PDE and Boundary-Value Problems
Winter Term 2016/2017

Lecture 20

Saarland University

15 February 2017

Purpose of Lesson

- To show how a PDE can be changed to a system of algebraic equations by replacing the partial derivatives in the differential equation with their finite-difference approximations. The system of algebraic equations can then be solved numerically by an iterative process in order to obtain an approximate solution to the PDE.
- To introduce the idea of explicit finite-difference methods and show how they can be used to solve hyperbolic and parabolic problems.

- We can solve elliptic BVPs (steady-state problems) where the PDE was satisfied in a given region of space, and the solution (or its derivative) was specified on the boundary.
- In those types of problems, we find the approximate solution at the interior grid points by solving a system of algebraic equations. In other words, the solution at all the interior grid points was found simultaneously.

Chapter 5. Numerical and Approximate Methods

- So far, we have studied several techniques for solving linear PDEs. However, most of the equations we've attackes were reasonably simple, had reasonably simple BCs, and had reasonably shaped domains.
- But many problems cannot be simplified to fit this general mold and must be solved by numerical approximations.
- To begin, we introduce the idea of finite differences. We then show how to use these finite differences to solve a Dirichlet problem inside a square.

Finite-Difference Approximations

- First, we recall the Taylor series expansion of a function $f(x)$

$$f(x+h) = f(x) + f'(x)h + \frac{f''(x)}{2!}h^2 + \dots$$

- If we truncate this series after two terms, we have the approximation

$$f(x+h) \cong f(x) + f'(x)h$$

  Hence, we can solve for $f'(x)$

$$f'(x) \cong \frac{f(x+h) - f(x)}{h}$$

  which is called the forward-difference approximation to the first derivative $f'(x)$.

Finite-Difference Approximations (cont.)

- We could also replace h by $-h$ in the Taylor series and arrive at the backward-difference approximation

$$f'(x) \cong \frac{f(x) - f(x-h)}{h}$$

or by subtracting

$$f(x-h) \cong f(x) - f'(x)h$$

from

$$f(x+h) \cong f(x) + f'(x)h$$

we can obtain the central-difference approximation

$$f'(x) \cong \frac{1}{2h}\left[f(x+h) - f(x-h)\right].$$

Finite-Difference Approximations (cont.)

- By retaining another term in the Taylor series, this type of analysis can be extended to arrive at the central-difference approximation of the second derivative $f''(x)$

$$f''(x) \cong \frac{1}{h^2}\left[f(x+h) - 2f(x) + f(x-h)\right].$$

- We now extend the finite-difference approximations to partial derivatives. If we begin with the Taylor series expansion in two variables

$$u(x+h,y) = u(x,y) + u_x(x,y)h + u_{xx}(x,y)\frac{h^2}{2!} + \dots$$
$$u(x-h,y) = u(x,y) - u_x(x,y)h + u_{xx}(x,y)\frac{h^2}{2!} - \dots$$

we can deduce the following:

$$u_x(x,y) \cong \frac{u(x+h,y) - u(x,y)}{h} \quad \text{(Forward difference)}$$
$$u_{xx}(x,y) \cong \frac{1}{h^2}\left[u(x+h,y) - 2u(x,y) + u(x-h,y)\right]$$
$$u_y(x,y) \cong \frac{1}{k}\left[u(x,y+k) - u(x,y)\right]$$
$$u_{yy}(x,y) \cong \frac{1}{k^2}\left[u(x,y+k) - 2u(x,y) + u(x,y-k)\right].$$

Remarks

- Which approximation of partial derivatives is used (forward, central, or backward) depends on the problem.
- We will consider the central-difference approximation.

Dirichlet Problem Solved by the Finite-Difference Method

To illustrate how to use these finite-difference approximations, we consider the simple Dirichlet problem.

Problem 20-1

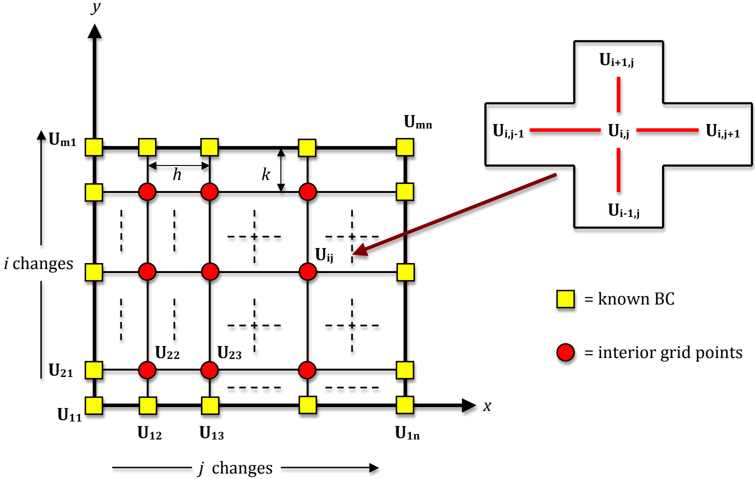
To find a function $u(x,y)$ that satisfies

$$\text{PDE:} \qquad u_{xx} + u_{yy} = 0, \qquad 0 < x < 1, \quad 0 < y < 1$$

$$\text{BCs:} \quad \begin{cases} u(x,y) = 0 & \text{On the top and sides of the square} \\ u(x,0) = \sin(\pi x) & 0 \leqslant x \leqslant 1 \end{cases}$$

We begin with the drawing the grid system on the *xy*-plane.

Grid lines for the Dirichlet problem inside a square



It is convenient to use the following notation:

$$
\begin{aligned}
u(x,y) &= u_{i,j}\\
u(x,y+k) &= u_{i+1,j}\\
u(x,y-k) &= u_{i-1,j}\\
u(x+h,y) &= u_{i,j+1}\\
u(x-h,y) &= u_{i,j-1}\\
u_x(x,y) &= \frac{1}{2h}(u_{i,j+1} - u_{i,j-1})\\
u_y(x,y) &= \frac{1}{2k}(u_{i+1,j} - u_{i-1,j})\\
u_{xx}(x,y) &= \frac{1}{h^2}(u_{i,j+1} - 2u_{i,j} + u_{i,j-1})\\
u_{yy}(x,y) &= \frac{1}{k^2}(u_{i+1,j} - 2u_{i,j} + u_{i-1,j})
\end{aligned}
$$

- Our strategy for solving the Dirichlet problem 20-1 is to replace the partial derivatives in Laplace's equation

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

by their finite-difference approximations.

- Using the compact notation $u_{i,j}$, we have the following difference equation:

$$\Delta u = \frac{1}{h^2}(u_{i,j+1} - 2u_{i,j} + u_{i,j-1}) + \frac{1}{k^2}(u_{i+1,j} - 2u_{i,j} + u_{i-1,j}) = 0.$$

- By letting the two discretization sizes h and k be the same, Laplace's equation is replaced by

$$(u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j}) = 0$$

or solving for $u_{i,j}$

$$\boxed{u_{i,j} = \frac{1}{4}\left(u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1}\right)} \qquad (20.1)$$

Remarks

1. $u_{i,j}$ stands for the solution at the interior grid points.
2. Equation (20.1) says that we can approximate the solution $u_{i,j}$ by averaging the solution at four neighboring grid points.

Numerical Algorithm for Solving the Dirichlet Problem (Liebmann's Method)

1. Seek the solution $u_{i,j}$ at the interior grid points by setting them equal to the average of all the BCs (reasonable start).
2. Systematically run over al the interior grid points, replacing the old estimates by the average of its four neighbors.

Remarks

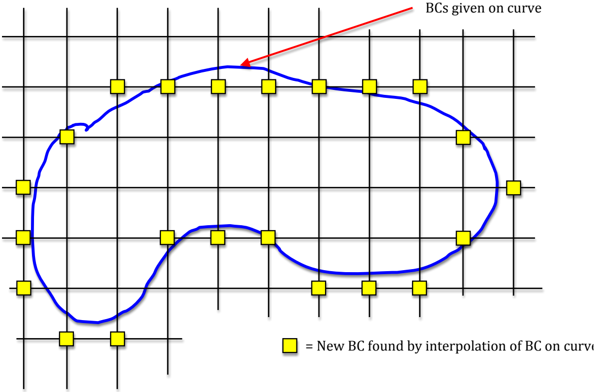
1. It doesn't make much difference in what order this process is carried out, but, generally, it si done in a row by row (or colunm by colunm) manner.
2. After a few iterations, this process will converge to an approximate solution of the problem.
3. The rate of convergence is generally slow but can be speeded up in a number of ways.

Remarks

- If we made our discretization sizes h and k smaller (so that we had more grid points), the analysis would be similar except that the system of obtained algebraic equations would be larger.
- In general, the number of equations will be equal to the number of interior grid points.
- To solve the Neumann problem where there are derivatives on the boundary we must also replace these derivatives by some finite difference approximation.
- We can also solve equations with variable coefiicients and nonhomogeneous equations by the finite-difference method.

Remarks (cont.)

- If the domain of the problem is an irregularly shaped region, we can overlay the region with grid lines and then approximate the solution at nearby grid points by interpolation the BCs.



An Explicit Finite-Difference Method

- Now we will show how time-dependent problems can be solved by finite-difference approximations.
- The idea is that if we are given the solution when time is zero, we can then find the solution for $t = \Delta t, 2\Delta t, 3\Delta t, \dots$ by means of a marching process.
- Replacing both the space and time derivatives by their finite-difference approximations, we can then solve for the solution $u_{i,j}$ in the difference equation explicitly in terms of the solution at earlier values of time.
- This process is called an explicit-type marching process, since we find the solution at a single value of time in terms of the solution at earlier values of time.

The Explicit Method for Parabolic Equations

- To show how the explicit finite-difference method works, we consider a representative problem from heat flow.
- Heat flows along a rod initially at temperature zero, where the left end of the rod is fixed at temperature one, and the right-hand side experiences a heat loss (or gain) proportional to the difference between the temperature at that end and an outside temperature that is given by $g(t)$.

The Explicit Method for Parabolic Equations (cont.)

Problem 20-2

To find a function $u(x,t)$ that satisfies

$$\text{PDE:} \quad u_t = u_{xx}, \qquad 0 < x < 1, \quad 0 < t < \infty$$

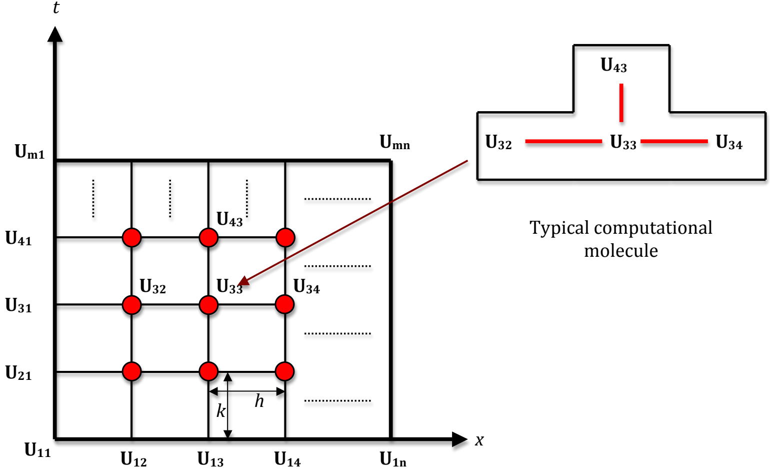
$$\text{BCs:} \quad \begin{cases} u(0,t) = 1 \\ u_x(1,t) = -\left[u(1,t) - g(t)\right] \end{cases} \qquad 0 < t < \infty$$

$$\text{IC:} \quad u(x,0) = 0 \qquad 0 \leqslant x \leqslant 1$$

To solve problem 20-2 by finite differnces, we start by drawing the usual rectangular grid system with grid points

$$x_j = jh \quad j = 0, 1, 2, \ldots, n$$
$$t_i = ik \quad i = 0, 1, 2, \ldots, m$$

Grid system for a heat-flow problem



- Note that on the figure of the grid system, the $u_{i,j}$ on the left and bottom are given BCs and ICs, and our job is to find the other $u_{i,j}$'s.
- To do this, we begin by replacing the partial derivatives u_t and u_{xx} in the heat equation with their approximations

$$\begin{aligned} u_t &= \frac{1}{k}\left[u(x, t+k) - u(x,t)\right] = \frac{1}{k}\left(u_{i+1,j} - u_{i,j}\right) \\ u_{xx} &= \frac{1}{h^2}\left[u(x+h,t) - 2u(x,t) + u(x-h,t)\right] \\ &= \frac{1}{h^2}\left(u_{i,j+1} - 2u_{i,j} + u_{i,j-1}\right) \end{aligned}$$

- By substituting these approximations into $u_t = u_{xx}$ and solving for the solution at the largest value of time, we have

$$\boxed{u_{i+1,j} = u_{i,j} + \frac{k}{h^2}\left[u_{i,j+1} - 2u_{i,j} + u_{i,j-1}\right]} \qquad (20.2)$$

Remark

(20.2) is the formula we are looking for, since it gives us the solution at one value of time in terms of the solution at earlier values of time.

- We are almost ready to begin the computations for problem 20-2. First, we must approximate the derivatives in the right-hand BC

$$u_x(1,t) = -\left[u(1,t) - g(t)\right]$$

by

$$\frac{1}{h}\left[u_{i,n} - u_{i,n-1}\right] = -\left[u_{i,n} - g_i\right], \tag{20.3}$$

where $g_i = g(ik)$ is given.

- Note that in (20.3) we have replaced $u_x(1,t)$ by the backward-difference approximation, since the forward-difference approximation would require knowing values of $u_{i,j}$ outside the domain.

- Solving (20.3) for $u_{i,n}$ gives us

$$u_{i,n} = \frac{u_{i,n-1} + hg_i}{1+h}. \tag{20.4}$$

Algorithm for the Explicit Method

1. Find the solution at the grid points for $t = \Delta t$ by using the explicit formula

$$u_{2,j} = u_{1,j} + \frac{k}{h^2}\left[u_{1,j+1} - 2u_{1,j} + u_{1,j-1}\right] \qquad j = 2, 3, \ldots, n-1$$

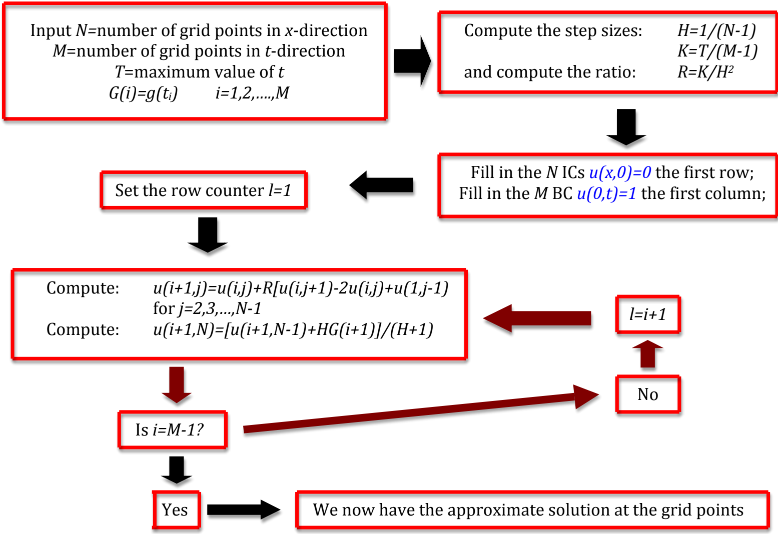
2. Find $u_{2,n}$ from formula (20.4)

$$u_{2,n} = \frac{u_{2,n-1} + hg_2}{1+h}.$$

Remark

- Steps 1 and 2 find the solution for $t = \Delta t$.
- To find the solution for $t = 2\Delta t$ repeat steps 1 and 2, moving up one more row (increase i by 1) and using the values of $u_{i,j}$ just computed.
- For $t = 3\Delta t, 4\Delta t, \dots$ keep repeating the same process.

On the flow diagram on the next page we explain in a precise manner how the computations should be carried out.


Input N=number of grid points in x-direction
M=number of grid points in t-direction
T=maximum value of t
G(i)=g(tᵢ) i=1,2,....,M
Compute the step sizes: H=1/(N-1)
K=T/(M-1)
and compute the ratio: R=K/H²
Fill in the N ICs u(x,0)=0 the first row;
Fill in the M BC u(0,t)=1 the first column;
Set the row counter l=1
Compute: u(i+1,j)=u(i,j)+R[u(i,j+1)-2u(i,j)+u(1,j-1)
for j=2,3,...,N-1
Compute: u(i+1,N)=[u(i+1,N-1)+HG(i+1)]/(H+1)
l=i+1
No
Is i=M-1?
Yes
We now have the approximate solution at the grid points

Remarks

- There is a serious dificiency in the explicit method, for if the step size in t is large compared to the step size in x, then machine roundoff error can grow until it ruins the accuracy of the solution.
- The relative size of these steps depends on the particular equation and the BCs, but, generally, the step size in t should be much smaller than the step size in x. We must have $k/h^2 \leqslant 0.5$ in order this method to work.
- A general rule of thumb is that as the step sizes Δt and Δx are made smaller, the truncation error of approximating partial derivatives by finite differences decreases. However, the smaller these grid sizes, the more computations necessary, and, hence, the roundoff error, as a result of rounding off our computations, will increase.

The Explicit Method for Hyperbolic Equation

Problem 20-3

To find a function $u(x,t)$ that satisfies

$$\text{PDE:} \quad u_{tt} = u_{xx}, \qquad 0 < x < 1, \quad 0 < t < \infty$$

$$\text{BCs:} \quad \begin{cases} u(0,t) = g_1(t) \\ u(1,t) = g_2(t) \end{cases} \quad 0 < t < \infty$$

$$\text{ICs:} \quad \begin{cases} u(x,0) = \phi(x) \\ u_t(x,0) = \psi(x) \end{cases} \quad 0 \leqslant x \leqslant 1$$

- Problem 20-3 can also be solved by the explicit finite-difference method. Here, we can approximate the derivatives u_{tt} and u_{xx} by

$$u_{tt} \cong \frac{1}{k^2}\left[u(x,t+k) - 2u(x,t) + u(x,t-k)\right]$$
$$u_{xx} \cong \frac{1}{h^2}\left[u(x+h,t) - 2u(x,t) + u(x-h,t)\right]$$

and the derivative $u_t(x,0)$ in the IC by

$$u_t(x,0) \cong \frac{1}{k}\left[u(x,k) - u(x,0)\right] = \frac{1}{k}\left[u(x,k) - \phi(x)\right].$$

- Solving for $u(x, t+k)$ explicitly in terms of the solution at earlier values of time gives

$$\boxed{\begin{aligned} u(x,t+k) &= 2u(x,t) - u(x,t-k) \\ &\quad + \left(\frac{k}{h}\right)^2 [u(x+h,t) - 2u(x,t) + u(x-h,t)] \end{aligned}} \tag{20.5}$$

- From (20.5) it is clear that we must already know the solution at two previous time steps, and, hence, we must use the initial-velocity condition

$$\frac{1}{k}\left[u(x,k) - \phi(x)\right] = \psi(x)$$

to get us started. Solving for $u(x,k)$ gives $u(x,k) = \phi(x) + k\psi(x)$, and, thus, we can find the solution for $t = \Delta t$.

An Implicit Finite-Difference Method (Crank-Nicolson Method)

- In implicit method, we again replace the partial derivatives in the problem by their finite-difference approximations, but unlike explicit methods (where we solved for $u_{i+1,j}$ explicitly in terms of earlier values), in implicit methods, we solve a system of equations in order to find the solution at the largest value of time.
- In other words, for each new value of time we solve a system of algebraic equations to find all the values.
- It should be mentioned that implicit methods allow us to take larger steps by doing more work per step.

The Heat-Flow Problem Solved by an Implicit Method

Problem 20-4

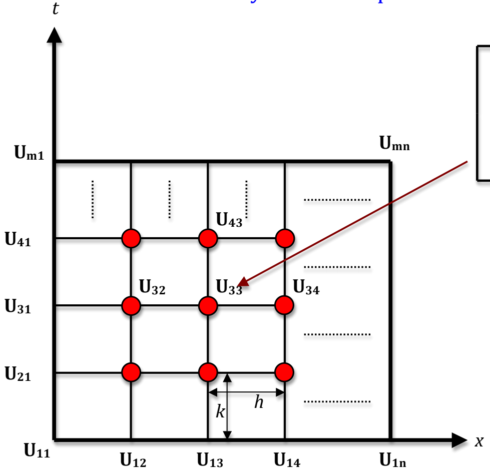
To find a function $u(x,t)$ that satisfies

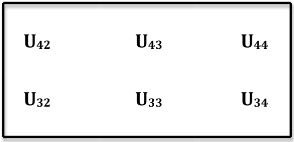
$$\text{PDE:} \quad u_t = u_{xx}, \qquad 0 < x < 1, \quad 0 < t < \infty$$

$$\text{BCs:} \quad \begin{cases} u(0,t) = 0 \\ u(1,t) = 0 \end{cases} \quad 0 < t < \infty$$

$$\text{IC:} \quad u(x,0) = 1 \qquad 0 \leqslant x \leqslant 1$$

Grid system for implicit scheme *(Δx=0.2)*





Typical computational molecule

- We replace the partial derivatives u_t and u_{xx} by the following approximations:

$$u_t(x,t) = \frac{1}{k}\left[u(x,t+k) - u(x,t)\right]$$

$$u_{xx}(x,t) = \frac{\lambda}{h^2}\left[u(x+h,t+k) - 2u(x,t+k) + u(x-h,t+k)\right] + \frac{(1-\lambda)}{h^2}\left[u(x+h,t) - 2u(x,t) + u(x-h,t)\right],$$

  where λ is a chosen number in the interval $[0,1]$.

- Note that our approximation for u_{xx} is a weighted average of the central-difference approximation to the derivative u_{xx} at time values t and $t+k$.

Remarks

- In the special case when $\lambda = 0.5$, it is just the ordinary average of these two central differences.
- If $\lambda = 0.75$, our approximation puts weights of 0.75 and 0.25 on each of the two terms.
- If $\lambda = 0$, it is usual explicit finite-difference method.

If we now substitute the approximations for u_t and u_{xx} into our problem, we get the new finite-difference problem

Problem 20-4a

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

$$\text{BCs:} \quad \begin{cases} u_{i,1} = 0 \\ u_{i,n} = 0 \end{cases}, \qquad i = 1, 2, \ldots, m$$

$$\text{IC:} \quad u_{1,j} = 1, \qquad j = 2, \ldots, n-1$$

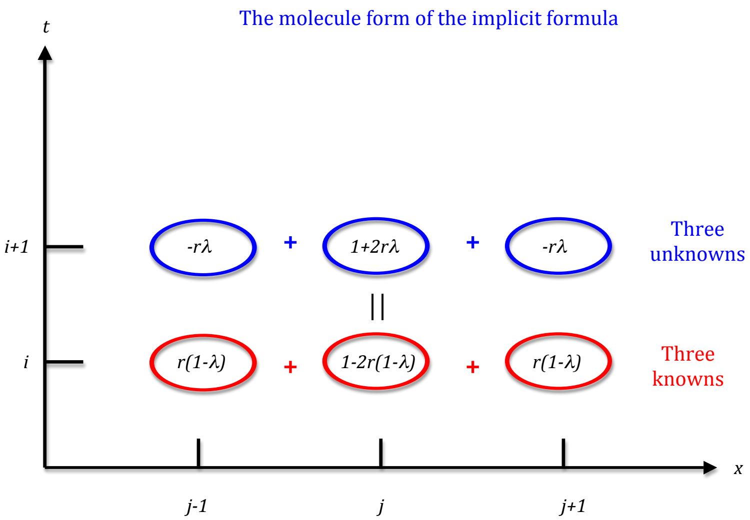
- If we rewrite the difference equation in problem 20-4a, putting the $u_{i,j}$'s with the largest time subscript (i-subscript) on the left-hand side of the equation, we arrive at

$$\boxed{\begin{aligned} -\lambda r u_{i+1,j+1} + (1+2r\lambda)u_{i+1,j} - \lambda r u_{i+1,j-1} \\ = r(1-\lambda)u_{i,j+1} + \left[1-2r(1-\lambda)\right]u_{i,j} \\ + r(1-\lambda)u_{i,j-1}, \end{aligned}} \qquad (20.6)$$

  where we have set $r = k/h^2$ for convenience.

- Note that for a fixed subscript i and for j going from 2 to $n-1$, this is a system of $n-2$ equations in the $n-2$ unknowns $u_{i+1,2}$, $u_{i+1,3}$, $u_{i+1,4}, \ldots, u_{i+1,n-1}$ [which are the interior grid points at $t = (i+1)\Delta t$.

- To help show exactly how $u_{i,j}$'s are involved into (20.6), we write it in the symbolic or molecular form (see next page)


The molecule form of the implicit formula
t
i+1
-rλ
+
1+2rλ
+
-rλ
Three unknowns
i
r(1-λ)
+
1-2r(1-λ)
+
r(1-λ)
Three knowns
x
j-1
j
j+1