UW–Madison Math/CS 714

Methods of Computational Mathematics I

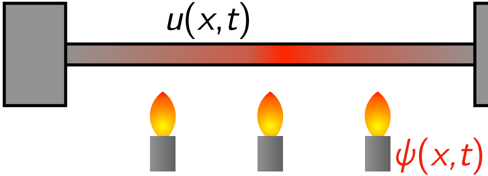
Elliptic equations I

Instructor: Yue Sun (yue.sun@wisc.edu)

September 16, 2025

Generalization to higher dimensions

The two-point boundary value problem example was based on looking at the steady state temperature distribution $u(x)$ of a one-dimensional rod



This led to the equation

$$u_{xx} = f$$

with boundary conditions of $u(a) = \alpha$ and $u(b) = \beta$.

Generalization to higher dimensions

There is a natural generalization of this to multiple dimensions, to find the steady state temperature distribution $u(x, y)$ in a domain $\Omega \subseteq \mathbb{R}^2$. Then

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

with Dirichlet boundary conditions $u = u_{\text{fix}}$ on the domain boundary $\partial\Omega$.

Elliptic partial differential equations

The equation

$$\nabla^2 u = u_{xx} + u_{yy} = f$$

for $u(x, y)$ is called the Poisson equation. When $f = 0$, the equation becomes

$$\nabla^2 u = u_{xx} + u_{yy} = 0,$$

which is the Laplace equation.

These equations arise in many contexts, such as heat conduction, electrostatics, gravitation, and probability theory. They are examples of elliptic equations

Digression: classification of PDEs[1]

There are three main classes of partial differential equations:

equation type	prototypical example	equation
hyperbolic	wave equation	$u_{tt} - u_{xx} = 0$
parabolic	heat equation	$u_t - u_{xx} = f$
elliptic	Poisson equation	$u_{xx} + u_{yy} = f$

Question: Where do these names come from?

[1] Covered in Appendix E of *Finite Difference Methods for Ordinary and Partial Differential Equations*

Digression: classification of PDEs

Answer: The names are related to conic sections

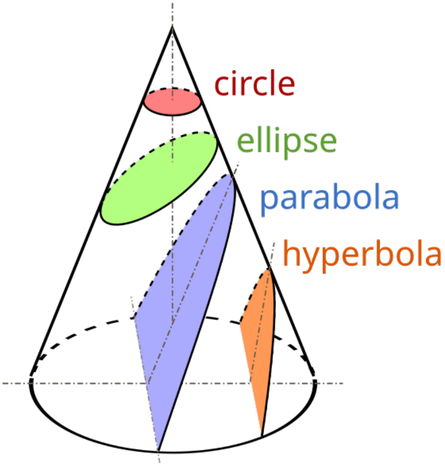
General second-order PDEs have the form

$$au_{xx} + bu_{xy} + cu_{yy} + du_x + eu_y + fu + g = 0$$

This "looks like" the quadratic function

$$q(x, y) = ax^2 + bxy + cy^2 + dx + ey$$

Digression: classification of PDEs



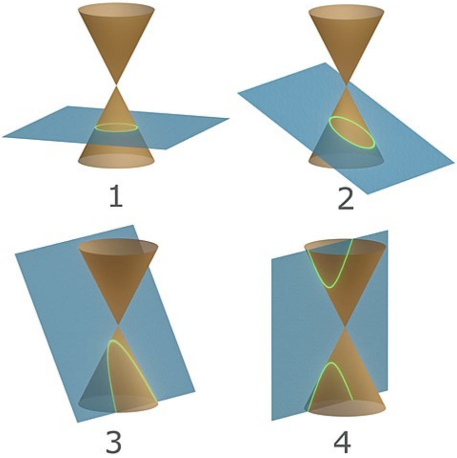


Figure: Conic sections[2]

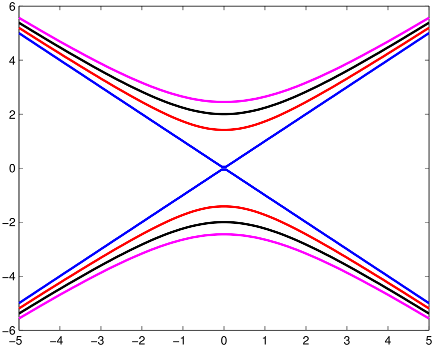
[2]Source: Wikipedia

PDEs: Hyperbolic

Wave equation: $u_{tt} - u_{xx} = 0$

Corresponding quadratic function is $q(x, t) = t^2 - x^2$

$q(x, t) = c$ gives a hyperbola, *e.g.* for $c = 0 : 2 : 6$, we have

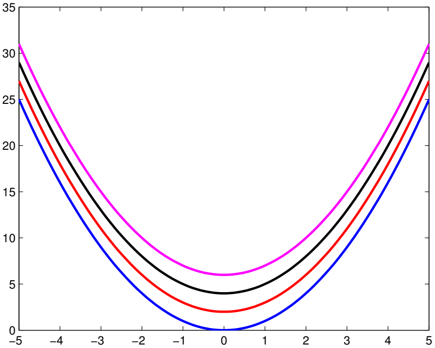


PDEs: Parabolic

Heat equation: $u_t - u_{xx} = 0$

Corresponding quadratic function is $q(x,t) = t - x^2$

$q(x,t) = c$ gives a parabola, *e.g.* for $c = 0 : 2 : 6$, we have

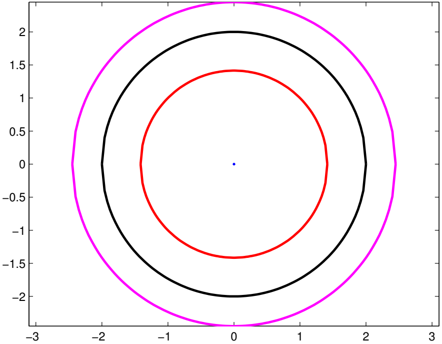


PDEs: Elliptic

Poisson equation: $u_{xx} + u_{yy} = f$

Corresponding quadratic function is $q(x, y) = x^2 + y^2$

$q(x, y) = c$ gives an ellipse, *e.g.* for $c = 0 : 2 : 6$, we have



PDEs

In general, it is not so easy to classify PDEs using conic section naming.

Many problems don't strictly fit into the classification scheme
(*e.g.* nonlinear, or higher order, or variable coefficient equations)

Nevertheless, the names hyperbolic, parabolic, elliptic are the standard ways of describing PDEs, based on the criteria:

- Hyperbolic: time-dependent, conservative physical process, no steady state
- Parabolic: time-dependent, dissipative physical process, evolves towards steady state
- Elliptic: describes systems at equilibrium/steady-state

Discretization of the Poisson equation

Consider solving the Poisson equation on the unit square $\Omega = [0, 1]^2$ with Dirichlet boundary conditions.

Introduce grid points (x_i, y_j) where $x_i = i\Delta x$ and $y_j = j\Delta y$. Let u_{ij} be the numerical approximation of $u(x_i, y_j)$, and define $f_{ij} = f(x_i, y_j)$.

Discretization of the Poisson equation

Using the centered-difference discretization, $\nabla^2 u = f$ can be written as

$$\frac{u_{i-1,j} - 2u_{ij} + u_{i+1,j}}{\Delta x^2} + \frac{u_{i,j-1} - 2u_{ij} + u_{i,j+1}}{\Delta y^2} = f_{ij}.$$

If $\Delta x = \Delta y = h$ then this can be written as

$$\frac{u_{i-1,j} + u_{i+1,j} + u_{i,j-1} + u_{i,j+1} - 4u_{i,j}}{h^2} = f_{ij}.$$

The left hand side is the five-point stencil for the Laplacian.

Linear system for the Poisson equation

For the case of equal grid spacing $\Delta x = \Delta y = h$, define m interior grid points in each direction, so that $h = 1/(m+1)$.

Values u_{ij} will be fixed by the Dirichlet boundary conditions when $i = 0$, $j = 0$, $i = m$, or $j = m$.

To solve at the interior grid points, we construct a linear system for the m^2 unknowns, given by for u_{ij} with $1 \leq i \leq m$ and $1 \leq j \leq m$.

We can now examine the accuracy and stability of this discretization, generalizing the methods introduced previously.

Linear system for the Poisson equation

See derivation: ordering the unknowns and equations

Code example #1

The program *poisson.py* solves the Poisson equation

$$\nabla^2 u = f$$

on the domain $\Omega = [0, 1]^2$ using

$$f(x, y) = \exp\left(-3\left((x - 0.3)^2 + (y - 0.7)^2\right)\right)$$

and

$$u = 0 \qquad \text{on } \partial\Omega.$$