Project Report on the Advection-Diffusion Equation

A Study on **1D Numerical Schemes** for Transport-Diffusion

Prakhar (B23091) Nishita (B23403)

Indian Institute of Technology Mandi
May 2025

Contents

1 Introduction: Modeling Pollutant Transport

The advection-diffusion equation (ADE) is fundamental for understanding how substances like pollutants, heat, or chemicals move through fluids. This equation captures two key physical processes:
advection, which represents transport due to the bulk motion of the fluid, and
diffusion, which models the spreading of the scalar due to concentration gradients.

The general form of the ADE is:

$$\underbrace{\frac{\partial u}{\partial t}}_{\text{Temporal change}} + \underbrace{\mathbf{v} \cdot \nabla u}_{\text{Advection}} = \underbrace{D\nabla^2 u}_{\text{Diffusion}} + \underbrace{S(x,t)}_{\text{Source}} \tag{1}$$

where $u(x,t)$ represents concentration, $\mathbf{v}$ is the velocity field, D is the diffusion coefficient, and $S(x,t)$ accounts for sources or sinks.
In this study, we focus on applying Physics-Informed Neural Networks (PINNs) to solve both the forward and inverse forms of the ADE. The velocity field can be assumed known, for instance from the Navier-Stokes solution, and as the scalar is passive, it does not influence the flow. Thus, the ADE can be solved independently. PINNs provide a mesh-free framework that leverages both observational data and governing physical laws, allowing for accurate modeling even when data is sparse or noisy.

2 Motivation: Environmental Applications

The ADE is crucial for modeling how pollutants propagate in fluid environments, making it essential for applications in environmental science, climate modeling, and forensic analysis. Understanding pollutant transport has critical applications in:

Application	Challenge
Carbon sequestration	Detecting CO_2 leaks
Oil spill response	Predicting contamination spread
Air quality monitoring	Identifying pollution sources

Table 1: Real-world applications of ADE modeling

Our work focuses particularly on the **inverse problem** - determining source characteristics from limited observations, which is essential for environmental forensics and industrial monitoring.Such capabilities are aligned with the broader goal of Carbon Capture, Utilisation and Storage (CCUS) technologies, which are part of global initiatives to combat climate change. Thus, applying PINNs to source identification in ADE not only has academic value but also significant societal and environmental impact.

3 Mathematical Foundations

3.1 Derivation from Conservation Laws

A conservation law specifies how some quantity is conserved in some volume, and by which mechanisms the quantity can increase or decrease. In the case of a pollutant, we

want that it can only enter or exit the control volume via two mechanisms: sources and boundary fluxes. The ADE can be derived from the general conservation law:

$$\frac{d}{dt}\int_\Omega u\,dV = \underbrace{\int_\Omega F\,dV}_{\text{Sources}} - \underbrace{\oint_{\partial\Omega} \mathbf{J}\cdot\mathbf{n}\,dS}_{\text{Flux}} \tag{2}$$

where the total flux $\mathbf{J}$ combines advective and diffusive components:

$$\mathbf{J} = \underbrace{u\mathbf{v}}_{\text{Advective flux}} - \underbrace{D\nabla u}_{\text{Diffusive flux}} \tag{3}$$

In the case of the diffusion, we can use Fick's first law of diffusion. It is derived from the fact that pollutant will diffuse from regions with high concentration to regions of low concentration. Fick's law of diffusion postulates that the diffusion flux points in the direction from high concentration to low concentration, and that the rate is directly proportional to the gradient of said concentration,

$$\mathbf{J}_{\text{Diff}} = -D\nabla u \tag{4}$$

Here, D is the diffusion rate. It determines how rapidly a peak of concentration, where u is large, disperses and flattens towards an average.

Now let the total flux $\mathbf{J}$ be the sum of these two contributions $\mathbf{J} = \mathbf{J}_{\text{Adv}} + \mathbf{J}_{\text{Diff}}$. Then by equation (1) we arrive at the following conservation law:

$$\frac{d}{dt}\int_\Omega u\,dx = \int_\Omega F(x,t;\mathcal{P})\,dx - \int_{\partial\Omega} u\mathbf{v}\cdot\mathbf{n}\,dS + \int_{\partial\Omega} D\nabla u\cdot\mathbf{n}\,dS \tag{5}$$

To explain the choice of signs, note the following. If the velocity at the surface is pointing in the same direction as the outward unit normal vector then $\mathbf{v}\cdot\mathbf{n} > 0$. If we assume $D = 0$ and $F = 0$, then all pollutant inside must be moving out from the domain. Therefore, we would expect the concentration inside to be decreasing in this instance. That is why we have the negative sign for the second term on the right-hand side.

Similarly, note what happens when we only look at the contribution from diffusion, $\mathbf{v} = 0$, and $F = 0$. If there is a higher concentration outside the domain than inside, then we would expect there to be a net inflow to the domain. In this case, the gradient ∇u would be pointing in the same direction as $\mathbf{n}$, which does give a positive value corresponding to increasing concentration inside.

Now, we apply Gauss' theorem on the two surface integrals to transform them into volume integrals. We obtain:

$$\frac{d}{dt}\int_\Omega u\,dx = \int_\Omega F(x,t;\mathcal{P})\,dx - \int_\Omega \nabla\cdot(u\mathbf{v})\,dx + \int_\Omega \nabla\cdot(D\nabla u)\,dx \tag{6}$$

Since all terms are valid under the same domain, we can transition over to the differential form of the conservation law. This allows us to get rid of the integral terms:

$$\frac{\partial u}{\partial t} + \nabla\cdot(u\mathbf{v}) = \nabla\cdot(D\nabla u) + F(x,t;\mathcal{P}) \tag{7}$$

Equation (6) is the most general version of the ADE. It can be further expanded as:

$$\frac{\partial u}{\partial t} + \mathbf{v}\cdot\nabla u + u\nabla\cdot\mathbf{v} = D\nabla^2 u + \nabla D\cdot\nabla u + F(x,t;\mathcal{P}) \tag{8}$$

3.2 Simplified Form

For incompressible flow ($\nabla \cdot \mathbf{v} = 0$) and constant diffusivity, we obtain:

$$\frac{\partial u}{\partial t} + \mathbf{v} \cdot \nabla u = D\nabla^2 u + F(x,t) \tag{9}$$

By specifying a velocity field $\mathbf{v}$, a source term F, diffusivity rate D and with a sufficient set of initial and boundary conditions, one should be able to find a unique solution for u. The velocity field can be arbitrarily set by specifying it as a space-time dependent function or by using a velocity-field dataset when solving the PDE numerically in the discretized case.

There are two main types of boundary conditions applied to this problem: Dirichlet and Neumann boundary conditions. The former can be written as follows:

$$u(x,t)\Big|_{\partial\Omega} = f(x,t), \quad t > 0 \tag{10}$$

This forces u to satisfy some known function f at the boundary. If say $f = 0$, then this would represent walls which absorb pollutant such that the concentration is negligible next to the walls.

The latter condition, the Neumann boundary condition, can be written as:

$$\mathbf{n} \cdot \nabla u(x,t)\Big|_{\partial\Omega} = f(x,t), \quad t > 0 \tag{11}$$

Here $\mathbf{n}$ is the outward unit normal vector to $\partial\Omega$. In this case, it is the directional gradient of u that is forced to satisfy the known function f at the boundary. This, on the other hand, describes the fluxes of u at the boundary. If $f = 0$, then the walls would be reflective and pollutant would stay within the domain. If $f > 0$ at any point on the boundary, then there would be a net positive flux of pollutant into the domain there and vice versa for $f < 0$.

For the sake of simplicity, the homogeneous Dirichlet condition was chosen for our case studies. In other words, we chose the Dirichlet boundary condition for when $f = 0$. Though this is the less realistic scenario out of the two options, it does simplify the derivation of well-posedness for the PDE, the numerical computation for generating data in MATLAB, and it simplifies the treatment of the PINN. Though it should be noted that PINNs have been shown to be able to deal with Neumann boundary conditions as well. In addition, homogeneous initial condition was chosen, $u(x,0) = 0$.

In summary, with these choices, the problem describes how pollutant generated by a source F develops in time under the influence of a velocity field $\mathbf{v}$. It is known that the concentration is zero at the boundaries and that the leak starts at $t = 0$. Thus, the only possible way for pollutant to enter the domain would be from any sources within the domain, specified by F.

4 Source Term Modeling

4.1 From Ideal to Practical

While a delta function $\delta(\mathbf{x} - \mathbf{x}_s)$ would ideally represent a point source, its singularity makes it unsuitable for numerical methods. Instead, we use a Gaussian approximation:

$$F(\mathbf{x}) = \lambda e^{-S\|\mathbf{x}-\mathbf{x}_s\|^2} \tag{12}$$

The Gaussian source provides several advantages:

- Differentiable everywhere (essential for PINNs)
- Parameter S controls localization
- Approaches delta function as $S \to \infty$
- Physically realistic (real sources have finite extent)

4.2 Illustrative Example

Figure 1 shows an example of the interplay between the ADE solution u and the velocity field $\mathbf{v}$. It shows how the pollutant generated at a source F at the point $\mathbf{x}_s = [14\ 5\ 0]^T$ varies in intensity and how it is advected and diffused downstream.

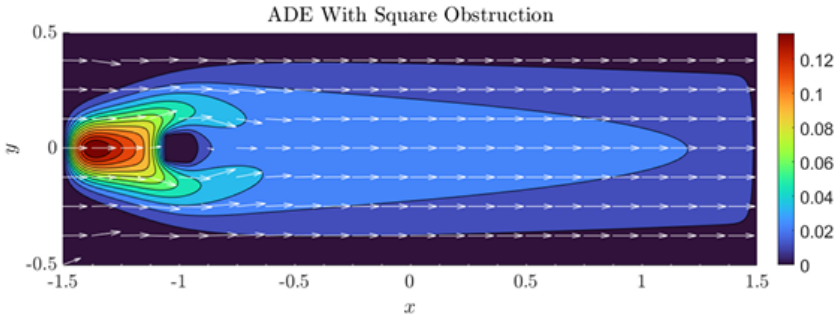

Figure 1: Illustration of pollutant evolution and velocity field in the ADE simulation.

Solution of the Advection-Diffusion Equation

To solve the advection-diffusion equation with decay, we start from the PDE:

$$\frac{\partial C}{\partial t} + u\frac{\partial C}{\partial x} = D\frac{\partial^2 C}{\partial x^2} - kC \tag{13}$$

where:

- C: concentration,
- D: diffusion constant,
- u: advection velocity,
- k: decay rate.

Initial condition: An instantaneous point source release of mass M at $x = 0$, $t = 0$:

$$C(x, 0) = M\delta(x)$$

Boundary condition:

$$\lim_{x \to \pm\infty} C(x, t) = 0$$

Fourier Transform

We apply the Fourier transform with respect to x, defined as:

$$\hat{C}(k,t) = \int_{-\infty}^{\infty} C(x,t)e^{-ikx}dx$$

This technique is useful because it converts partial differential equations into ordinary differential equations. Specifically:

- Differentiation in space becomes multiplication in frequency:
  $\mathcal{F}\left[\frac{\partial C}{\partial x}\right] = ik\hat{C}(k,t)$, and
  $\mathcal{F}\left[\frac{\partial^2 C}{\partial x^2}\right] = -k^2\hat{C}(k,t)$.
- The delta function initial condition becomes a constant in Fourier space.

Applying the Fourier transform to both sides of the PDE:

$$\mathcal{F}\left[\frac{\partial C}{\partial t} + u\frac{\partial C}{\partial x}\right] = \mathcal{F}\left[D\frac{\partial^2 C}{\partial x^2} - kC\right]$$

yields:

$$\frac{\partial \hat{C}}{\partial t} = -iuk\hat{C} - Dk^2\hat{C} - k\hat{C}$$

This simplifies to:

$$\frac{\partial \hat{C}}{\partial t} = -(Dk^2 + iuk + k)\hat{C}$$

Solving the ODE in Frequency Space

This is a first-order linear ODE in t. The general solution is:

$$\hat{C}(k,t) = \hat{C}(k,0)\exp\left[-(Dk^2 + iuk + k)t\right]$$

From the initial condition $C(x,0) = M\delta(x)$, we use the Fourier identity $\mathcal{F}[\delta(x)] = 1$ to get:

$$\hat{C}(k,0) = M$$

So we obtain:

$$\hat{C}(k,t) = M\exp\left[-(Dk^2 + iuk + k)t\right]$$

Inverse Fourier Transform

To retrieve $C(x,t)$ from $\hat{C}(k,t)$, we take the inverse Fourier transform:

$$C(x,t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{C}(k,t)e^{ikx}dk$$

Substitute $\hat{C}(k,t)$:

$$C(x,t) = \frac{M}{2\pi} \int_{-\infty}^{\infty} \exp\left[-Dk^2t - iukt - kt + ikx\right] dk$$

Combine exponential terms:

$$C(x,t) = \frac{M}{2\pi} \int_{-\infty}^{\infty} \exp\left[-Dk^2t + ik(x-ut) - kt\right] dk$$

The integral is a standard Gaussian form. Completing the square in the exponent and using the identity:

$$\int_{-\infty}^{\infty} e^{-ak^2+bk} dk = \sqrt{\frac{\pi}{a}} e^{\frac{b^2}{4a}}$$

gives the final solution:

$$C(x,t) = \frac{M}{\sqrt{4\pi Dt}} \exp\left[-\frac{(x-ut)^2}{4Dt}\right] \exp(-kt)$$

Interpretation

- The Gaussian term $\exp\left[-\frac{(x-ut)^2}{4Dt}\right]$ describes spreading due to diffusion, centered at $x = ut$.
- The ut term indicates the peak moves with velocity u due to advection.
- The decay term $\exp(-kt)$ reduces the overall concentration exponentially with time.

Physics-Informed Neural Network (PINN) Implementation for Advection-Diffusion Equation

To solve the advection-diffusion equation using a Physics-Informed Neural Network (PINN), we utilize the neural network's ability to approximate the solution to partial differential equations (PDEs) while ensuring that the solution adheres to the physical constraints defined by the PDE, the initial condition, and the boundary condition.

Formulation of the Problem

The advection-diffusion equation with decay is given by:

$$\frac{\partial C}{\partial t} + u\frac{\partial C}{\partial x} = D\frac{\partial^2 C}{\partial x^2} - kC$$

where:

- $C(x,t)$: concentration of the substance at position x and time t,
- D: diffusion constant,
- u: advection velocity,
- k: decay rate.

Initial condition:

$$C(x, 0) = M\delta(x)$$

Boundary condition:

$$\lim_{x \to \pm\infty} C(x, t) = 0$$

Physics-Informed Neural Network Architecture

The neural network is trained to approximate the solution $C(x, t)$ using the following steps:

- **Neural Network Structure:** The neural network consists of an input layer with two nodes representing the spatial coordinate x and time t, followed by several hidden layers, and an output layer representing the concentration $C(x, t)$.
- **Loss Function:** The loss function is constructed from the PDE residual, the initial condition, and the boundary condition. The network's output is penalized if it deviates from the physical constraints.
- **PDE Residual:** The residual is computed by substituting the neural network's output into the left-hand side of the PDE:

  $$R(x, t) = \frac{\partial C}{\partial t} + u\frac{\partial C}{\partial x} - D\frac{\partial^2 C}{\partial x^2} + kC$$

  The network is trained to minimize the residual across the domain.
- **Boundary Conditions:** To enforce the boundary conditions, the network is trained on points near the boundaries, ensuring that $C(x, t) \to 0$ as $x \to \pm\infty$.
- **Initial Condition:** The network is trained to satisfy the initial condition at $t = 0$, ensuring that $C(x, 0) = M\delta(x)$.

Training the PINN

The training process involves minimizing the following total loss function:

$$\mathcal{L}_{total} = \mathcal{L}_{PDE} + \mathcal{L}_{BC} + \mathcal{L}_{IC}$$

where:

- $\mathcal{L}_{PDE}$: Loss associated with the PDE residual,
- $\mathcal{L}_{BC}$: Loss associated with the boundary conditions,
- $\mathcal{L}_{IC}$: Loss associated with the initial condition.

Implementation Steps

Advantages of PINNs

- PINNs do not require discretization of the spatial and temporal domains, making them well-suited for high-dimensional problems.
- The network can generalize to different boundary and initial conditions without the need for re-meshing or re-solving.
- PINNs provide an efficient approach for solving PDEs when analytical solutions are difficult or unavailable.

Neural Network Architecture and Training Process

Observations

In this section, we present the results from our Physics-Informed Neural Network (PINN) implementation. The following figures and GIFs showcase the predicted source paths, the loss history during training, and the comparison between pure advection and the full advection-diffusion simulation.

Predicted Source Paths

The following figure shows the predicted source paths calculated using PINN for different initial guesses. These paths represent the concentration of the substance over time for various initial conditions.

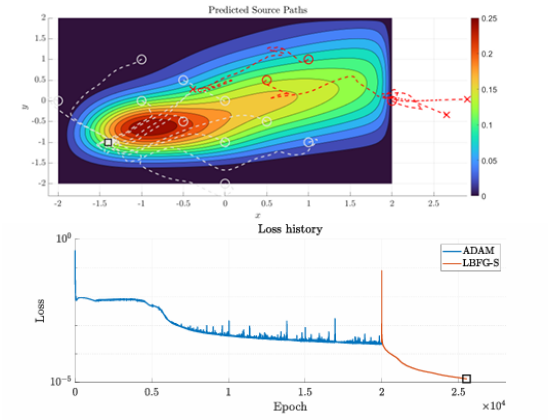

Figure 2: (Above) Predicted source paths for different initial guesses. (Below) Loss history during the training of the PINN.

GIFs: Pure Advection vs Main Simulation

Below are GIFs comparing the behavior of the system under pure advection and the full advection-diffusion simulation (including diffusion and decay effects).

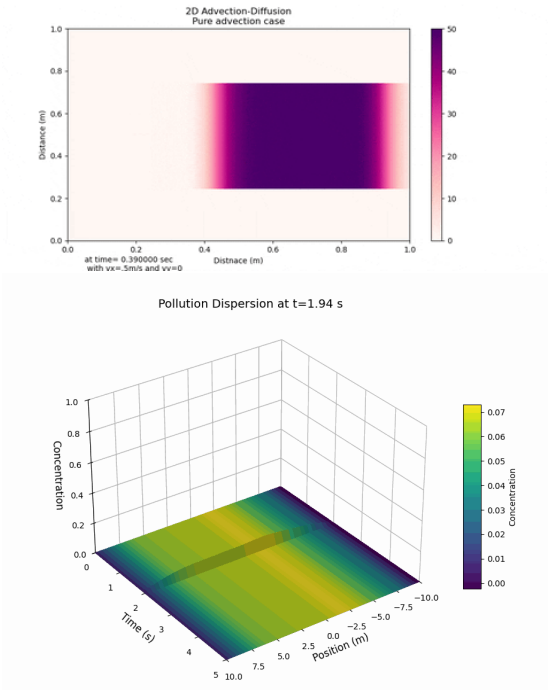

Figure 3: (Above) GIF showing pure advection simulation. (Below) GIF showing the main simulation with advection, diffusion, and decay.

References

[1] Cantor, B. (2020). *The Equations of Materials*. Oxford University Press.

[2] Ewing, R., and H. Wang. (2001). A summary of numerical methods for time-dependent advection-dominated partial differential equations. *Journal of Computational and Applied Mathematics*, pp. 423–445.

[3] Gustafsson, B., Kreiss, H.-O., Oliger, J. (2013). *Time Dependent Problems and Difference Methods*. Wiley.