NOTES AND STUDIES IN SPECTRAL METHODS

Computational Études

A Spectral Approach

Dr. Denys Dutykh

Mathematics Department
College of Computing and Mathematical Sciences
Khalifa University of Science and Technology
Abu Dhabi, UAE

2026

Contents

Preface 2
1 Introduction 5
  1.1 The Spectral Promise 6
  1.2 The Philosophy of “Études” 7
  1.3 Collocation: Computing in Physical Space 7
  1.4 A Modern Workflow 8
2 Classical Second Order PDEs and Separation of Variables 10
  2.1 Heat Equation with Periodic Boundary Conditions 11
  2.2 Numerical Illustration 15
  2.3 Wave Equation with Dirichlet Boundary Conditions 17
  2.4 Numerical Illustration 21
  2.5 Laplace Equation in a Periodic Strip 24
  2.6 Numerical Illustration 28
  2.7 Conclusions 30

Preface

The purpose of this book is twofold: to explain to students why spectral methods work and why they are so remarkably efficient, and to teach them how to implement these methods using modern computational tools.

This text is conceived as a collection of **Computational Études**. In musical education, an étude is a composition designed to perfect a specific technique — be it rapid scales, complex arpeggios, or delicate phrasing — while remaining a pleasing piece of music in its own right. Similarly, each chapter here presents a focused mathematical concept paired with its computational realization: a study that is both instructive and complete.

Who Is This Book For?

We have written primarily for graduate students in applied mathematics, physics, and engineering who seek a hands-on understanding of spectral methods. The reader should be comfortable with calculus, linear algebra, and basic programming. Prior exposure to numerical methods is helpful but not essential — we build the necessary foundations as we proceed.

How Is the Book Organized?

Each chapter is designed to be largely self-contained. We begin with fundamental concepts — interpolation and differentiation — before advancing to time-stepping schemes and applications. The mathematical exposition is deliberately concise, favoring clarity over exhaustive rigor. Proofs are included when they illuminate; otherwise, we direct the reader to authoritative references.

Reproducible Science

This book is also an experiment in **reproducible science**. Every figure, every table, every numerical result you see in these pages was generated by code available in the accompanying repository. We provide implementations in both Python and MATLAB, allowing readers to choose their preferred environment. The Python code emphasizes accessibility and integration with the open-source scientific ecosystem; the MATLAB code leverages its historical significance in numerical computing and, where appropriate, the Advanpix Multiprecision Computing Toolbox for extended precision arithmetic.

How to Use This Book

We invite you to treat this book not as a static reference, but as a workshop. Clone the repository, run the scripts, modify the parameters, break the code, and fix it. That is the only way to truly master the art of spectral methods.

The complete source code and the Typst manuscript are available at:
https://github.com/dutykh/computational-etudes/

Dr. Denys Dutykh

Abu Dhabi, UAE
January 2026

CHAPTER 1

Introduction

Differential equations serve as the fundamental language of the physical sciences, describing phenomena ranging from the propagation of sound waves to the flow of heat and the dynamics of fluids. Finding exact analytical solutions to these equations is a luxury rarely afforded in practical applications. Consequently, the scientist and the engineer must turn to numerical approximation.

Broadly speaking, numerical methods for differential equations fall into two categories: local methods and global methods. The former, including Finite Difference and Finite Element Methods, approximate the unknown solution using functions that are non-zero only on small sub-domains (elements or grid stencils). These methods are robust and flexible, handling complex geometries with grace. However, their accuracy is typically algebraic; refining the grid by a factor of two might improve the error by a factor of four or eight, but rarely more. From a computational perspective, local methods are *myopic*: to compute a derivative at a grid point, they look only at immediate neighbors.

Spectral methods represent the global approach. They approximate the solution as a linear combination of continuous, global basis functions—typically trigonometric polynomials (Fourier series) for periodic problems or Chebyshev polynomials for non-periodic ones. In stark contrast to local schemes, spectral methods are *holistic*: the derivative at any single point depends on the function values at *every other point* in the domain. Mathematically, this is equivalent to fitting a single high-degree polynomial through all data points. This global coupling is what allows information to propagate instantly across the grid, granting us the remarkable convergence that we call “spectral accuracy.”

1.1 The Spectral Promise

The fundamental argument for spectral methods is one of efficiency. If the solution to a problem is smooth, the coefficients of its expansion in a proper global basis decay exponentially fast. This phenomenon is known as spectral accuracy.

In practical terms, this means that spectral methods can achieve a level of precision with a few dozen degrees of freedom that a finite difference scheme might require thousands of grid points to match. While a fourth-order finite difference method implies that the error $\varepsilon \sim O(N^{-4})$, a spectral method boasts $\varepsilon \sim O(c^{-N})$ for some constant $c > 1$. When the solution is analytic, the convergence is explosive; the error drops into the “spectral valley” until it hits the floor of machine precision.

This global dependence has a computational consequence: spectral differentiation matrices are *dense*, not sparse. Where a finite difference scheme produces banded matrices that are cheap to store and invert, spectral methods fill in every entry. However, the extraordinary accuracy means we need so few points (often just dozens where finite differences would require thousands) that we can afford this density. The cost per degree of freedom is higher, but the total cost for a given accuracy is dramatically lower.

However, this power is not without its price. Spectral methods are unforgiving regarding grid placement. We cannot simply choose points where we please; for non-periodic problems, the mathematics dictates that points must cluster at boundaries (the celebrated Chebyshev points) to prevent the interpolation from diverging. Attempting high-degree polynomial interpolation on an equispaced grid leads to the notorious Runge phenomenon, where oscillations grow without bound near the boundaries. This sensitivity to geometry is what restricts spectral methods primarily to simple domains, but within those domains, they reign supreme.

This book aims to demystify this “spectral magic.” We will see that it is not magic at all, but a direct consequence of the smoothness of the underlying functions and the careful choice of basis and grid.

1.2 The Philosophy of “Études”

The title of this volume, Computational Études, reflects a specific pedagogical philosophy. In musical education, an étude is a composition designed to practice a particular technical skill—be it rapid scales or complex arpeggios—while remaining a pleasing piece of music in its own right.

In this text, our “technical skills” are not rapid scales or arpeggios, but rather handling stiffness in time-stepping, managing aliasing in nonlinear products, enforcing boundary conditions through tau methods or lifting functions, and filtering spurious oscillations. Just as a Chopin Étude transforms a technical exercise into art, a well-written spectral code transforms a mathematical formula into a robust simulation. The études collected here are designed to cultivate this virtuosity.

We approach spectral methods not through dry, abstract theorems, but through concrete, self-contained studies. Each chapter focuses on a specific mathematical concept—interpolation, differentiation, aliasing, or time-stepping—and explores it through a compact, runnable implementation.

We deliberately restrict our focus primarily to one-dimensional problems. This choice is strategic. The mathematical essence of spectral methods—the treatment of boundaries, the distribution of collocation points, and the structure of differentiation matrices—is fully present in one dimension. Extending these ideas to two or three dimensions usually involves tensor products, which add significant programming overhead without necessarily adding new conceptual depth. By staying in 1D, we keep our code short, readable, and focused on the physics and mathematics.

1.3 Collocation: Computing in Physical Space

While the theory of spectral methods relies on orthogonal expansions — Fourier series for periodic problems, Chebyshev series otherwise — the actual computation often proceeds differently. Rather than manipulating expansion coefficients directly (the *modal* or *Galerkin* approach), we typically work with function values at carefully chosen grid points (the *nodal* or *collocation* approach, also called *pseudospectral*).

The collocation philosophy dominates this book for a practical reason: it handles nonlinear terms with ease. When the governing equation contains products like $u \cdot u_x$, the modal approach requires computing convolutions of coefficient sequences — a tedious operation. Collocation simply evaluates the product pointwise on the grid. This directness makes pseudospectral methods the tool of choice for most computational applications, and it is the approach we shall master through these études.

The reader should be aware that both viewpoints — modal and nodal — illuminate the same underlying mathematics. The Fast Fourier Transform provides the bridge, allowing us to move efficiently between coefficient space and physical space as needed.

1.4 A Modern Workflow

Finally, this book is an experiment in reproducible science. The days of presenting numerical results as static, unverifiable images are passing. The results you see in these pages were generated by the code available in the accompanying repository. We utilize a dual-language approach:

- **Python**: For accessibility and integration with the vast open-source scientific ecosystem.
- **Matlab**: For its historical significance in this field and its concise matrix syntax, often utilizing the Advanpix Multiprecision Computing Toolbox to explore phenomena that lie beyond standard double precision.

We invite you to treat this book not as a static reference, but as a workshop. Run the scripts, change the parameters, break the code, and fix it. That is the only way to truly learn the art of spectral methods.

CHAPTER 2

Classical Second Order PDEs and Separation of Variables

In this opening chapter we derive exact solutions for three classical linear partial differential equations: the *heat equation* (parabolic), the *wave equation* (hyperbolic), and the *Laplace equation* (elliptic). These solutions are found by the *method of separation of variables*, which expresses the solution as an infinite series of eigenfunctions.

Why begin a book on *numerical* methods with *analytical* solutions? Because separation of variables is the theoretical ancestor of spectral methods. When we later truncate these infinite series at some finite N and compute with only the first N modes, we are doing exactly what a spectral solver does—but with pen and paper first. This chapter thus serves as the conceptual bridge between classical analysis and modern computation.

We treat three model problems:

- heat equation with periodic boundary conditions in one spatial dimension,
- wave equation on a bounded interval,
- Laplace equation in a simple domain.

We begin with a complete analytic solution of the heat equation. The other two examples will follow the same pattern.

2.1 Heat Equation with Periodic Boundary Conditions

We consider the one dimensional heat equation on the interval $[0, 2\pi]$ with periodic boundary conditions. The unknown $u(x,t)$ represents, for example, the temperature at point x and time t.

The problem is

$$\frac{\partial u}{\partial t}(x,t) = \frac{\partial^2 u}{\partial x^2}(x,t), \quad x \in [0, 2\pi],\ t > 0,$$

with periodic boundary conditions

$$u(x + 2\pi, t) = u(x,t)$$

for all real x and all $t > 0$, and initial condition

$$u(x,0) = f(x), \quad x \in [0, 2\pi].$$

We assume that f is smooth and 2π periodic:

$$f(x + 2\pi) = f(x).$$

Our goal is to obtain an explicit representation of $u(x,t)$ as an infinite series and to see how separation of variables leads naturally to a Fourier series in space.

2.1.1 Step 1: Separation Ansatz

We look for nontrivial solutions of the form

$$u(x,t) = X(x) \cdot T(t),$$

where X depends only on x and T depends only on t.

Substituting into the PDE gives

$$X(x) \cdot T'(t) = X''(x) \cdot T(t).$$

We assume X and T are not identically zero, so we can divide both sides by $X(x) \cdot T(t)$:

$$\frac{T'(t)}{T(t)} = \frac{X''(x)}{X(x)}.$$

The left side depends only on t, the right side only on x. Therefore both sides must be equal to the same constant, which we denote by $-\lambda$:

$$\frac{T'(t)}{T(t)} = \frac{X''(x)}{X(x)} = -\lambda.$$

We obtain two ordinary differential equations:

$$T'(t) + \lambda T(t) = 0,$$

$$X''(x) + \lambda X(x) = 0.$$

The periodic boundary conditions for u imply periodic conditions for X:

$$X(0) = X(2\pi), \quad X'(0) = X'(2\pi).$$

We have arrived at a spatial eigenvalue problem for X.

2.1.2 Step 2: Spatial Eigenvalue Problem with Periodic Boundary Conditions

We now solve

$$X''(x) + \lambda X(x) = 0,$$

with

$$X(0) = X(2\pi), \quad X'(0) = X'(2\pi).$$

We consider three cases: $\lambda < 0$, $\lambda = 0$, and $\lambda > 0$.

Case 1: $\boldsymbol{\lambda < 0}$

Write $\lambda = -\mu^2$ with $\mu > 0$. The equation becomes

$$X''(x) - \mu^2 X(x) = 0.$$

The general solution is

$$X(x) = Ae^{\mu x} + Be^{-\mu x}.$$

Imposing periodicity $X(0) = X(2\pi)$ gives

$$A + B = Ae^{2\mu\pi} + Be^{-2\mu\pi}.$$

Imposing $X'(0) = X'(2\pi)$ gives

$$\mu(A - B) = \mu(Ae^{2\mu\pi} - Be^{-2\mu\pi}).$$

Since $\mu > 0$, we can divide by μ and rewrite both conditions as a homogeneous linear system:

$$A(1 - e^{2\mu\pi}) + B(1 - e^{-2\mu\pi}) = 0,$$

$$A(1 - e^{2\mu\pi}) - B(1 - e^{-2\mu\pi}) = 0.$$

Adding these two equations gives

$$2A(1 - e^{2\mu\pi}) = 0.$$

Since $\mu > 0$, we have $e^{2\mu\pi} > 1$, so $1 - e^{2\mu\pi} \neq 0$. Therefore $A = 0$.

Subtracting the second equation from the first gives

$$2B(1 - e^{-2\mu\pi}) = 0.$$

Since $\mu > 0$, we have $e^{-2\mu\pi} < 1$, so $1 - e^{-2\mu\pi} \neq 0$. Therefore $B = 0$.

Hence the only solution is $A = B = 0$, the trivial solution. There are no nontrivial periodic eigenfunctions for $\lambda < 0$, and we discard this case.

Case 2: $\boldsymbol{\lambda = 0}$

The equation reduces to

$$X''(x) = 0.$$

Its general solution is

$$X(x) = A + Bx.$$

Periodicity $X(0) = X(2\pi)$ gives

$$A = A + 2\pi B$$

so $B = 0$. Then $X(x) = A$ is constant.

The derivative is $X'(x) = 0$, so $X'(0) = X'(2\pi)$ is automatically satisfied.

Thus $\lambda = 0$ gives one eigenfunction

$$X_0(x) = 1$$

(up to a multiplicative constant).

Case 3: $\boldsymbol{\lambda > 0}$

Write $\lambda = k^2$ with $k > 0$. The equation becomes

$$X''(x) + k^2 X(x) = 0.$$

The general solution is

$$X(x) = A\cos(kx) + B\sin(kx).$$

We now impose periodicity. First,

$$X(0) = A, \quad X(2\pi) = A\cos(2\pi k) + B\sin(2\pi k).$$

The condition $X(0) = X(2\pi)$ gives

$$A = A\cos(2\pi k) + B\sin(2\pi k).$$

Next,

$$X'(x) = -Ak\sin(kx) + Bk\cos(kx),$$

so

$$X'(0) = Bk, \quad X'(2\pi) = -Ak\sin(2\pi k) + Bk\cos(2\pi k).$$

The condition $X'(0) = X'(2\pi)$ gives

$$Bk = -Ak\sin(2\pi k) + Bk\cos(2\pi k).$$

We can divide by k the last identity (for $k \neq 0$) and write the system as

$$A(1 - \cos(2\pi k)) - B\sin(2\pi k) = 0,$$

$$A\sin(2\pi k) + B(1 - \cos(2\pi k)) = 0.$$

For a nontrivial pair (A, B) the determinant of the system must vanish:

$$(1 - \cos(2\pi k))^2 + (\sin(2\pi k))^2 = 0.$$

The left side is a sum of squares, so it is zero if and only if

$$1 - \cos(2\pi k) = 0, \quad \sin(2\pi k) = 0.$$

Hence

$$\cos(2\pi k) = 1, \quad \sin(2\pi k) = 0.$$

This happens exactly when k is an integer:

$$k = n, \quad n \in \mathbb{Z}.$$

The case $n = 0$ corresponds to $\lambda = 0$, which we have already treated. For $n \geqslant 1$ we obtain eigenvalues

$$\lambda_n = n^2, \quad n = 1, 2, 3, \ldots$$

For each $n \geqslant 1$ the corresponding eigenfunctions can be chosen as

$$X_n^{(c)}(x) = \cos(nx), \quad X_n^{(s)}(x) = \sin(nx).$$

These functions are 2π periodic, and their derivatives are also 2π periodic, so the boundary conditions are satisfied.

We have therefore found a complete set of spatial eigenfunctions for the heat equation with periodic boundary conditions:

- a constant mode $X_0(x) = 1$ (eigenvalue $\lambda_0 = 0$),
- cosine modes $X_n^{(c)}(x) = \cos(nx)$,
- sine modes $X_n^{(s)}(x) = \sin(nx)$,

with eigenvalues $\lambda_n = n^2$ for $n \geqslant 1$.

2.1.3 Step 3: Time Dependent Factors

For each eigenvalue λ the corresponding time factor satisfies

$$T'(t) + \lambda T(t) = 0.$$

The solution is

$$T(t) = Ce^{-\lambda t}.$$

For the constant mode $\lambda_0 = 0$ we obtain

$$T_0(t) = C_0$$

(constant in time).

For $n \geqslant 1$ we have

$$T_n(t) = C_n e^{-n^2 t}.$$

Combining space and time, we form products $u(x,t) = X(x) \cdot T(t)$ to obtain separated solutions. For each mode, the arbitrary constant from the time factor can be absorbed into a single new constant. Specifically, we write

$$\begin{aligned} u_0(x,t) &= A_0, \\ u_n^{(c)}(x,t) &= A_n \cos(nx) \cdot e^{-n^2 t}, \\ u_n^{(s)}(x,t) &= B_n \sin(nx) \cdot e^{-n^2 t}, \end{aligned}$$

where we have renamed the constants: $A_0 = C_0$, and for $n \geqslant 1$, the constant A_n absorbs the coefficient from the cosine mode while B_n absorbs the coefficient from the sine mode. Note that each separated solution carries its own independent arbitrary constant.

Since the heat equation is linear, any linear combination of these separated solutions is again a solution. Therefore the general solution that satisfies the periodic boundary conditions can be written as an infinite series

$$u(x,t) = A_0 + \sum_{n=1}^{\infty} (A_n \cos(nx) + B_n \sin(nx)) e^{-n^2 t}.$$

The coefficients A_0, A_n, B_n remain to be determined from the initial condition.

2.1.4 Step 4: Imposing the Initial Condition and Fourier Series

We impose the initial condition

$$u(x,0) = f(x).$$

Setting $t = 0$ in the general solution we obtain

$$u(x,0) = A_0 + \sum_{n=1}^{\infty} (A_n \cos(nx) + B_n \sin(nx)) = f(x).$$

This is exactly the Fourier series expansion of the 2π periodic function f. Under mild regularity assumptions, f has a Fourier series

$$f(x) = a_0 + \sum_{n=1}^{\infty} (a_n \cos(nx) + b_n \sin(nx))$$

with Fourier coefficients

$$\begin{aligned} a_0 &= \frac{1}{2\pi} \int_0^{2\pi} f(x)\,\mathrm{d}x, \\ a_n &= \frac{1}{\pi} \int_0^{2\pi} f(x) \cos(nx)\,\mathrm{d}x, \quad n \geqslant 1, \\ b_n &= \frac{1}{\pi} \int_0^{2\pi} f(x) \sin(nx)\,\mathrm{d}x, \quad n \geqslant 1. \end{aligned}$$

By uniqueness of the Fourier expansion, we must have

$$A_0 = a_0, \quad A_n = a_n, \quad B_n = b_n.$$

Thus the coefficients in the heat equation solution are exactly the Fourier coefficients of the initial data.

2.1.5 Step 5: Final Explicit Solution and Interpretation

Substituting these coefficients into the general solution we obtain the explicit formula

$$u(x,t) = a_0 + \sum_{n=1}^{\infty} (a_n \cos(nx) + b_n \sin(nx)) e^{-n^2 t}, \quad t > 0.$$

This infinite sum solves the heat equation with periodic boundary conditions and initial data f. Each Fourier mode decays exponentially in time at a rate proportional to its eigenvalue n^2. High frequency modes (large n) decay faster, which expresses the smoothing effect of the heat equation.

The constant term a_0 does not decay. It represents the average value of f on $[0, 2\pi]$, which is preserved by the heat flow.

From a spectral viewpoint, the functions

$$1, \cos(x), \sin(x), \cos(2x), \sin(2x), \ldots$$

form an eigenbasis of the spatial operator

$$Lu = \frac{\partial^2 u}{\partial x^2}$$

with periodic boundary conditions. The evolution of each eigenmode is independent and given simply by multiplication by $e^{-n^2 t}$ in time.

Later, in the numerical part of this book, we will approximate $u(x,t)$ by truncating the sum to finitely many modes:

$$u_{N(x,t)} = a_0 + \sum_{n=1}^{N} (a_n \cos(nx) + b_n \sin(nx)) e^{-n^2 t}.$$

This truncation is the essence of a Fourier spectral method. The analytic solution derived here is the infinite dimensional limit of that numerical procedure.

2.2 Numerical Illustration

To visualize the smoothing effect of heat diffusion, we compute the truncated Fourier series solution for a triangle wave initial condition:

$$f(x) = \pi - |x - \pi|, \quad x \in [0, 2\pi].$$

This function is continuous but has a corner (non-differentiable point) at $x = \pi$. Its Fourier series contains only cosine terms with coefficients decaying as $1/n^2$:

$$f(x) = \frac{\pi}{2} + \frac{4}{\pi} \sum_{k=1}^{\infty} \frac{\cos((2k-1)x)}{(2k-1)^2}.$$

The key portion of the implementation evaluates the truncated Fourier series at any point in space and time. In Python:

```python
def heat_solution(x, t, a0, a_n, b_n):
    u = np.full_like(x, a0, dtype=float)
    n_modes = len(a_n) - 1
    for n in range(1, n_modes + 1):
        decay = np.exp(-n**2 * t)
        u += (a_n[n] * np.cos(n * x) + b_n[n] * np.sin(n * x)) * decay
    return u
```

The equivalent MATLAB implementation:

Matlab

```
u = a0 * ones(size(x));
for n = 1:N_MODES
    decay = exp(-n^2 * t);
    u = u + (a_n(n+1) * cos(n*x) + b_n(n+1) * sin(n*x)) * decay;
end
```

Figure 1 shows the evolution of $u_N(x,t)$ with $N = 50$ modes at several time values. At $t = 0$ the triangle wave is faithfully reproduced. As time increases, the higher frequency modes decay exponentially faster than the lower ones (the n-th mode decays as $e^{-n^2 t}$), and the solution rapidly smooths toward the constant equilibrium $u = \pi/2$.

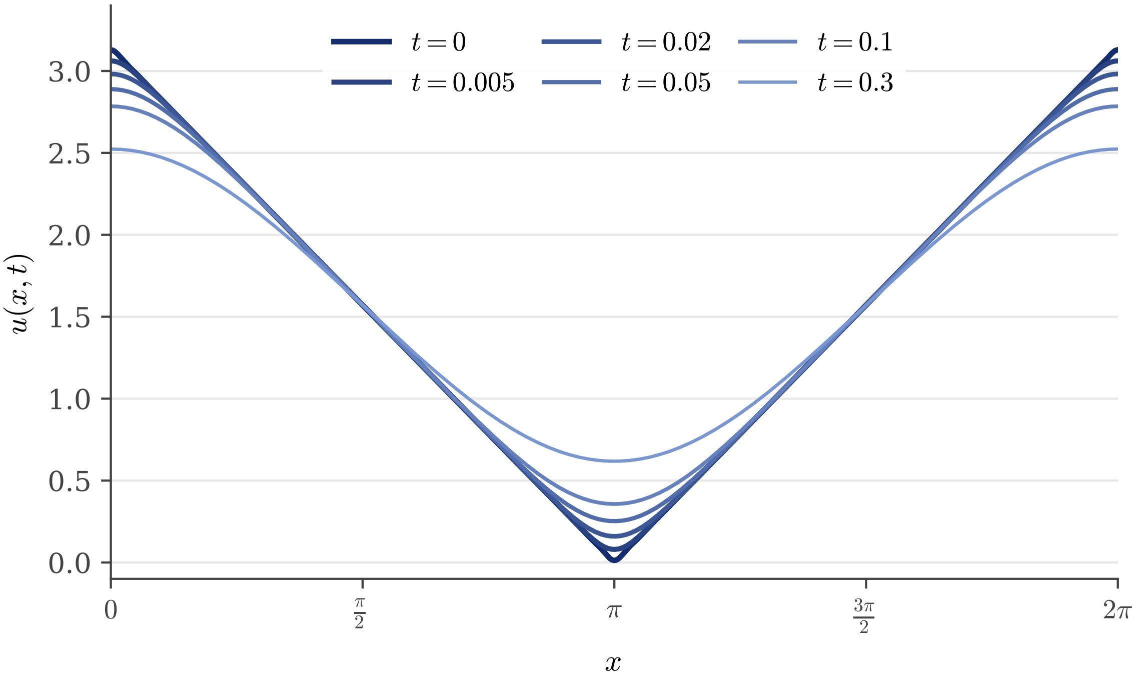

Figure 1: Evolution of the heat equation solution with a triangle wave initial condition. The higher frequency modes decay rapidly, smoothing the initial corner at $x = \pi$.

The code that generated this figure is available in both Python and MATLAB:

- `codes/python/ch02_classical_pdes/heat_equation_evolution.py`
- `codes/matlab/ch02_classical_pdes/heat_equation_evolution.m`

A complementary view of the solution is provided by the waterfall plot in Figure 2, which displays the entire space-time evolution as a three-dimensional surface. The smoothing effect of the heat equation is clearly visible: the initial sharp triangle wave rapidly flattens as time progresses, with the solution approaching the constant equilibrium state $u = \pi/2$.

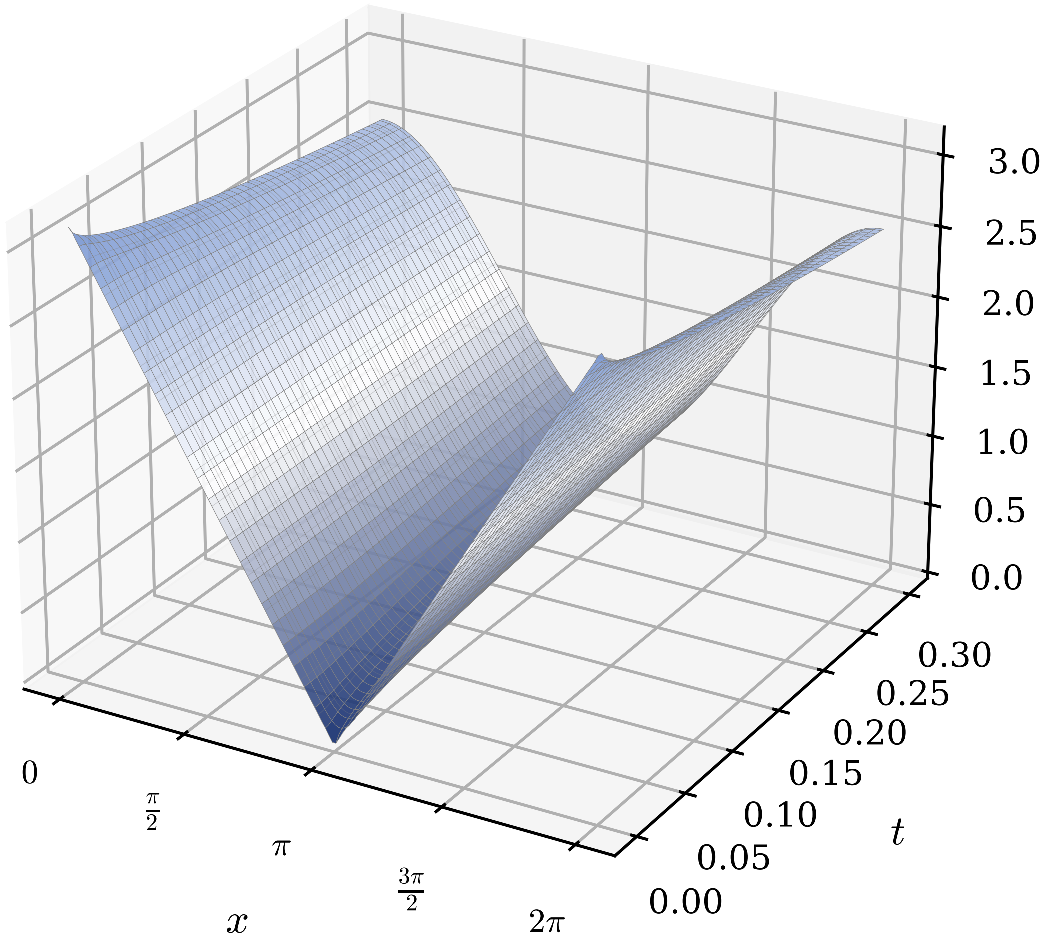

Figure 2: Waterfall plot showing the complete space-time evolution of the heat equation solution. The initial triangle wave smooths rapidly as higher frequency modes decay exponentially.

2.3 Wave Equation with Dirichlet Boundary Conditions

We now consider the one dimensional wave equation on a finite interval with homogeneous Dirichlet boundary conditions. This is the classical model for a vibrating string of length L with both ends fixed.

Let $u(x,t)$ denote the vertical displacement of the string at position $x \in [0,L]$ and time $t > 0$. The equation of motion is

$$\frac{\partial^2 u}{\partial t^2}(x,t) = c^2 \frac{\partial^2 u}{\partial x^2}(x,t), \quad 0 < x < L,\ t > 0,$$

where $c > 0$ is the wave speed.

The boundary conditions express that the endpoints of the string are clamped:

$$u(0,t) = 0, \quad u(L,t) = 0, \quad t > 0.$$

We prescribe the initial displacement and initial velocity:

$$u(x,0) = f(x), \quad \frac{\partial u}{\partial t}(x,0) = g(x), \quad 0 < x < L,$$

with suitable functions f and g that vanish at $x = 0$ and $x = L$.

As in the heat equation example, we use separation of variables and obtain a representation of the solution as an infinite series in spatial eigenfunctions. This time the temporal factors are oscillatory instead of decaying.

2.3.1 Step 1: Separation Ansatz

We look for nontrivial solutions of the form

$$u(x,t) = X(x) \cdot T(t).$$

Substituting into the wave equation gives

$$X(x) \cdot T''(t) = c^2 X''(x) \cdot T(t).$$

Assuming X and T are not identically zero, we divide both sides by $c^2 X(x) \cdot T(t)$:

$$\frac{T''(t)}{c^2 T(t)} = \frac{X''(x)}{X(x)}.$$

The left side depends only on t, the right side only on x. Therefore both sides must be equal to a constant, which we denote by $-\lambda$:

$$\frac{T''(t)}{c^2 T(t)} = \frac{X''(x)}{X(x)} = -\lambda.$$

We obtain the pair of ordinary differential equations

$$T''(t) + c^2 \lambda T(t) = 0,$$
$$X''(x) + \lambda X(x) = 0,$$

with boundary conditions

$$X(0) = 0, \quad X(L) = 0.$$

As in the heat equation case, we have a spatial eigenvalue problem for X.

2.3.2 Step 2: Spatial Eigenvalue Problem with Dirichlet Boundary Conditions

We must solve

$$X''(x) + \lambda X(x) = 0, \quad 0 < x < L,$$

with

$$X(0) = 0, \quad X(L) = 0.$$

We again consider three cases: $\lambda < 0$, $\lambda = 0$, and $\lambda > 0$.

Case 1: $\lambda < 0$

Write $\lambda = -\mu^2$ with $\mu > 0$. The equation becomes

$$X''(x) - \mu^2 X(x) = 0.$$

The general solution is

$$X(x) = Ae^{\mu x} + Be^{-\mu x}.$$

The boundary condition at $x = 0$ gives

$$X(0) = A + B = 0 \quad \Rightarrow \quad B = -A.$$

Then

$$X(L) = Ae^{\mu L} - Ae^{-\mu L} = A\big(e^{\mu L} - e^{-\mu L}\big).$$

The condition $X(L) = 0$ implies

$$A\big(e^{\mu L} - e^{-\mu L}\big) = 0.$$

Since $e^{\mu L} \neq e^{-\mu L}$ for $\mu > 0$, we must have $A = 0$. Then $B = 0$ and the solution is trivial. Therefore there are no nontrivial eigenfunctions for $\lambda < 0$.

Case 2: $\lambda = 0$

The equation reduces to

$$X''(x) = 0,$$

whose general solution is

$$X(x) = A + Bx.$$

The boundary conditions give

$$X(0) = A = 0,$$

$$X(L) = A + BL = BL = 0.$$

Hence $B = 0$ and X is again trivial. There is no nontrivial eigenfunction for $\lambda = 0$.

Case 3: $\lambda > 0$

Write $\lambda = k^2$ with $k > 0$. The equation becomes

$$X''(x) + k^2 X(x) = 0.$$

The general solution is

$$X(x) = A\cos(kx) + B\sin(kx).$$

The boundary condition at $x = 0$ gives

$$X(0) = A = 0.$$

So $X(x) = B\sin(kx)$. The boundary condition at $x = L$ gives

$$X(L) = B\sin(kL) = 0.$$

For a nontrivial solution we need $B \neq 0$, so we must have

$$\sin(kL) = 0.$$

Therefore

$$kL = n\pi, \quad n = 1, 2, 3, \ldots$$

The corresponding values of k are

$$k_n = \frac{n\pi}{L}, \quad n = 1, 2, 3, \ldots$$

We conclude that the eigenvalues and eigenfunctions are

$$\lambda_n = k_n^2 = \left(\frac{n\pi}{L}\right)^2,$$

$$X_n(x) = \sin\left(\frac{n\pi x}{L}\right), \quad n = 1, 2, 3, \ldots$$

Each X_n vanishes at $x = 0$ and $x = L$, as required by the Dirichlet boundary conditions. The family $\{X_n\}_{n \geqslant 1}$ is orthogonal in $L^2(0, L)$:

$$\int_0^L \sin\left(\frac{n\pi x}{L}\right)\sin\left(\frac{m\pi x}{L}\right) \mathrm{d}x = \begin{cases} 0 & \text{if } n \neq m \\ L/2 & \text{if } n = m. \end{cases}$$

These eigenfunctions will form the spatial basis in our series solution.

2.3.3 Step 3: Time Dependent Factors

For each eigenvalue λ_n the time factor T_n satisfies

$$T_{n''}(t) + c^2 \lambda_n T_n(t) = 0.$$

Using $\lambda_n = (n\pi/L)^2$ we can write

$$T_{n''}(t) + \omega_n^2 T_n(t) = 0,$$

where

$$\omega_n = c\frac{n\pi}{L}, \quad n = 1, 2, 3, \ldots$$

The general solution of this second order linear ODE is

$$T_n(t) = A_n \cos(\omega_n t) + B_n \sin(\omega_n t),$$

where A_n and B_n are constants.

Combining the space and time factors, we obtain separated solutions

$$u_n(x,t) = (A_n \cos(\omega_n t) + B_n \sin(\omega_n t)) \sin\left(\frac{n\pi x}{L}\right), \quad n = 1, 2, 3, \ldots$$

Because the wave equation is linear, any linear combination of these separated solutions is again a solution. Therefore the general solution satisfying the Dirichlet boundary conditions can be written as an infinite series

$$u(x,t) = \sum_{n=1}^{\infty} (a_n \cos(\omega_n t) + b_n \sin(\omega_n t)) \sin\left(\frac{n\pi x}{L}\right),$$

for suitable coefficients a_n and b_n.

These coefficients will be determined from the initial conditions.

2.3.4 Step 4: Imposing the Initial Conditions and Sine Series

We now use the initial displacement and velocity.

At $t = 0$ we have

$$u(x,0) = \sum_{n=1}^{\infty} (a_n \cos(0) + b_n \sin(0)) \sin\left(\frac{n\pi x}{L}\right) = \sum_{n=1}^{\infty} a_n \sin\left(\frac{n\pi x}{L}\right).$$

The initial condition $u(x,0) = f(x)$ becomes

$$f(x) = \sum_{n=1}^{\infty} a_n \sin\left(\frac{n\pi x}{L}\right).$$

This is the Fourier sine series of f on the interval $(0, L)$.

Similarly, we differentiate u with respect to t:

$$\frac{\partial u}{\partial t}(x,t) = \sum_{n=1}^{\infty} (-a_n \omega_n \sin(\omega_n t) + b_n \omega_n \cos(\omega_n t)) \sin\left(\frac{n\pi x}{L}\right).$$

Evaluating at $t = 0$ gives

$$\frac{\partial u}{\partial t}(x,0) = \sum_{n=1}^{\infty} b_n \omega_n \sin\left(\frac{n\pi x}{L}\right).$$

The initial condition $\frac{\partial u}{\partial t}(x,0) = g(x)$ becomes

$$g(x) = \sum_{n=1}^{\infty} b_n \omega_n \sin\left(\frac{n\pi x}{L}\right).$$

So the sequence $\{a_n\}$ consists of the Fourier sine coefficients of f, and the sequence $\{b_n \omega_n\}$ consists of the Fourier sine coefficients of g.

Using the orthogonality relations, we obtain explicit formulas for the coefficients. For $n \geqslant 1$,

$$a_n = \frac{2}{L} \int_0^L f(x) \sin\left(\frac{n\pi x}{L}\right) \mathrm{d}x,$$

and

$$b_n \omega_n = \frac{2}{L} \int_0^L g(x) \sin\left(\frac{n\pi x}{L}\right) \mathrm{d}x.$$

Therefore

$$b_n = \frac{2}{L\omega_n} \int_0^L g(x) \sin\left(\frac{n\pi x}{L}\right) \mathrm{d}x = \frac{2}{L} \frac{1}{cn\pi/L} \int_0^L g(x) \sin\left(\frac{n\pi x}{L}\right) \mathrm{d}x.$$

In summary,

$$a_n = \frac{2}{L}\int_0^L f(x)\sin\Big(\frac{n\pi x}{L}\Big)\,\mathrm{d}x,$$
$$b_n = \frac{2}{L\omega_n}\int_0^L g(x)\sin\Big(\frac{n\pi x}{L}\Big)\,\mathrm{d}x, \quad \omega_n = c\frac{n\pi}{L}.$$

2.3.5 Step 5: Final Explicit Solution and Interpretation

Substituting these coefficients into the series, we obtain the explicit solution of the wave equation with Dirichlet boundary conditions:

$$u(x,t) = \sum_{n=1}^{\infty}[a_n\cos(\omega_n t) + b_n\sin(\omega_n t)]\sin\Big(\frac{n\pi x}{L}\Big),$$

where $\omega_n = cn\pi/L$ and the Fourier sine coefficients are

$$a_n = \frac{2}{L}\int_0^L f(y)\sin\Big(\frac{n\pi y}{L}\Big)\,\mathrm{d}y, \quad b_n = \frac{2}{n\pi c}\int_0^L g(y)\sin\Big(\frac{n\pi y}{L}\Big)\,\mathrm{d}y.$$

Each term in the sum is a normal mode of vibration: a standing wave with spatial shape $\sin(n\pi x/L)$ and temporal oscillation at frequency ω_n. The coefficients of $\cos(\omega_n t)$ and $\sin(\omega_n t)$ are determined by the initial displacement f and initial velocity g through their Fourier sine coefficients.

Comparing with the heat equation:

- For the heat equation, each mode decayed like $e^{-n^2 t}$ and the solution became smoother in time.
- For the wave equation, each mode oscillates periodically in time with constant amplitude, reflecting conservation of energy in the undamped string.

From a spectral viewpoint, the functions

$$\sin\Big(\frac{\pi x}{L}\Big), \sin\Big(\frac{2\pi x}{L}\Big), \sin\Big(\frac{3\pi x}{L}\Big), \ldots$$

form an eigenbasis of the spatial operator

$$Lu = \frac{\partial^2 u}{\partial x^2}$$

with Dirichlet boundary conditions. In this basis, the evolution is diagonal: each mode evolves independently according to a simple harmonic oscillator in time.

As in the heat equation example, a spectral method will approximate $u(x,t)$ by truncating the infinite sum. For some integer $N \geqslant 1$ we consider the finite approximation

$$u_N(x,t) = \sum_{n=1}^{N}(a_n\cos(\omega_n t) + b_n\sin(\omega_n t))\sin\Big(\frac{n\pi x}{L}\Big).$$

The analytic series above is the infinite dimensional limit of this spectral representation.

2.4 Numerical Illustration

To visualize the oscillatory behavior of the vibrating string, we compute the truncated Fourier sine series solution for a plucked string initial condition. The string is plucked at its center, forming a triangular initial displacement:

$$f(x) = \begin{cases} \frac{2h}{L}x & \text{for } 0 \leqslant x \leqslant L/2 \\ 2h(1 - x/L) & \text{for } L/2 \leqslant x \leqslant L \end{cases}$$

with zero initial velocity $g(x) = 0$. Here h denotes the height of the pluck at the center.

The Fourier sine coefficients of this triangular shape are

$$a_n = \frac{8h}{n^2\pi^2} \sin\left(\frac{n\pi}{2}\right),$$

which gives nonzero values only for odd n, with alternating signs.

The key portion of the implementation computes the solution at any point in space and time. In Python:

Python

```python
def wave_solution(x, t, a_n, b_n, L, c):
    u = np.zeros_like(x, dtype=float)
    n_modes = len(a_n) - 1
    for n in range(1, n_modes + 1):
        omega_n = c * n * np.pi / L
        spatial = np.sin(n * np.pi * x / L)
        temporal = a_n[n] * np.cos(omega_n * t) + b_n[n] * np.sin(omega_n * t)
        u += temporal * spatial
    return u
```

The equivalent MATLAB implementation:

Matlab

```matlab
u = zeros(size(x));
for n = 1:N_MODES
    omega_n = C * n * pi / L;
    spatial = sin(n * pi * x / L);
    temporal = a_n(n+1) * cos(omega_n * t) + b_n(n+1) * sin(omega_n * t);
    u = u + temporal * spatial;
end
```

Figure 3 shows the evolution of $u_N(x,t)$ with $N = 50$ modes at several time values within half a period $T = 2L/c$. The string oscillates back and forth, with the triangular shape inverting at $t = T/2$. Unlike the heat equation, the wave equation preserves energy and the solution does not decay — it continues oscillating indefinitely.

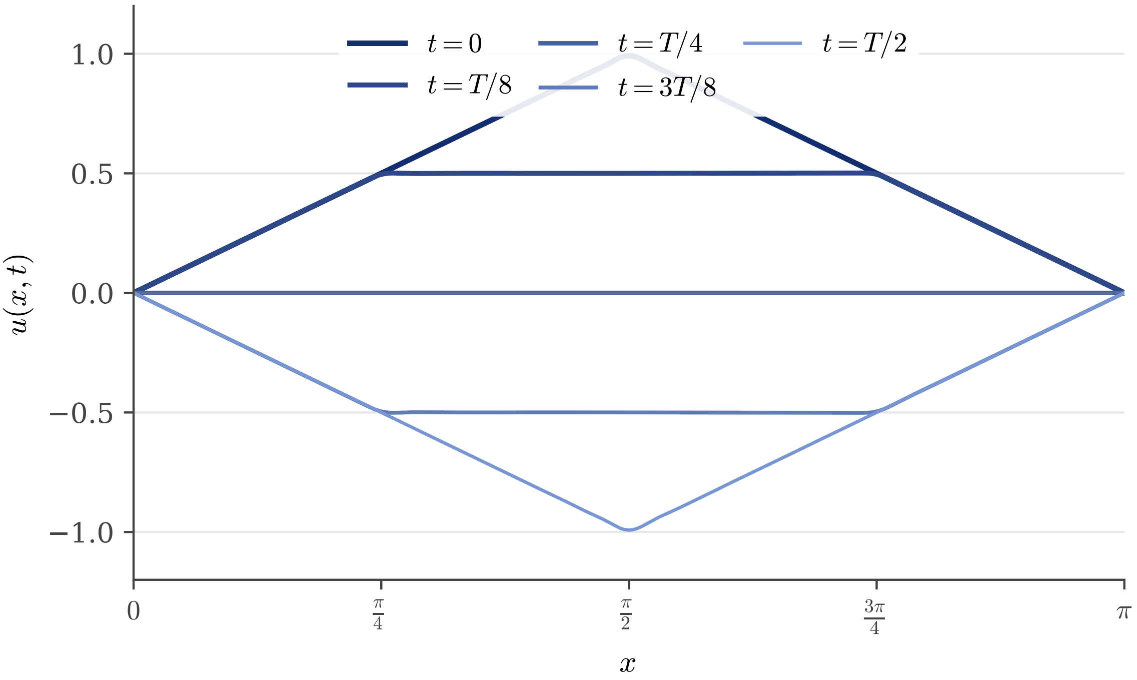

Figure 3: Evolution of the wave equation solution with a plucked string initial condition. The string oscillates with period $T = 2\pi$, inverting at $t = T/2$.

The code that generated this figure is available in both Python and MATLAB:

- `codes/python/ch02_classical_pdes/wave_equation_evolution.py`
- `codes/matlab/ch02_classical_pdes/wave_equation_evolution.m`

The waterfall plot in Figure 4 provides a complete view of the oscillatory dynamics over one full period. Unlike the heat equation, the wave equation conserves energy: the solution oscillates indefinitely without decay, and the periodic nature of the motion is clearly visible in the three-dimensional representation.

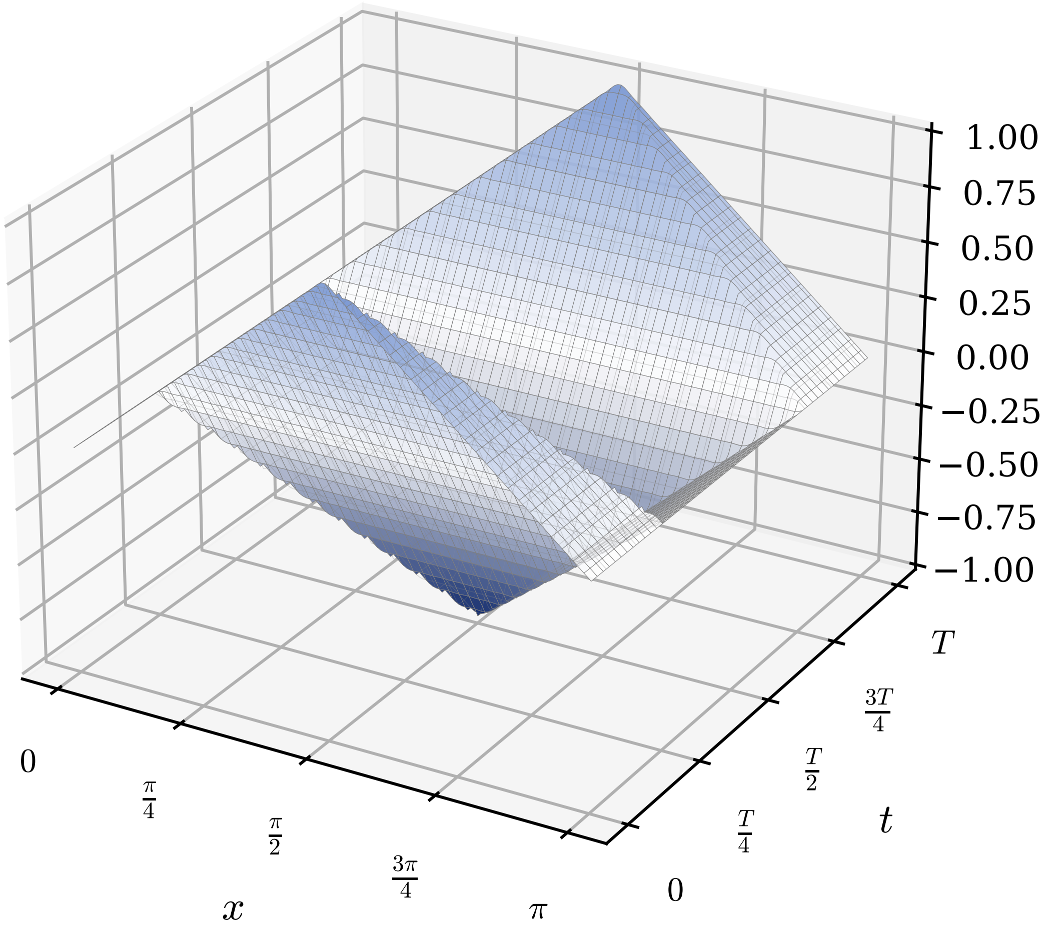

Figure 4: Waterfall plot showing the complete space-time evolution of the wave equation solution over one period T. The plucked string oscillates between its initial shape and its mirror image.

2.5 Laplace Equation in a Periodic Strip

For the elliptic case we consider the Laplace equation in a simple two dimensional domain that is periodic in one direction and bounded in the other. This setting connects naturally with the periodic heat equation example and again leads to a Fourier series representation in the periodic direction.

Let

$$D = \{(x, y) \in \mathbb{R}^2 : 0 < x < 2\pi,\ 0 < y < 1\}.$$

We seek a harmonic function $u(x, y)$ solving

$$u_{xx}(x, y) + u_{yy}(x, y) = 0, \quad (x, y) \in D,$$

with periodic boundary conditions in x

$$u(x + 2\pi, y) = u(x, y), \quad \text{for all real } \ x,\ 0 < y < 1,$$

and Dirichlet conditions in y

$$u(x, 0) = f(x), \quad u(x, 1) = 0, \quad 0 \leqslant x \leqslant 2\pi.$$

We assume that f is 2π periodic and smooth:

$$f(x+2\pi) = f(x).$$

As in the parabolic and hyperbolic examples, we apply separation of variables and obtain a representation of u as an infinite Fourier series in x with y dependent coefficients.

2.5.1 Step 1: Separation Ansatz

We look for nontrivial separated solutions of the form

$$u(x,y) = X(x) \cdot Y(y).$$

Substituting into the Laplace equation gives

$$X''(x) \cdot Y(y) + X(x) \cdot Y''(y) = 0.$$

Assuming X and Y are not identically zero, we divide by $X(x) \cdot Y(y)$:

$$\frac{X''(x)}{X(x)} + \frac{Y''(y)}{Y(y)} = 0.$$

The first term depends only on x, the second only on y. Therefore both must be equal to constants whose sum is zero. We introduce a separation constant λ and write

$$\frac{X''(x)}{X(x)} = -\lambda, \quad \frac{Y''(y)}{Y(y)} = \lambda.$$

This leads to the two ordinary differential equations

$$X''(x) + \lambda X(x) = 0,$$
$$Y''(y) - \lambda Y(y) = 0.$$

The periodic boundary conditions for u in the x direction imply the periodic conditions

$$X(x+2\pi) = X(x), \quad \text{for all real } \ x.$$

The Dirichlet conditions in y will be enforced later on the full series. For the separated functions Y we will impose appropriate conditions at $y = 1$, while the condition at $y = 0$ will be handled via the Fourier coefficients of f.

We have once more an eigenvalue problem in the periodic direction.

2.5.2 Step 2: Eigenfunctions in the Periodic Direction

The equation for X with periodic boundary conditions is exactly the same as in the heat equation example:

$$X''(x) + \lambda X(x) = 0, \quad X(x+2\pi) = X(x).$$

We recall the result: there is a constant mode with eigenvalue $\lambda_0 = 0$,

$$X_0(x) = 1,$$

and for each integer $n \geqslant 1$ there are cosine and sine modes with eigenvalues

$$\lambda_n = n^2,$$
$$X_n^{(c)}(x) = \cos(nx), \quad X_n^{(s)}(x) = \sin(nx).$$

The family

$$1, \cos(x), \sin(x), \cos(2x), \sin(2x), \ldots$$

forms an orthogonal basis in $L^2(0, 2\pi)$ for 2π periodic functions.

For each such eigenvalue we now solve the corresponding equation for Y.

2.5.3 Step 3: Equations for the y Dependent Factors

For each eigenvalue λ we have

$$Y''(y) - \lambda Y(y) = 0.$$

We treat separately the constant mode $\lambda_0 = 0$ and the nonzero modes $\lambda_n = n^2$ for $n \geqslant 1$.

Constant mode $\boldsymbol{\lambda_0 = 0}$

For $\lambda_0 = 0$ the equation reduces to

$$Y_0''(y) = 0.$$

The general solution is

$$Y_0(y) = A_0 + B_0 y.$$

We want separated solutions that satisfy the homogeneous boundary condition at $y = 1$:

$$u(x, 1) = 0 \quad \text{for all} \;\; x.$$

For the constant mode this means

$$X_0(x) \cdot Y_0(1) = Y_0(1) = 0,$$

hence

$$Y_0(1) = A_0 + B_0 = 0.$$

We choose a convenient normalization so that $Y_0(0) = 1$. Then $A_0 = 1$ and the relation $A_0 + B_0 = 0$ gives $B_0 = -1$. Thus

$$Y_0(y) = 1 - y.$$

This separated mode

$$u_0(x, y) = X_0(x) \cdot Y_0(y) = 1 - y$$

is harmonic, periodic in x, and vanishes at $y = 1$, with value 1 at $y = 0$.

Higher modes $\boldsymbol{\lambda_n = n^2}$ for $\boldsymbol{n \geqslant 1}$

For $\lambda_n = n^2$ the equation is

$$Y_n''(y) - n^2 Y_n(y) = 0.$$

The general solution can be written in hyperbolic form

$$Y_n(y) = \alpha_n \cosh(ny) + \beta_n \sinh(ny).$$

We require that each separated mode vanish at $y = 1$:

$$Y_n(1) = 0.$$

To match later the Fourier coefficients of f at $y = 0$, it is convenient to normalize so that

$$Y_n(0) = 1.$$

Imposing $Y_n(0) = 1$ gives

$$\alpha_n = 1.$$

Then

$$Y_n(1) = \cosh(n) + \beta_n \sinh(n) = 0$$

so

$$\beta_n = -\frac{\cosh(n)}{\sinh(n)} = -\coth(n).$$

Thus

$$Y_n(y) = \cosh(ny) - \coth(n) \cdot \sinh(ny).$$

An alternative and simpler expression uses the hyperbolic sine function of the distance to the boundary $y = 1$. One checks that

$$Y_n(y) = \frac{\sinh(n(1 - y))}{\sinh(n)}$$

satisfies

$$Y_n''(y) - n^2 Y_n(y) = 0,$$
$$Y_n(1) = 0,$$
$$Y_n(0) = 1.$$

Indeed, $Y_n(1) = \sinh(0)/\sinh(n) = 0$, $Y_n(0) = \sinh(n)/\sinh(n) = 1$, and

$$Y_n''(y) = n^2 \frac{\sinh(n(1-y))}{\sinh(n)} = n^2 Y_n(y).$$

We will use the form

$$Y_n(y) = \frac{\sinh(n(1-y))}{\sinh(n)}, \quad n \geqslant 1.$$

2.5.4 Step 4: Building the Series Solution

Each separated solution corresponding to the eigenfunctions in x and the functions Y_n in y has the form

$$\begin{aligned} u_0(x,y) &= C_0 \cdot Y_0(y) = C_0(1-y), \\ u_n^{(c)}(x,y) &= C_n \cdot \cos(nx) \cdot Y_n(y), \\ u_n^{(s)}(x,y) &= D_n \cdot \sin(nx) \cdot Y_n(y), \quad n \geqslant 1, \end{aligned}$$

for some constants C_0, C_n, D_n.

Since the Laplace equation is linear and the boundary condition at $y = 1$ is homogeneous, any linear combination of these separated solutions is again a solution that vanishes at $y = 1$. Therefore a general solution satisfying the periodic condition in x and the Dirichlet condition $u(x,1) = 0$ can be written as

$$u(x,y) = C_0(1-y) + \sum_{n=1}^{\infty} [C_n \cos(nx) + D_n \sin(nx)] \frac{\sinh(n(1-y))}{\sinh(n)}.$$

It remains to impose the boundary condition at $y = 0$,

$$u(x,0) = f(x).$$

At $y = 0$ we obtain

$$u(x,0) = C_0 + \sum_{n=1}^{\infty} [C_n \cos(nx) + D_n \sin(nx)],$$

because $Y_0(0) = 1$ and $Y_n(0) = 1$ for $n \geqslant 1$.

Hence the boundary condition $u(x,0) = f(x)$ becomes

$$f(x) = C_0 + \sum_{n=1}^{\infty} [C_n \cos(nx) + D_n \sin(nx)].$$

This is exactly the Fourier series expansion of f. For a 2π periodic f we have

$$f(x) = a_0 + \sum_{n=1}^{\infty} [a_n \cos(nx) + b_n \sin(nx)],$$

with coefficients

$$\begin{aligned} a_0 &= \frac{1}{2\pi} \int_0^{2\pi} f(x) \, dx, \\ a_n &= \frac{1}{\pi} \int_0^{2\pi} f(x) \cos(nx) \, dx, \quad n \geqslant 1, \\ b_n &= \frac{1}{\pi} \int_0^{2\pi} f(x) \sin(nx) \, dx, \quad n \geqslant 1. \end{aligned}$$

By uniqueness of the Fourier expansion, we must have

$$C_0 = a_0, \quad C_n = a_n, \quad D_n = b_n.$$

2.5.5 Step 5: Final Explicit Solution and Interpretation

Substituting these coefficients into the series we obtain the explicit representation

$$u(x,y) = a_0(1-y) + \sum_{n=1}^{\infty} [a_n \cos(nx) + b_n \sin(nx)] \frac{\sinh(n(1-y))}{\sinh(n)}, \quad 0 < y < 1.$$

Here a_0, a_n, and b_n are the Fourier coefficients of the boundary data f as defined above.

This series converges (under mild assumptions on f) to the unique harmonic function that is periodic in x, equal to f on $y = 0$, and zero on $y = 1$.

From a spectral viewpoint:

- The functions 1, $\cos(nx)$, $\sin(nx)$ are eigenfunctions of the one dimensional Laplacian $X \mapsto X''$ with periodic boundary conditions in x, with eigenvalues $\lambda_0 = 0$ and $\lambda_n = n^2$.
- For each spatial frequency n in the periodic direction, the dependence in the transverse direction y is determined by the simple ordinary differential equation $Y'' - \lambda_n Y = 0$ with boundary condition $Y(1) = 0$ and normalization $Y(0) = 1$. This gives the hyperbolic profiles

$$Y_0(y) = 1 - y,$$
$$Y_n(y) = \frac{\sinh(n(1-y))}{\sinh(n)}, \quad n \geqslant 1.$$

- The boundary data f at $y = 0$ is expanded in the eigenbasis $\{1, \cos(nx), \sin(nx)\}$ and each Fourier mode is propagated into the interior of the strip with its own y dependent factor $Y_n(y)$.

Analytically, the solution is an infinite sum of separated solutions. In a spectral method we will truncate this sum to finitely many modes in x,

$$u_N(x,y) = a_0(1-y) + \sum_{n=1}^{N} [a_n \cos(nx) + b_n \sin(nx)] \frac{\sinh(n(1-y))}{\sinh(n)},$$

and approximate the harmonic function inside the strip by this finite Fourier representation.

2.6 Numerical Illustration

To visualize the structure of harmonic functions in the strip, we compute the truncated Fourier series solution for a boundary condition that contains two modes:

$$f(x) = \sin(x) + \frac{1}{2}\sin(3x).$$

For this particular boundary data, the Fourier coefficients are simply $b_1 = 1$ and $b_3 = 1/2$, with all other coefficients zero. The solution can be written explicitly as

$$u(x,y) = \sin(x)\frac{\sinh(1-y)}{\sinh(1)} + \frac{1}{2}\sin(3x)\frac{\sinh(3(1-y))}{\sinh(3)}.$$

The key portion of the implementation evaluates the truncated series on a two-dimensional grid. In Python:

```python
def laplace_solution(x, y, a0, a_n, b_n):
    u = a0 * (1 - y)
    n_modes = len(a_n) - 1
    for n in range(1, n_modes + 1):
        if abs(a_n[n]) < 1e-15 and abs(b_n[n]) < 1e-15:
            continue
        y_factor = np.sinh(n * (1 - y)) / np.sinh(n)
        u += (a_n[n] * np.cos(n * x) + b_n[n] * np.sin(n * x)) * y_factor
    return u
```

The equivalent MATLAB implementation:

```matlab
U = a0 * (1 - Y);
for n = 1:N_MODES
    if abs(a_n(n+1)) < 1e-15 && abs(b_n(n+1)) < 1e-15
        continue;
    end
    y_factor = sinh(n * (1 - Y)) / sinh(n);
    U = U + (a_n(n+1) * cos(n*X) + b_n(n+1) * sin(n*X)) .* y_factor;
end
```

Figure 5 shows the solution $u(x, y)$ in the strip $[0, 2\pi] \times [0, 1]$. At the bottom boundary $y = 0$, the solution matches the prescribed boundary data $f(x)$. As y increases toward the top boundary, the solution decays to zero. Crucially, the higher frequency mode ($n = 3$) decays much faster than the lower frequency mode ($n = 1$), as the hyperbolic factor $\sinh(n(1 - y))/\sinh(n)$ decreases more rapidly for larger n. This illustrates the smoothing effect of harmonic extension into the interior.

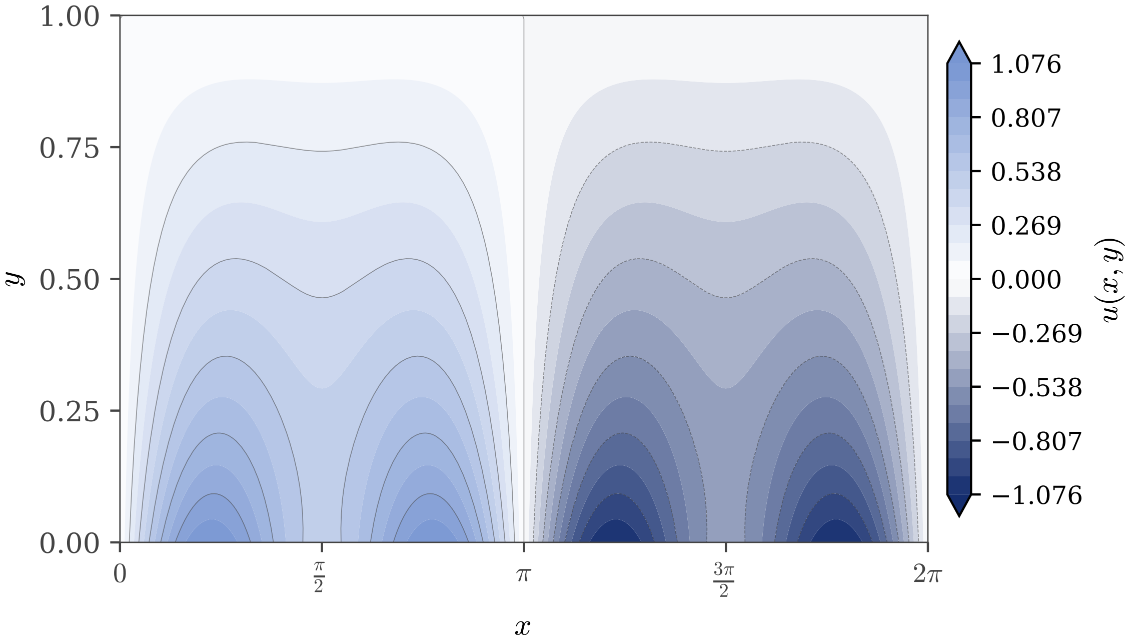

Figure 5: Solution of the Laplace equation in the periodic strip with boundary data $f(x) = \sin(x) + \frac{1}{2}\sin(3x)$ at $y = 0$ and $u = 0$ at $y = 1$. Higher frequency modes decay faster toward the interior.

The code that generated this figure is available in both Python and MATLAB:

- `codes/python/ch02_classical_pdes/laplace_equation_2d.py`
- `codes/matlab/ch02_classical_pdes/laplace_equation_2d.m`

2.7 Conclusions

The three examples presented in this chapter—the heat equation, the wave equation, and the Laplace equation—share a common mathematical structure that will guide us throughout the rest of this book. In each case, separation of variables reduces a partial differential equation to a family of ordinary differential equations: an eigenvalue problem in the spatial variable and a simpler equation governing the behavior in the remaining variable (time or the transverse coordinate). The eigenfunctions form an orthogonal basis, and the solution is expressed as an infinite series

$$u(x,t) = \sum_k \hat{u}_k(t)\varphi_k(x)$$

whose coefficients are determined by the initial or boundary data through Fourier projections. This is the DNA of spectral methods.

Let us distill the key ideas:

1. **Separation**. We decompose the solution into modes that evolve independently (or nearly so). Each mode satisfies a simpler equation than the original PDE.
2. **Basis**. The spatial part of each mode is an eigenfunction of a differential operator —Fourier exponentials for periodic problems, trigonometric functions for Dirichlet conditions, Chebyshev or Legendre polynomials for more general settings.
3. **Truncation**. In practice we cannot sum infinitely many terms due to the apparent finitude of our Universe. We retain only the first N modes, and the accuracy of this approximation depends critically on how fast the coefficients $\hat{u}_k$ decay.

When the solution is smooth, the coefficients decay *exponentially*, and a modest N suffices for high accuracy. This is the source of spectral methods' legendary efficiency.

The analytical solutions derived in this chapter are beautiful, but they are also fragile. They apply only to linear equations on simple domains with special boundary conditions. The moment we encounter a nonlinearity, a complicated geometry, or variable coefficients, we must turn to computation. The chapters that follow will develop the computational machinery needed to turn these analytical insights into practical algorithms:

- **FFT and its cousins**: how to move efficiently between physical space and coefficient space.
- **Differentiation matrices**: how to compute derivatives spectrally.
- **Quadrature rules**: how to compute inner products and projections.
- **Time stepping**: how to advance the ODE system for the coefficients.

Armed with these tools, we will be able to solve problems far beyond the reach of pen-and-paper analysis.