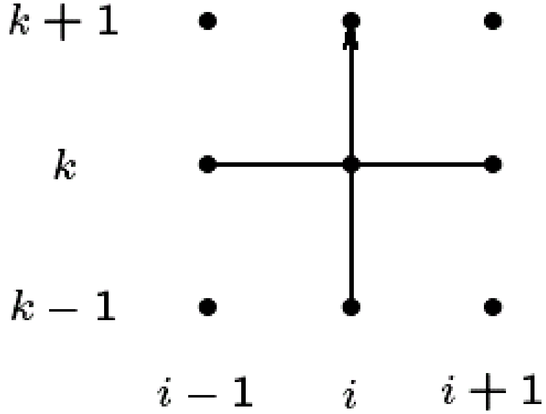
Domain of dependence for wave equation for given point is triangle with apex at given point and with sides of slope $1/\sqrt{c}$ and $-1/\sqrt{c}$



CFL condition implies step sizes must satisfy

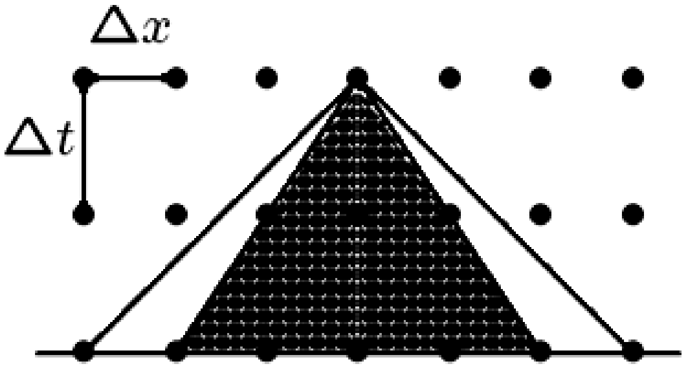
$$\Delta t \le \frac{\Delta x}{\sqrt{c}}$$

Stencil:



Slop:

$$\frac{\Delta t}{\Delta x} > \frac{1}{\sqrt{c}}$$



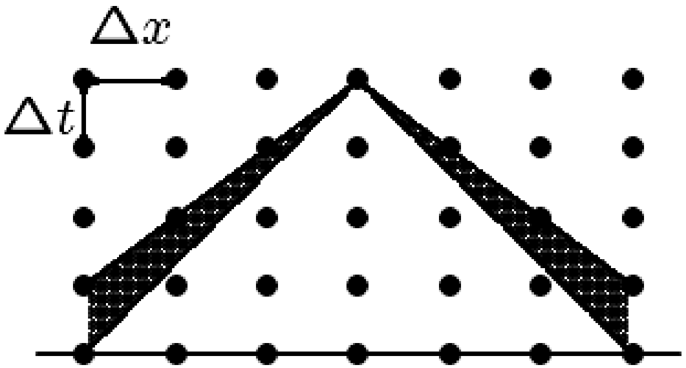
Unstable finite difference scheme

阴影：差分格式的依赖域

CFL条件：差分格式依赖域 > PDE依赖域则稳定

Slop:

$$\frac{\Delta t}{\Delta x} < \frac{1}{\sqrt{c}}$$



Stable finite difference scheme

总结——时变初边值问题的求解

- 半离散方法（离散空间变量）
  - 有限差分法（Method of Line）
  - Collocation method（谱方法、有限元）
- 全离散方法（沿时间轴推进）
  - 热方程（抛物型）
  - 波动方程（双曲型）
  - 局部截断误差、稳定性
  - 隐格式——时间后向欧拉、梯形公式
  - 差分方法的收敛性：一致性、稳定性（Lax等价性定理）
  - 稳定性分析——矩阵法、依赖域分析（针对显格式双曲型方程的CFL条件）

差分格式的**stencil**图

Time-Independent Problems

We next consider time-independent, elliptic PDEs in two space dimensions, such as *Helmholtz* equation

椭圆型

$$u_{xx} + u_{yy} + \lambda u = f(x, y)$$

稳态问题、和时间无关

Important special cases:

Poisson equation: $\lambda = 0$

Laplace equation: $\lambda = 0$ and $f = 0$

For simplicity, we will consider this equation on unit square

模型问题

Numerous possibilities for boundary conditions specified along each side of square:

第一类 *Dirichlet*: u specified

第二类 *Neumann*: u_x or u_y specified

Mixed: combinations of these specified 混和边界条件

Finite Difference Methods

Finite difference methods for such problems proceed as before:

- Define discrete mesh of points within domain of equation
- Replace derivatives in PDE by finite differences
- Seek numerical solution at mesh points

Unlike time-dependent problems, solution not produced by marching forward step by step in time

并非按时间一步一步求

Approximate solution determined at all mesh points simultaneously by solving single system of algebraic equations

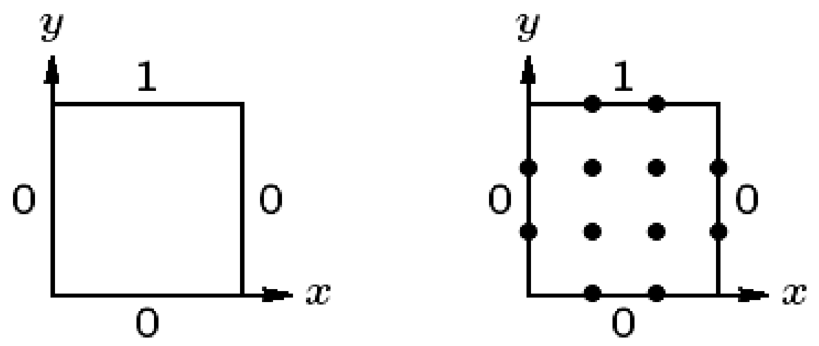
通过解代数方程直接得到所有点的近似解，“纯边值”问题

Example: Laplace Equation

Consider Laplace equation

$$u_{xx} + u_{yy} = 0$$

on unit square with boundary conditions shown on left below



Define discrete mesh in domain, including boundaries, as shown on right above

Interior grid points where we will compute approximate solution are given by

$$(x_i, y_j) = (ih, jh), \qquad i, j = 1, \ldots, n,$$

where in example $n = 2$ and $h = 1/(n + 1) = 1/3$

为了准确性，实际步长可能很小

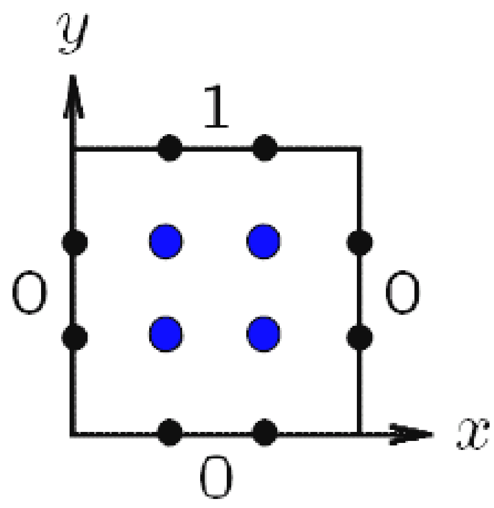
Finite Difference Methods

- Define discrete mesh of points within domain of equation
- Replace derivatives in PDE by finite differences
- Seek numerical solution at mesh points

并非按时间一步一步求

通过解代数方程直接得到所有点的近似解; “纯边值” 问题

Example: Laplace equation on unit square



二阶中心差分

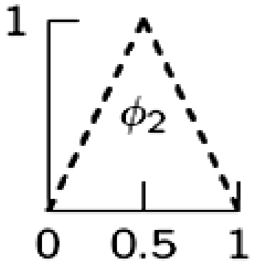
$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{h^2} = 0$$

$$Ax = \begin{bmatrix} 4 & -1 & -1 & \\ -1 & 4 & & -1 \\ -1 & & 4 & -1 \\ & -1 & -1 & 4 \end{bmatrix} \begin{bmatrix} u_{1,1} \\ u_{2,1} \\ u_{1,2} \\ u_{2,2} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

$\longrightarrow$ 稀疏矩阵解法

$$\begin{bmatrix} u_{1,1} \\ u_{2,1} \\ u_{1,2} \\ u_{2,2} \end{bmatrix} = \begin{bmatrix} 0.125 \\ 0.125 \\ 0.375 \\ 0.375 \end{bmatrix}$$

基函数是B样条



Finite element methods are also applicable to boundary value problems for PDEs as well as for ODEs

Conceptually, there is no change in going from one dimension to two or three dimensions:

- Solution is represented as linear combination of basis functions
- Some criterion (e.g., Galerkin) is applied to derive system of equations that determines coefficients of linear combination

高维定义域空间的离散

四面体
六面体

Main practical difference is that instead of subintervals in one dimension, elements usually become triangles or rectangles in two dimensions, or tetrahedra or hexahedra in three dimensions

Finite Element Methods, continued

Basis functions typically used are bilinear or bicubic functions in two dimensions or trilinear or tricubic functions in three dimensions, analogous to "hat" functions or piecewise cubics in one dimension

二维：
双线性、双三次插值

三维：
三线性、三三次

Increase in dimensionality means that linear system to be solved is much larger, but it is still sparse due to local support of basis functions

含未知量增加

矩阵仍然稀疏

Finite element methods for PDEs are extremely flexible and powerful, but detailed treatment of them is beyond scope of this course

总结——时不变边值问题的求解

- 椭圆型方程的种类——亥姆荷兹方程、泊松方程、拉普拉斯方程
- 三种边界条件——Dirichlet、Neumann、mixed
- 有限差分解法
  - 基本步骤
  - 一个Laplace方程的例子
- 有限元解法
  - 基本步骤
  - 高维情况带来的一些复杂情况

Matlab topics

- Matlab commands for PDE

一维空间的抛物型, 椭圆型方程

  - *pdepe* (Solve initial-boundary value problems for parabolic-elliptic PDEs in 1-D)
  - Syntax: *sol =pdepe(m,pdefun,icfun,bcfun,xmesh,tspan)*
  - m表示定义域类型: 长条形(0), 圆柱对称(1), 球对称(2)
  - Pde方程: $c\left(x,t,u,\frac{\partial u}{\partial x}\right)\frac{\partial u}{\partial t} = x^{-m}\frac{\partial}{\partial x}\left(x^m f\left(x,t,u,\frac{\partial u}{\partial x}\right)\right) + s\left(x,t,u,\frac{\partial u}{\partial x}\right)$
  - pdefun: *[c,f,s] = pdefun(x, t, u, dudx)*
  - bcfun: $p(x,t,u) + q(x,t)f\left(x,t,u,\frac{\partial u}{\partial x}\right) = 0$
  - Demo $u_t = u_{xx}, \quad t \geq 0, \quad u(0,x) = \sin x$
  - PDE Toolbox: 2-D FEM方法

c为对角阵, 其余向量

热传导问题: **pde_1**