Elliptic Partial Differential Equations

1. Partial Differential Equations
   - Poisson and Laplace
   - A first example of the Laplace equation

2. The Finite Difference Method
   - A rectangular mesh of points
   - Apply the lexicographical order
   - Computational experiments

3. Conclusions
   - About numerical analysis

MCS 471 Lecture 40
Numerical Analysis
Jan Verschelde, 24 November 2021
Partial Differential Equations

1. Poisson and Laplace
   - a first example of the Laplace equation

2. the Finite Difference Method
   - a rectangular mesh of points
   - apply the lexicographical order
   - computational experiments

3. Conclusions
   - about numerical analysis
Partial Differential Equations

We look for functions $u(x, y)$ as the solution of the Partial Differential Equation (or PDE):

$$Au_{xx} + Bu_{xy} + Cu_{yy} + F(x, y, u, u_x, u_y) = 0,$$

where $x$ and $y$ are the independent variables, and

$$u_x = \frac{du}{dx}, \quad u_y = \frac{du}{dy}, \quad u_{xx} = \frac{d^2u}{dx^2}, \quad u_{yy} = \frac{d^2u}{dy^2}, \quad u_{xy} = \frac{d^2u}{dxdy}.$$

Instead of $y$, time $t$ may be the second independent variable.

Second order PDEs are classified by $B^2 - 4AC$:

- If $B^2 - 4AC = 0$, then the PDE is *parabolic* (heat).
- If $B^2 - 4AC > 0$, then the PDE is *hyperbolic* (wave).
- If $B^2 - 4AC < 0$, then the PDE is *elliptic* (steady state).
the Poisson and Laplace equations

Let \( u(x, y) \) be a twice differentiable function, the \textit{Laplacian} of \( u \) is

\[
\Delta u = u_{xx} + u_{yy}.
\]

The \textit{Poisson equation} has the format

\[
\begin{cases}
    \Delta u = f(x, y) & \text{for all } x \in [a, b], y \in [c, d], \\
    u(x, c) = g_1(x) & \text{for all } x \in [a, b], \\
    u(x, d) = g_2(x) & \text{for all } x \in [a, b], \\
    u(a, y) = g_3(y) & \text{for all } y \in [c, d], \\
    u(b, y) = g_4(y) & \text{for all } y \in [c, d].
\end{cases}
\]

If \( f = 0 \), then the equation is the \textit{Laplace equation}.

We distinguish between two “pure” types of boundary conditions:

1. \textbf{Dirichlet}: values on the boundary of \([a, b] \times [c, d] \) (as above),
2. \textbf{Neumann}: derivatives on the boundary of \([a, b] \times [c, d] \).
the Laplace equation is elliptic

Our general format of a PDE is:

\[ Au_{xx} + Bu_{xy} + Cu_{yy} + F(x, y, u, u_x, u_y) = 0. \]

The Laplace equation: \( u_{xx} + u_{yy} = 0. \)

In the format above, \( A = 1, B = 0, \) and \( C = 1. \)

We then compute:

\[ B^2 - 4AC = 0 - 4 \cdot 1 \cdot 1 = -4 < 0. \]

Thus, the Laplace equation is elliptic.
Elliptic Partial Differential Equations

1. Partial Differential Equations
   - Poisson and Laplace
   - a first example of the Laplace equation

2. the Finite Difference Method
   - a rectangular mesh of points
   - apply the lexicographical order
   - computational experiments

3. Conclusions
   - about numerical analysis
a first example of the Laplace equation

Consider the Laplace equation over $[0, 1] \times [0, 1]$:

$$
\begin{align*}
\Delta u &= 0 \quad \text{for all } x \in [0, 1], y \in [0, 1], \\
u(x, 0) &= x^2 \quad \text{for all } x \in [0, 1], \\
u(x, 1) &= x^2 - 1 \quad \text{for all } x \in [0, 1], \\
u(0, y) &= -y^2 \quad \text{for all } y \in [0, 1], \\
u(1, y) &= 1 - y^2 \quad \text{for all } y \in [0, 1].
\end{align*}
$$

A solution to this equation is $u(x, y) = x^2 - y^2$.

This could model the temperature distribution on a square floor.
the solution $u(x, y) = x^2 - y^2$
Elliptic Partial Differential Equations

1. Partial Differential Equations
   - Poisson and Laplace
   - a first example of the Laplace equation

2. the Finite Difference Method
   - a rectangular mesh of points
   - apply the lexicographical order
   - computational experiments

3. Conclusions
   - about numerical analysis
a rectangular mesh of points

The rectangle \([a, b] \times [c, d]\) is divided

1. using \(M\) equidistant interior points in \([a, b]\),
2. using \(N\) equidistant interior points in \([c, d]\).

\[
\begin{align*}
&y_1 = y_0 \\
&y_2 \\
&y_3 \\
&y_4 = y_4 \\
&d = y_4
\end{align*}
\]

\[
\begin{align*}
a = x_0 & \quad x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 = b
\end{align*}
\]

Black dots are given boundary values. Red dots are unknown.
applying central differences

We apply central differences to the Poisson equation:

\[ u_{xx} + u_{yy} = f(x, y), \]

using step \( h \) for \( x \) and step \( k \) for \( y \):

\[
\frac{u(x - h, y) - 2u(x, y) + u(x + h, y)}{h^2} + \frac{u(x, y - k) - 2u(x, y) + u(x, y + k)}{k^2} + O(h^2) + O(k^2) = f(x, y),
\]

where \( h \) and \( k \) are as follows:
- for \( M \) interior points in \([a, b]\): \( h = (b - a)/(M + 1) \),
- for \( N \) interior points in \([c, d]\): \( k = (d - c)/(N + 1) \).
the discrete Poisson equation

With step $h$ for $x$ and step $k$ for $y$:

$$\frac{u(x - h, y) - 2u(x, y) + u(x + h, y)}{h^2} + \frac{u(x, y - k) - 2u(x, y) + u(x, y + k)}{k^2} = f(x, y).$$

Let $z_{i,j}$ be the approximation for $u(a + i h, c + j k)$:

$$\frac{z_{i-1,j} - 2z_{i,j} + z_{i+1,j}}{h^2} + \frac{z_{i,j-1} - 2z_{i,j} + z_{i,j+1}}{k^2} = f(x_i, y_j),$$

with $(x_i, y_j)$ defined as

- $x_i = a + i h$, for $i = 0, 1, \ldots, M + 1$, and
- $y_j = c + j k$, for $j = 0, 1, \ldots, N + 1$. 
solving for $M \times N$ unknowns

For $h = (b - a)/(M + 1)$, $k = (d - c)/(N + 1)$, the $z_{i,j}$ are solutions of

$$\frac{z_{i-1,j} - 2z_{i,j} + z_{i+1,j}}{h^2} + \frac{z_{i,j-1} - 2z_{i,j} + z_{i,j+1}}{k^2} = f(x_i, y_j),$$

$i = 1, 2, \ldots, M$, $j = 1, 2, \ldots, N$, over a grid of $M \times N$ points.
1. Partial Differential Equations
   - Poisson and Laplace
   - A first example of the Laplace equation

2. The Finite Difference Method
   - A rectangular mesh of points
   - Apply the lexicographical order
   - Computational experiments

3. Conclusions
   - About numerical analysis
ordering the $M \times N$ variables

We have a linear system of $M \times N$ equations in $M \times N$ variables.

To order the interior points, use the lexicographic order:

$(1, 1), (1, 2), (1, 3), (2, 1), (2, 2), (2, 3), (3, 1), (3, 2), (3, 3), (4, 1), (4, 2), (4, 3)$.

The point $(x_i, y_j)$, indexed by $(i, j)$, is mapped to $j + (i - 1)N$. 
ordering the $M \times N$ equations

The variable $z_{i,j}$ appears central in the equation $(i, j)$:

$$\frac{z_{i-1,j} - 2z_{i,j} + z_{i+1,j}}{h^2} + \frac{z_{i,j-1} - 2z_{i,j} + z_{i,j+1}}{k^2} = f(x_i, y_j),$$

for $i = 1, 2, \ldots, M$ and $j = 1, 2, \ldots, N$.

Apply the map that sends $(i, j)$ to $j + (i - 1)N$: $b_{j+(i-1)N} = f(x_i, y_j)$.

Equation $(i, j)$ is mapped to equation $j + (i - 1)N$, $Z_{j+(i-1)N} = z_{i,j}$:

$$\frac{1}{h^2}Z_{j+(i-2)N} + \frac{1}{h^2}Z_{j+iN} - \left(\frac{2}{h^2} + \frac{2}{k^2}\right)Z_{j+(i-1)N}$$

$$+ \frac{1}{k^2}Z_{j-1+(i-1)N} + \frac{1}{k^2}Z_{j+1+(i-1)N} = b_{j+(i-1)N}.$$

The boundary values are needed for $i = 1, i = M, j = 1, j = N$, 

---

**Numerical Analysis (MCS 471)**

**Elliptic PDEs**

L-40 24 November 2021 16 / 33
the matrix of the linear system

Equation $j + (i - 1)N$

\[
\frac{1}{h^2} Z_{j+(i-2)N} + \frac{1}{h^2} Z_{j+iN} - \left( \frac{2}{h^2} + \frac{2}{k^2} \right) Z_{j+(i-1)N} \\
+ \frac{1}{k^2} Z_{j-1+(i-1)N} + \frac{1}{k^2} Z_{j+1+(i-1)N} = b_{j+(i-1)N}
\]

defines the coefficients on row $r = j + (i - 1)N$ of the matrix $A$:

\[
A_{r,j+(i-2)N} = \frac{1}{h^2}, \quad A_{r,j+(i-1)N} = -\left( \frac{2}{h^2} + \frac{2}{k^2} \right), \quad A_{r,j+iN} = \frac{1}{h^2},
\]

\[
A_{r,j-1+(i-1)N} = \frac{1}{k^2}, \quad \text{and} \quad A_{r,j+1+(i-1)N} = \frac{1}{k^2},
\]

where $i = 1, i = M, j = 1, j = N$ require special attention.
using the boundary values

We label the functions on the boundary of \([a, b] \times [c, d]\) as

\[
\begin{align*}
\text{bottom: } & \quad u(x, c) = g_1(x) \quad \text{for all } x \in [a, b], \\
\text{top: } & \quad u(x, d) = g_2(x) \quad \text{for all } x \in [a, b], \\
\text{left: } & \quad u(a, y) = g_3(y) \quad \text{for all } y \in [c, d], \\
\text{right: } & \quad u(b, y) = g_4(y) \quad \text{for all } y \in [c, d].
\end{align*}
\]

The bottom and top respectively correspond to \(i = 0\) and \(i = M + 1\); the left and right respectively correspond to \(j = 0\) and \(j = N + 1\).

Assign to \(b_{j+(i-1)N}\) the following

1. \(b_{j+(i-1)N} := b_{j+(i-1)N} - g_1(x_i)/k^2\), for \(j = 1\) and for \(i = 1, 2, \ldots, M\),
2. \(b_{j+(i-1)N} := b_{j+(i-1)N} - g_2(x_i)/k^2\), for \(j = N\) and for \(i = 1, 2, \ldots, M\),
3. \(b_{j+(i-1)N} := b_{j+(i-1)N} - g_3(y_j)/h^2\), for \(i = 1\) and for \(j = 1, 2, \ldots, N\),
4. \(b_{j+(i-1)N} := b_{j+(i-1)N} - g_4(y_j)/h^2\), for \(i = M\) and for \(j = 1, 2, \ldots, N\).
on the structure of the matrix $A$

Exercise 1:

1. Draw the stencil for the general equation.

2. Is the matrix $A$ tridiagonal?
   Do experiments with the posted Julia code.
   Describe the structure of the matrix $A$.

3. We used the lexicographical order to define the linear system.
   Could other orders be more beneficial?
Elliptic Partial Differential Equations

1. Partial Differential Equations
   - Poisson and Laplace
   - A first example of the Laplace equation

2. The Finite Difference Method
   - A rectangular mesh of points
   - Apply the lexicographical order
   - Computational experiments

3. Conclusions
   - About numerical analysis
specification of a Julia function

```
""

    laplace(bottomf::Function,topf::Function,
            leftf::Function,rightf::Function,
            a::Float64,b::Float64,c::Float64,d::Float64,
            M::Int,N::Int)

applies the finite difference method with a lexicographic order

to the Laplace equation over the rectangle \([a,b]\times[c,d]\),

using M interior points in \([a,b]\) and N interior points in \([c,d]\),

for a total of M*N interior points.

```

bottomf and topf define the bottom and top boundary values,
leftf and rightf define the left and right boundary values.

Example:

```
    bottom(x) = x^2
    top(x) = x^2 - 1.0
    left(y) = -y^2
    right(y) = 1.0 - y^2

approx = laplace(bottom,top,left,right,0.0,1.0,0.0,1.0,4,3)
```

"""
function laplace(bottomf::Function, topf::Function, 
    leftf::Function, rightf::Function, 
    a::Float64, b::Float64, c::Float64, d::Float64, 
    M::Int, N::Int)

    h = (b-a)/(M+1)
    k = (d-c)/(N+1)
    mat = zeros(M*N,M*N)
    rhs = zeros(M*N)
    for i=1:M
        # j = 1
        rhs[1+(i-1)*N] = rhs[1+(i-1)*N] - bottomf(a+i*h)/k^2
        # j = N
        rhs[N+(i-1)*N] = rhs[N+(i-1)*N] - topf(a+i*h)/k^2
    end
    for j=1:N
        # i = 1
        rhs[j] = rhs[j] - leftf(c+j*k)/h^2
        # i = M
        rhs[j+(M-1)*N] = rhs[j+(M-1)*N] - rightf(c+j*k)/h^2
    end
the definition of the matrix

for i=1:M
    for j=1:N
        row = j + (i-1)*N
        mat[row,row] = -(2/h^2 + 2/k^2)
        if i > 1
            mat[row,j+(i-2)*N] = 1/h^2
        end
        if i < M
            mat[row,j+i*N] = 1/h^2
        end
        if j > 1
            mat[row,j-1+(i-1)*N] = 1/k^2
        end
        if j < N
            mat[row,j+1+(i-1)*N] = 1/k^2
        end
    end
end
the solution of the linear system

```matlab
x = mat\rhs
sol = zeros(M,N)
for i=1:M
    for j=1:N
        sol[i,j] = x[j+(i-1)*N]
    end
end

return sol
end
```
running on the first example

```
$ julia example1Laplace.jl

The exact solution:
4×3 Array{Float64,2}:
  -2.250e-02  -2.100e-01  -5.225e-01
  9.750e-02  -9.000e-02  -4.025e-01
  2.975e-01  1.100e-01  -2.025e-01
  5.775e-01  3.900e-01   7.750e-02

The approximate solution:
4×3 Array{Float64,2}:
  -2.250e-02  -2.100e-01  -5.225e-01
  9.750e-02  -9.000e-02  -4.025e-01
  2.975e-01  1.100e-01  -2.025e-01
  5.775e-01  3.900e-01   7.750e-02

Norm of difference: 3.914e-16
```
Why is the error so small?

Exercise 2:

1. Apply the central difference formulas to \( u(x, y) = x^2 - y^2 \).

2. Do the central differences formulas on \( x^2 - y^2 \) explain why the observed error is so small?
a second example of the Laplace equation

Consider the Laplace equation over \([0, 1] \times [1, 2]\):

\[
\begin{align*}
\Delta u &= 0 \quad \text{for all } x \in [0, 1], y \in [1, 2], \\
u(x, 1) &= \ln(x^2) \quad \text{for all } x \in [0, 1], \\
u(x, 2) &= \ln(x^2 + 4) \quad \text{for all } x \in [0, 1], \\
u(0, y) &= 2 \ln(y) \quad \text{for all } y \in [1, 2], \\
u(1, y) &= \ln(y^2 + 1) \quad \text{for all } y \in [1, 2].
\end{align*}
\]

A solution to this equation is \(u(x, y) = \ln(x^2 + y^2)\).
running on the second example

$ julia example2Laplace.jl

The exact solution :
3×3 Array{Float64,2}:
  4.855e-01  8.383e-01  1.139e+00
  5.947e-01  9.163e-01  1.198e+00
  7.538e-01  1.034e+00  1.288e+00

The approximate solution :
3×3 Array{Float64,2}:
  4.847e-01  8.376e-01  1.139e+00
  5.944e-01  9.159e-01  1.197e+00
  7.539e-01  1.034e+00  1.288e+00

Norm of difference : 1.258e-03
The approximate solution:

3×3 Array{Float64,2}:

$$\begin{array}{ccc}
4.847e-01 & 8.376e-01 & 1.139e+00 \\
5.944e-01 & 9.159e-01 & 1.197e+00 \\
7.539e-01 & 1.034e+00 & 1.288e+00
\end{array}$$

Norm of difference : 1.258e-03

\[ [a, b] = [0, 1], \ M = 3 \Rightarrow h = (b - a)/(M + 1) = 1/4 \]

\[ [c, d] = [1, 2], \ N = 3 \Rightarrow k = (d - c)/(N + 1) = 1/4 \]

The error of the central differences is \( O(h^2) + O(k^2) \).

\[ h^2 = \frac{1}{16} = k^2 \] and \( \frac{1}{16} = 6.25e-02 \geq 1.258e-03. \]

The observed error is below the theoretical error.
Exercise 3:
Consider the Laplace equation over \([0, 1] \times [0, 1]\):

\[
\begin{align*}
\Delta u &= 0 \quad \text{for all } x \in [0, 1], y \in [0, 1], \\
u(x, 0) &= \sin(\pi x) \quad \text{for all } x \in [0, 1], \\
u(x, 1) &= \sin(\pi x) \quad \text{for all } x \in [0, 1], \\
u(0, y) &= 0 \quad \text{for all } y \in [0, 1], \\
u(1, y) &= 0 \quad \text{for all } y \in [0, 1].
\end{align*}
\]

1. Solve this equation for \((M, N) = (10, 10)\) and for \((20, 20)\).
2. Compare the difference between the solution for \((10, 10)\) and \((20, 20)\). Can you explain the difference?
Elliptic Partial Differential Equations

1. Partial Differential Equations
   - Poisson and Laplace
   - a first example of the Laplace equation

2. the Finite Difference Method
   - a rectangular mesh of points
   - apply the lexicographical order
   - computational experiments

3. Conclusions
   - about numerical analysis
Numerical methods were applied to three examples of PDEs.

1. **The heat equation models the diffusion of heat.**

   Forward differences lead to an explicit, conditionally stable method. A stable and implicit method is obtained via backward differences, but the backward difference method is still not accurate enough. The method of Crank-Nicholson is both stable and accurate.

2. **The wave equation models the oscillation of a violin string.**

   Central differences give an accurate, conditionally stable method.

3. **The steady state equation models a temperature distribution.**

   Central differences define a linear system with predictable accuracy.
about numerical analysis

Definition (Nick Trefethen, SIAM News 1992)
Numerical analysis is the study of algorithms for the problems of continuous mathematics.

This course followed the traditional calculus based approach.

After the fundamentals on floating-point arithmetic, we covered root finding, linear system solving, interpolation, and data fitting.

The second part was oriented towards differential equations, involving the application of numerical differentiation and integration.

We study the efficiency and accuracy of numerically stable algorithms for well conditioned problems.