East Afr. J. Biophys. Comput. Sci. (2023), Vol. 4, Issue. 1, 52-74

East African Journal of Biophysical and Computational Sciences

Journal homepage : https://journals.hu.edu.et/hu-journals/index.php/eajbcs

# Numerical Solutions of Advection Diffusion Equations Using Finite Element Method

Kassahun Getnet Mekonen*, Zerihun Kinfe Birhanu

*Department of Mathematics, College of Natural and Computational Sciences, Hawassa University, Hawassa, Ethiopia.*

**KEYWORDS**:

Finite Element Method; Varational Formulation; Numerical integration

**ABSTRACT**

In this paper, we have implemented the finite element method for the numerical solution of a boundary and initial value problems, mainly on solving the one and two dimensional advection-diffusion equation with constant parameters. In doing so, the basic idea is to first rewrite the problem as a variational equation, and then seek a solution approximation from the space of continuous piece-wise linear's. This discretization procedure results in a linear system that can be solved by using a numerical algorithm for systems of these equations. The techniques are based on the finite element approximations using Galerkin's method in space resulting system of the first order ODE's and then solving this first order ODE's using backward Euler descritization in time. For the two dimensional problems, we use the ODE solver ODE15I to descritize time. The validity of the numerical model is verified using different test examples. The computed results showed that the use of the current method is very applicable for the solution of the advection-diffusion equation.

## INTRODUCTION

An advection-diffusion equation (ADE) is a mathematical model that has been used to model the concentration of pollutants. It gives the amount of pollutant concentration fields a fter i nput o f t he velocity data from the hydrodynamic model which are derived from mass balances. Formally the ADE equation is given by

$$u_t + a\nabla u = \nabla(D\nabla u) + f, \qquad (1)$$

where $u$ is the concentration of the pollutant, $a$ is the velocity of the considered particle, $D$ the diffusion coefficient and $f$ defines the sources and sinks due to different processes.

For the vast majority of geometries and problems, Eq. 1 cannot be solved with analytical methods, and an approximation of the equations can be constructed with different types of discretizations. Many numerical schemes have been implemented to approximately solve the ADE (Lima et al., 2021; Mahmud, 2012; Pochai and Deepana, 2011; Lian et al., 2016; Szymkiewicz and Gka-

* *Corresponding author:*
*Email: kassahunm@hu.edu.et +251918491415* *https://dx.doi.org/10.4314/eajbcs.v4i1.5S*

siorowski, 2021). In numerical method, a discrete approximations for the solution is computed by descritizing the given domain in to different sets of sub domains. In this paper we were focus on finite element method to solve the PDE given in Eq. 1. We will implement the method for one and two dimensional PDE's. In this method, first we develop a weak formulation, from which we derive the discretization by multiplying both sides of the ADE equation by a test function and integration by parts (Green-Gauss Theorem) to reduce second order derivatives to first order terms, i.e., weak formulation. Then we represent the approximate solution by the linear combination of basis functions, by constructing a set of basis functions based on the mesh of our domain. That is, the solution $u$ can be approximated by a function $u_h$ using the linear combinations of the basis functions $\phi_i$ according to the following expressions:

$$u \approx u_h = \sum u_i \phi_i. \tag{2}$$

Here, $\phi_i$ denotes the basis functions and $u_i$ denotes the coefficients of the functions that approximate $u$ with $u_h$. After this we get a system of linear equations and we solve the linear system of equations to obtain the approximate solution.

The development of finite element method has favored by the progress of computer technology and numerical calculus, and originally applied for mechanical structures (Lima et al., 2021; Mahmud, 2012; Donea and Huerta, 2003) . Several procedures have been tried to interpret separately the advection and diffusion pollutant transport. The FEM can help to face more complex problems and the privilege importance of the method is that it can be adapted to complex geometry domains, but the element wise intervals can assume any form of size, and obviously there is an expense of more burdensome calculations. We had to do the descritization process of the finite element method for the $1D$ and $2D$ Poisson equations in which it is an auxiliary step in solving the advection diffusion equation with the FEM. Usually the numerical solutions of PDE' including the equation (Eq. 1) are done with the finite difference method.

## THE MATHEMATICAL MODEL FORMULATION

To derive the advection diffusion equations for the application of pollution models, consider an elementary water body. Water quality within this body depends on the polluting substance mass present there. The water quality models describe the change in the mass of a polluting substance within the water body. The change is calculated as the difference between mass-flows (mass fluxes) entering and leaving this water body, considering also the effects of internal sources and sinks of the substance, if any. The mechanism of mass transfer into and out of this water body includes the following processes:

- Mass is transported by the flow, $a$, of the velocity vector. This process is termed as the advective mass transfer. The transfer of mass, that is the mass flux can be calculated as $u \times a$, where $u$ is the concentration of the substance in the water.
- The dispersive mass transfer is usually expressed by the law of Fix which states that the transport of

the substance in the direction of a space is proportional to the concentration of this substance in that direction and the proportionality factor being the coefficient of dispersion, $D\nabla u$.

By considering a volume element of porous mediums in three dimensional cartesian coordinates the equations are derived (Bajellan, 2015). Since we are considering advection and diffusion as the two modes of transport of a fluid within the porous medium, we can represent these two transport modes in the $x$-direction mathematically as:

$$\text{transport by advection} = audA \ ,$$

$$\text{transport by diffusion} = D_x \frac{\partial u}{\partial x} \ ,$$

where $dA$ is an elemental cross-sectional area of the cubic element, and $D_x$ is the diffusion coefficient in the $x$-direction.

Assuming that the two components (advection and diffusion) may be superposed, the total amount of material transported parallel to any given direction is obtained by summing the advective and diffusive transports. Using the mass balance approach by equating the difference between the mass of material entering a volume element and that leaving the element (i.e., net influx of mass) to the rate of accumulation of mass inside the volume.

$$\frac{\partial}{\partial t}u(x,t) + \frac{\partial}{\partial x}\left(a(x,t)u(x,t)\right) = \frac{\partial}{\partial x}\left(d(x,t)\frac{\partial}{\partial x}u(x,t)\right) + f(t,x). \tag{3}$$

The multidimensional advection-diffusion equation is used for analyzing mixing problems in rivers. One of the practical difficulties is that the equation requires some prior information about water depths, velocities, and diffusion coefficients, which could not conveniently be gathered in field experiments. In some particular mixing problems, however, some of the terms in the multi-dimensional advection-diffusion equation are negligibly small, so that the problem can be simplified by reducing the model to one dimension (Lima et al., 2021; Hundsdorfer, 1996) and the one dimensional advection diffusion equation is given in Eq. 3.

The time-derivative term expresses accumulation of mass at a point in space, the advection term $a\nabla u$ transport of mass with the flow, and the diffusion term $D\nabla^2 u$ reflects transport of mass due to molecular diffusion (Langtangen, 1999). We shall consider the equation in the space interval $\Omega \subset \mathbb{R}$ with time $t \geq 0$. An initial condition $u(x, 0)$ will be given and we also assume that suitable boundary conditions are provided, and for our work we consider the velocity field and the diffusion term as constants.

Both advection and diffusion move a pollutant material from one place to another, but each accomplishes this differently. The essential difference of the advection and diffusion is that advection moves the pollutants in one way (downstream) but diffusion goes in both ways (regardless of a stream direction). This is seen in the respective mathematical expressions of the advection equation $a\frac{\partial u}{\partial x}$ which has a first-order derivative, and the diffusion equation $D\frac{\partial^2 u}{\partial x^2}$ that has a second-order derivative.

Questions are arise like, can we have cases of fast advection and relatively weak diffusion and other cases of negligible advection and fast diffusion? To answer this question, we must compare the sizes of the $a\frac{\partial u}{\partial x}$ and $D\frac{\partial^2 u}{\partial x^2}$ terms to each other, and this is accomplishes by introducing scales.

| Variable | scale | choice of value |
|---|---|---|
| $u$ | $U$ | The concentration value such as initial, boundary, or average value |
| $a$ | $V$ | The maximum velocity value |
| $x$ | $X$ | Approximate length of the domain or size of release location |

Using these scales, we can derive estimates of the sizes of the different terms. Since the derivative $\frac{\partial u}{\partial x}$ is expressing the difference in concentration over a distance of infinitesimal limit, we can estimate it to be approximately $\frac{U}{X}$, and the advection term scales as:

$$a\frac{\partial u}{\partial x} \sim V\frac{U}{X}.$$

Similarly, the second derivative $\frac{\partial^2 u}{\partial x^2}$ represents the difference of a gradient over a specified distance and is estimated at $\frac{(\frac{U}{X})}{X} = \frac{U}{X^2}$, and the diffusion term scales as:

$$D\frac{\partial^2 u}{\partial x^2} \sim D\frac{U}{X^2}.$$

Equipped with these estimates, we can then compare the two processes by forming the ratio of their scales:

$$\frac{\text{Advecton}}{\text{Diffusion}} = \frac{V\frac{U}{X}}{D\frac{U}{X^2}} = \frac{VX}{D}$$

This ratio is dimensionless and Traditionally, it is called the Peclet number and is denoted by $Pe$:

$$Pe = \frac{VX}{D}.$$

If $Pe \ll 1$ (if $Pe < 0.1$): the advection term will result significantly smaller than the diffusion term. Physically, diffusion dominates and advection is negligible. So, spreading occurs symmetrically despite of the flow of the directional bias. If we wish to simplify the problem, we may drop the $a\frac{\partial u}{\partial x}$ term, as if $a$ were nil (no amount at all). The relative error occurred in the solution is expected to be on the order of the Peclet number, and the smaller $Pe$ leads to the smaller error. The solutions established with diffusion only were based on such simplification and are thus valid as long as $Pe \ll 1$.

If $Pe \gg 1$ (if $Pe > 10$): the advection term is significantly bigger than the diffusion term. Physically, the diffusion term is negligible and advection dominates, and spreading is existent, with the patch (small area) of pollutant being simply moved along by the flow. If we wish to simplify the problem, we may drop the $D\frac{\partial^2 u}{\partial x^2}$ term, as if $D$ were zero. The relative error occurred in doing the solution is expected as the order of the Peclet number inverse ($1/Pe$), and the larger $Pe$ will result the smaller error.

If $Pe \sim 1$ (in practice, if $0.1 < Pe < 10$): the advection and diffusion terms are not significantly different which results for the non dominance of the two in the process. The full equation must be utilized as there will no approximation to the equation will be justified.

# NUMERICAL METHOD

In this paper we use the finite element method (FEM) to approximate the solution of the advection diffusion equation. The method is examined as an emerging tool for the approximate solution of differential equations describing different physical processes (Yang et al., 2020). It is based on the basic finite e lement procedures, those are: the variation form formulation of the problem, the discretization of the formulation in a finite element, and the solution of the resulting finite element equations. FEM cuts a given domain into several elements (pieces of the domain) and connected in a finite number of nodal points.

The FEM is based on the integration of the terms in the equation to be solved, in form of point discretization schemes. It utilizes the method of weighted residuals and integration by parts (Green-Gauss Theorem) to reduce second order derivatives to first order terms. The solution domain is discretized into individual elements and these elements are operated upon individually and then solved globally using matrix solution techniques. Such a task could be done automatically by a computer, but it necessitates an amount of mathematical skill that to day still requires human involvement, (Brenner et al., 2008).

The theories of finite element methods provided the reasons why it worked well for the class boundary/initial value problems (Lima et al., 2021; Ahsan, 2012; Larson and Bengzon, 2010). Extension of the mathematical basis to nonlinear and non-structural problems was achieved through the method of weighted residuals (MWR), originally conceived by Galerkin in the early 20th century. The basics of the method requires multiplying of the governing differential equation by a set of predetermined weights and integrating the resulting product over a region. Most of the finite element method uses the Galerkin's method to establish the approximations of the governing equations, (Aragonés et al., 2019; Lima et al., 2021; Ahsan, 2012; Yang et al., 2020; Brenner et al., 2008). It allows us to convert a continuous form of the problem, such as the weak formulation for the partial differential equation into a discrete problem that may be solved numerically.

## Finite Element Implementation of the $1D$ Governing Equation

Let us consider the one dimensional Advection-Diffusion equation given by:

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = D\frac{\partial^2 u}{\partial x^2} + f,\ u(0) = u(1) = 0, \tag{4}$$

where $u$ is the concentration of the pollutant, $a$ is the velocity, $f$ is the source term, and $D$ is the diffusion coefficient, with all the three variables be constants.

### Implementation of the $1D$ Advection Equation

Let we first consider only an advection equation, that is the diffusion term does not exist;

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0.$$

The first step is constructing a variational or weak formulation, by multiplying both sides of the differential equation by a test function $v(x)$ satisfying the boundary conditions (BC) $v(0) = 0, v(1) = 0$ and $v \in H_0^1(0,1)$, where $H_0^1(0,1)$ is the Sobolev space,

$$H_0^1(0,1) = \{v \in L^2(0,1); v' \in L^2(0,1)\},$$

and it is a function space where all the functions are bounded. Let us now define a sub-space of $H$ where we can find our solution $u$. We call this $V$ and

$$V = \{v \in H(X) : v|_{\partial\Omega} = 0\},\ \text{where } \Omega \text{ is our domain,}$$

Then multiplying and integrating both sides in the domain we have that:

$$\frac{\partial u}{\partial t}.v + a\frac{\partial u}{\partial x}.v = 0,$$

$$\int_0^1 \left(\frac{\partial u}{\partial t}.v + a\frac{\partial u}{\partial x}.v\right) = 0.$$

That is;

$$\int_0^1 \frac{\partial u}{\partial t}.v + \int_0^1 a\frac{\partial u}{\partial x}.v = 0, \qquad (5)$$

which is the weak formulation of the one dimensional advection equation.

**Advantages of weak form compared to strong form**

Equation 5 is the final weak formulation. It is equivalent to the strong form, since we can reverse all the steps, and get back to the original equation. Firstly, if we look at the strong form, we have two separate partial derivatives of $u$, so the strong form requires that $u$ be continuously differentiable until at least second partial derivative. Our new formulations has lowered this requirement to only first partial derivatives by transforming one of the partial derivatives onto the weight-function $v(x, y)$. This is the first big advantage of a weak formulation. The subspace $V$ is not difficult to understand; it is a subspace of $H$ because our weak form requires that the functions are in $H$; our strong form requires that $u$ be 0 along the boundary, so $V$ is the subspace of all function which are zero on the boundary.

The next step is to generate a mesh, let be a uniform Cartesian mesh $x_i = ih, i = 0, 1, ..., n,$ where $h = \frac{1}{n}$, and we define the intervals as $[x_{i-1}, x_i], i = 1, 2, ..., n.$
After generating a mesh we construct a set of basis functions based on the mesh for each intervals, such as the piece wise linear functions for $i = 1, 2, ..., n-1$. The characteristic basis functions are characterized by the following property, (Quarteroni and Quarteroni, 2009)

$$\phi_i(x_j) = \delta_{ij},\ i, j = 1, ..., n-1, \qquad (6)$$

where $\delta_{ij}$ being the Kronecker delta. The function $\phi_i$ is therefore piece wise linear and are fix with one node (vertex) and associate the value one to this node and zero at the remaining nodes of the partition (see Fig. 1, (Larson and Bengzon, 2010)).

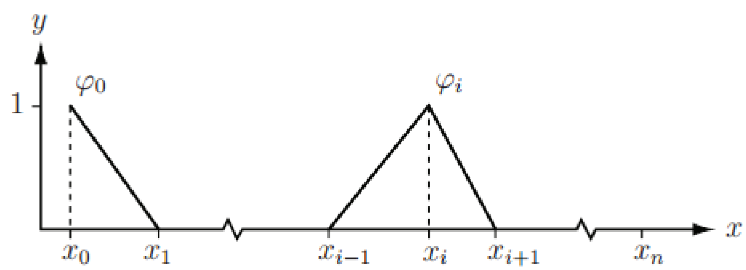



Figure 1: The basis (hat) function $\phi_i$ associated to node $x_j$, in this figure $\varphi_i$ on a mesh. Also shown is the half hat $\phi_0$.

Its expression is given by

$$\phi_i(x) = \begin{cases} \frac{x - x_{i-1}}{x_i - x_{i-1}}, & \text{if } x_{i-1} \le x \le x_i \\ \frac{x_{i+1} - x}{x_{i+1} - x_i}, & \text{if } x_i \le x \le x_{i+1} \\ 0, & \text{other wise} \end{cases}, \text{ for } i = 1, 2, \ldots, n-1. \qquad (7)$$

That is,

$$\phi_i(x_j) = \delta_{ij} = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{if } i \neq j \end{cases}. \qquad (8)$$

We use this hat (basis) functions through out the $1D$ space of this research using equal spaced step size ($x_{i+1}-x_i = h$, for all $i$). We say that the functions are basis for the following reasons. If we want to approximate our continuous function $u$ with a piece wise continuous linear function $u'$, these functions are what we need. These functions are linearly independent of each other; it is not possible to make one out of a combination of others. For example, only one of these functions, $\phi_i$, is non-zero (equal to 1) at node $i$. The next step in approximating a PDE with FEM is represent the approximate (FE) solution by the linear combination of such basis functions, (Quarteroni and Quarteroni, 2009) as

$$u_h(x) = \sum_{j=1}^{n-1} c_j \phi_i(x), \tag{9}$$

where the coefficients $c_j$ are the unknowns to be determined. since the hat (basis) functions are piece wise linear, $u_h(x)$ is also a piece wise linear function, although this is not usually the case for the true solution u(x), and here we have,

$$u_h(x_j) = \sum_{i=1}^{n-1} c_j \phi_i(x_j) = c_j.$$

We then derive a linear system of equations for the coefficients by substituting the approximate solution $u_h(x)$ for the exact solution $u(x)$ in the weak form:

Then let the approximate solution for $u$ be given by a linear combination of basis functions $\phi_i = \delta_{ij}$, as given in Eq. 10, and also for $v$ as in Eq. 11. Now we find a finite element solution of the discrete problem by using the hat functions $\phi_i(x)$ defined in Eq. 7. For the given basis function the approximation of $u$ and $v$ can be written as :

$$u(t,x) = \sum_{i=1}^{N-1} u_i \phi_i(x), \tag{10}$$

$$v(t,x) = \sum_{j=1}^{N-1} v_j \phi_j(x). \tag{11}$$

Now substituting Eq.10 and Eq.11 in the weak formulation of the equation Eq.5, we have:

$$\int_0^1 \frac{\partial}{\partial t} \sum_{i=1}^{N-1} u_i \phi_i . \sum_{j=1}^{N-1} v_j \phi_j + \int_0^1 a \frac{\partial}{\partial x} \left( \sum_{i=1}^{N-1} u_i \phi_i \right) . \sum_{j=1}^{N-1} v_j \phi_j = 0.$$

which then implies,

$$\sum_{j=1}^{N-1} v_j \left( \frac{\partial}{\partial t} \sum_{i=1}^{N-1} u_i \int_0^1 \phi_i . \phi_j + a \sum_{i=1}^{N-1} u_i \int_0^1 \phi_i' . \phi_j \right) = 0.$$

That is

$$\frac{\partial}{\partial t} \sum_{i=1}^{N-1} u_i \int_0^1 \phi_i . \phi_j + a \sum_{i=1}^{N-1} u_i \int_0^1 \phi_i' . \phi_j = 0.$$

In a matrix form it can be written as:

$$M\dot{U} + aBU = 0, \tag{12}$$

where,

- $M$ is the mass matrix with entries:

$$M_{i,j} = \int_0^1 \phi_i(x)\phi_j(x)dx.$$

- B is a matrix with entries:

$$B_{ij} = \int_0^1 \phi_i'(x)\phi_j(x)dx = \begin{cases} 0, & \text{if } i = j \\ \frac{-1}{2}, & \text{if } i - j = 1 \\ \frac{1}{2}, & \text{if } j - i = 1 \\ 0, & \text{other wise} \end{cases} . \tag{13}$$

Here $M$ and $A$ are tridiagonal matrices and Eq.12 is a simple system of ODE.

The next step is to descritize the sys- tem Eq.12 in time. Here we were consider finite difference approximations specially the implicit Euler (Back ward Euler) method. By using the back ward Euler scheme, the system Eq.12 results the following system of algebraic equations:

$$M\left(\frac{U^{n+1} - U^n}{\Delta t}\right) + aBU^{n+1} = 0.$$

Here $U^n$ denotes $U$ at time $t = t^n = \Delta t n$, and $\Delta t$ is the time step. Rearranging the terms we obtain the system:

$$\left(\frac{M}{\Delta t} + aB\right)U^{n+1} = \frac{1}{\Delta t}MU^n,\ n = 0, 1, 2, \ldots,$$

to be solved for $U^{n+1}$ by using initial condition for $U^0 = U(x, t = 0)$.

Euler Back ward represents an implicit scheme which is stable for all choices of $\Delta t$ (?). Since the scheme is implicit, we have to solve a system of algebraic equations at each time step.

## Assembly of the mass matrix $M$ in $1D$

Let us now go through the details of how to assemble the mass matrix $M$. We begin by calculating the entries $M_{i,j}$ of the mass matrix, which involve products of hat functions given in Eq. 7. Since each hat is a linear polynomial, the product of two hats is a quadratic polynomials. Thus, Simpson's formula (Eq. 18) can be used to integrate $M_{i,j} = \int_\Omega \phi_i\phi_j dx$ exactly. Moreover, since the hats $\phi_i$ and $\phi_j$ lack common support for $|i - j| > 1$ only $M_{i,i}$, $M_{i,i+1}$ , and $M_{i+1,i}$ need to be calculated. All other matrix entries are zero by default. This is clearly seen from Figure 2, (Larson and Bengzon, 2010) showing two neighboring hat functions and their support. As a consequence, the mass matrix $M$ is tridiagonal.

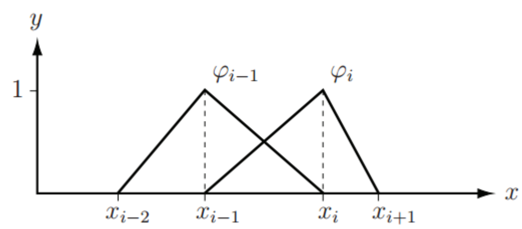



Figure 2: Illustration of the hat functions $\phi_{i-1}$ and $\phi_i$, in this figure $\varphi$, and their support.

On the interval $I = (a, b)$, Simpson's formula is of the form, (Larson and Bengzon, 2010)

$$\int_a^b f(x)dx = \frac{f(a) + 4f(\frac{a+b}{2}) + f(b)}{6}(b-a). \tag{14}$$

Let we start on the diagonal entries of $M$, $M_{i,i}$ using Simpson's formula:

$$\begin{aligned} M_{i,i} &= \int_\Omega \phi_i^2 dx, \\ &= \int_{x_{i-1}}^{x_i} \phi_i^2 dx + \int_{x_i}^{x_{i+1}} \phi_i^2 dx, \\ &= \frac{0 + 4(\frac{1}{2})^2 + 1}{6} h_i + \frac{1 + 4(\frac{1}{2})^2 + 0}{6} h_{i+1}, \\ &= \frac{h_i}{3} + \frac{h_{i+1}}{3}, \text{ for } i = 1, 2, \ldots, n-1, \end{aligned}$$

where $h_{i+1} = x_{i+1} - x_i$ and $h_i = x_i - x_{i-1}$, but in our case we use a uniform mesh length $h = x_{i+1} - x_i = x_i - x_{i-1}$ and $M_{ii} = \frac{2h}{3}$. The first and last diagonal entry are $M_{00} = \frac{h_1}{3}$ and $M_{nn} = \frac{h_n}{3}$, respectively, since the hat functions $\phi_0$ and $\phi_n$ are only half.

We know continue with the sub diagonal entries $M_{i+1,i}$ still using Simpson's formula we have

$$\begin{aligned} M_{i+1,i} &= \int_\Omega \phi_{i+1}\phi_i dx, \\ &= \int_{x_i}^{x_{i+1}} \phi_{i+1}\phi_i dx, \\ &= \frac{1.0 + 4(\frac{1}{2})^2 + 0.1}{6} h_{i+1} \\ &= \frac{h_{i+1}}{6}, \text{ for } i = 0, 1, 2, \ldots, n-1. \end{aligned}$$

By using a similar calculation the super diagonal entries $M_{i,i+1}$ have the values

$$M_{i,i+1} = \frac{h_{i+1}}{6}, \text{ for } i = 0, 1, 2, \ldots, n-1.$$

Hence, the mass matrix $m$ takes the form

$$M = \begin{bmatrix} \frac{h_1}{3} & \frac{h_1}{6} & & & & & \\ \frac{h_1}{6} & \frac{h_1}{3} + \frac{h_2}{3} & \frac{h_2}{6} & & & & \\ & \frac{h_2}{6} & \frac{h_2}{3} + \frac{h_3}{3} & \frac{h_3}{6} & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & \frac{h_{n-1}}{6} & \frac{h_{n-1}}{3} + \frac{h_n}{3} & \frac{h_n}{6} \\ & & & & \frac{h_n}{6} & \frac{h_n}{3} \end{bmatrix}. \tag{15}$$

The global mass matrix $M$ can be written as a sum of $n$ simpler elemental matrices as:

$$M = \begin{bmatrix} \frac{h_1}{3} & \frac{h_1}{6} & & \\ \frac{h_1}{6} & \frac{h_1}{3} & & \\ & & & \\ & & & \end{bmatrix} + \begin{bmatrix} & & & \\ & \frac{h_2}{3} & \frac{h_2}{6} & \\ & \frac{h_2}{6} & \frac{h_2}{3} & \\ & & & \end{bmatrix} + \ldots + \begin{bmatrix} & & \\ & \frac{h_n}{3} & \frac{h_n}{6} \\ & \frac{h_n}{6} & \frac{h_n}{3} \end{bmatrix}.$$

i.e., $M = M^{\Omega_1} + M^{\Omega_2} + \ldots + M^{\Omega_n}$.

Each matrix $M^{\Omega_e}, i = 1, 2, \ldots, n$, is obtained by restricting the integration to one sub interval or element $\Omega_e$ and is therefore called an element mass matrix. From the sum we see that on each element $e$ this small block takes the form: $M^e = \frac{h}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$, where $h$ is the length of $e$. We refer to $M^e$ as the local element mass matrix. The summation of the element mass matrices into the global mass matrix is called assembling.

Now the system of equation, that is Eq.12 is a simple system of ordinary differential equations. To solve this system of ode's, we have to use a back ward Euler method and a Matlab soft ware to solve the system at each time steps using an initial condition.

**Implementation of the $1D$ Diffusion Equation**

Let we now consider the diffusion equation

$$\frac{\partial u}{\partial t} = D\frac{\partial^2 u}{\partial x^2}.$$

We multiply the equation by a test function $v(x)$, which satisfies the boundary conditions $v(0) = v(1) = 0$ and then integrating by parts we have that:

$$\frac{\partial u}{\partial t}.v = D\frac{\partial^2 u}{\partial x^2}.v$$

$$\int_0^1 \left(\frac{\partial u}{\partial t}.v\right) = \int_0^1 \left(D\frac{\partial^2 u}{\partial x^2}.v\right)$$

Hence by using integration by parts, we obtain

$$\int_0^1 \frac{\partial u}{\partial t}.v = -D\int_0^1 \frac{\partial u}{\partial x}.\frac{\partial v}{\partial x}, \tag{16}$$

which is the weak formulation of the one dimensional Diffusion equation.

Now substituting Eq.10 and Eq.11 in the weak formulation of the equation Eq.16, we have:

$$\int_0^1 \frac{\partial}{\partial t}\sum_{i=1}^{N-1} u_i\phi_i.\sum_{j=1}^{N-1} v_j\phi_j = -D\int_0^1 \frac{\partial}{\partial x}\sum_{i=1}^{N-1} u_i\phi_i.\frac{\partial}{\partial x}\sum_{j=1}^{N-1} v_j\phi_j.$$

which then implies,

$$\sum_{j=1}^{N-1} v_j \left(\frac{\partial}{\partial t}\sum_{i=1}^{N-1} u_i \int_0^1 \phi_i.\phi_j\right) = \sum_{j=1}^{N-1} v_j \left(-D\sum_{i=1}^{N-1} u_i \int_0^1 \phi_i'.\phi_j'\right).$$

That is

$$\frac{\partial}{\partial t}\sum_{i=1}^{N-1} u_i \int_0^1 \phi_i.\phi_j = -D\sum_{i=1}^{N-1} u_i \int_0^1 \phi_i'.\phi_j'.$$

In a matrix form it can be written as:

$$M\dot{U} + DAU = 0, \tag{17}$$

where $M$ is the mass matrix with entries given in Eq. 15 and $A$ is the stiffness matrix with entries given in Eq. 19.

Here the matrix $A$ is often referred to as the stiffness matrix, a name coming from corresponding matrices in the context of structural problems.

## Assembly of the stiffness matrix in $1D$

The stiffness matrix $A$ is symmetric for this simple problem, which makes the computation of the matrix faster since we don't have to compute all of the elements, symmetric matrices are also much faster to invert. Here $\phi_i$'s are the hat functions given in Eq. 7, the entries of each element of the stiffness matrix $A$ is given by

$$A_{i,j} = \int_0^1 \phi_i' \phi_j' dx,$$
$$= \sum_{e=1}^{N-1} \int_{\Omega_e} \phi_i' \phi_j' dx,$$
$$= \sum_{e=1}^{N-1} A_{ij}^e.$$

similarly the load vector

$$F_i = \int_0^1 f\phi_i dx = \sum_{e=1}^{N-1} \int_{\Omega_e} f\phi_i dx = \sum_{e=1}^{N-1} F_i^e.$$

On the interval $I = (a, b)$ Simpson's formula is of the form, (Larson and Bengzon, 2010)

$$\int_a^b f(x)dx = \frac{f(a) + 4f(\frac{a+b}{2}) + f(b)}{6}(b-a). \tag{18}$$

Then we can be illustrate by the hat functions and Simpsons formula as follows:

$$\phi_i(x) = \begin{cases} \frac{x-x_{i-1}}{h_i}, & \text{if } x_{i-1} \le x \le x_i \\ \frac{x_{i+1}-x}{h_{i+1}}, & \text{if } x_i \le x \le x_{i+1} \\ 0, & \text{other wise} \end{cases}, \quad \phi_{i-1}(x) = \begin{cases} \frac{x-x_{i-2}}{h_{i-1}}, & \text{if } x_{i-2} \le x \le x_{i-1} \\ \frac{x_i-x}{h_i}, & \text{if } x_{i-1} \le x \le x_i \\ 0, & \text{other wise} \end{cases},$$

and $\phi_{i+1}(x) = \begin{cases} \frac{x-x_i}{h_{i+1}}, & \text{if } x_i \le x \le x_{i+1} \\ \frac{x_{i+2}-x}{h_{i+2}}, & \text{if } x_{i+1} \le x \le x_{i+2} \\ 0, & \text{other wise} \end{cases}$.

Now

$$A_{i,i} = a(\phi_i, \phi_i)^e = \int_{\Omega_e} \phi_i' \phi_i' dx,$$
$$= \int_{x_{i-1}}^{x_i} \phi_i' \phi_i' dx + \int_{x_i}^{x_{i+1}} \phi_i' \phi_i' dx,$$
$$= \int_{x_{i-1}}^{x_i} \frac{1}{h_i}\frac{1}{h_i} dx + \int_{x_i}^{x_{i+1}} \frac{-1}{h_{i+1}}\frac{-1}{h_{i+1}} dx,$$
$$= \frac{h_i}{6}\left(\frac{1}{h_i^2} + \frac{4}{h_i^2} + \frac{1}{h_i^2}\right) + \frac{h_{i+1}}{6}\left(\frac{1}{h_{i+1}^2} + \frac{4}{h_{i+1}^2} + \frac{1}{h_{i+1}^2}\right),$$
$$= \frac{1}{h_i} + \frac{1}{h_{i+1}}.$$

,

$$A_{i-1,i} = a(\phi_{i-1}, \phi_i)^e = \int_{\Omega_e} \phi_{i-1}' \phi_i' dx,$$
$$= \int_{x_{i-1}}^{x_i} \phi_{i-1}' \phi_i' dx + \int_{x_i}^{x_{i+1}} \phi_{i-1}' \phi_i' dx,$$
$$= \int_{x_{i-1}}^{x_i} \frac{-1}{h_i}\frac{1}{h_i} dx,$$
$$= \frac{h_i}{6}\left(\frac{-1}{h_i^2} + \frac{-4}{h_i^2} + \frac{-1}{h_i^2}\right) = \frac{-1}{h_i}.$$

and

$$
\begin{aligned}
A_{i,i+1} = a(\phi_{i+1},\phi_i)^e &= \int_{\Omega_e} \phi'_{i+1}\phi'_i dx,\\
&= \int_{x_{i-1}}^{x_i} \phi'_{i+1}\phi'_i dx + \int_{x_i}^{x_{i+1}} \phi'_{i+1}\phi'_i dx,\\
&= \int_{x_i}^{x_{i+1}} \frac{-1}{h_{i+1}}\frac{1}{h_{i+1}} dx,\\
&= \frac{h_{i+1}}{6}\left(\frac{-1}{h_{i+1}^2} + \frac{-4}{h_{i+1}^2} + \frac{-1}{h_{i+1}^2}\right) = \frac{-1}{h_{i+1}}.
\end{aligned}
$$

Each generic interior element contributes to the stiffness matrix of a $2\times 2$ sub matrix.

$$
A = \int_0^1 \phi'_i\phi'_j dx = \sum_{e=1}^{N-1} A^e = \begin{bmatrix}
\frac{1}{h_1} & \frac{-1}{h_1} & & & & \\
\frac{-1}{h_1} & \frac{1}{h_1}+\frac{1}{h_2} & \frac{-1}{h_2} & & & \\
 & -\frac{-1}{h_2} & \frac{1}{h_2}+\frac{1}{h_3} & \frac{-1}{h_3} & & \\
 & & \ddots & \ddots & \ddots & \\
 & & & \frac{-1}{h_{n-1}} & \frac{1}{h_{n-1}}+\frac{1}{h_n} & \frac{-1}{h_n} \\
 & & & & \frac{-1}{h_n} & \frac{1}{h_n}
\end{bmatrix}. \tag{19}
$$

The global stiffness matrix $A$ can be written as a sum of $n$ simpler elemental matrices as:

$$
A = \frac{1}{h_1}\begin{bmatrix} 1 & -1 & & \\ -1 & 1 & & \\ & & & \\ & & & \end{bmatrix} + \frac{1}{h_2}\begin{bmatrix} & & & \\ & 1 & -1 & \\ & -1 & 1 & \\ & & & \end{bmatrix} + \ldots + \frac{1}{h_n}\begin{bmatrix} & & & \\ & & & \\ & & 1 & -1 \\ & & -1 & 1 \end{bmatrix}.
$$

i.e., $A = A^{\Omega_1} + A^{\Omega_2} + \ldots + A^{\Omega_n}$.
Each matrix $A^{\Omega_e}, e = 1,2,\ldots,n$, is obtained by restricting the integration to one sub interval or element $\Omega_e$ and is therefore called an element stiffness matrix. From the sum we see that on each element $e$ this small block takes the form: $A^e = \frac{1}{h}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$, where $h$ is the length of $e$. We refer to $A^e$ as the local element stiffness matrix.

Now the system of equation, that is Eq.17 also is a simple system of ordinary differential equations and we can solve this system of Ode's to get the solution of the original PDE.

## Implementation of One Dimensional Advection Diffusion equation

Let us now solve the $1D$ governing (advection diffusion) equation

$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = D\frac{\partial^2 u}{\partial x^2} + f,\ u(0) = u(1) = 0,$

by using FEM. Let we find the weak formulation of the equation by multiplying the equation with a test function $v(x)$, which satisfies the boundary conditions $v(0) = v(1) = 0$ and then integrating by parts as the same procedure above. we have that:

$$
\frac{\partial u}{\partial t}.v + a\frac{\partial u}{\partial x}.v = D\frac{\partial^2 u}{\partial x^2}.v + f.v,
$$

$$
\int_0^1 \left(\frac{\partial u}{\partial t}.v + a\frac{\partial u}{\partial x}.v\right) = \int_0^1 \left(D\frac{\partial^2 u}{\partial x^2}.v + f.v\right).
$$

Using integration by parts, we have then

$$\int_0^1 \frac{\partial u}{\partial t}.v + \int_0^1 a\frac{\partial u}{\partial x}.v = -D\int_0^1 \frac{\partial u}{\partial x}.\frac{\partial v}{\partial x} + \int_0^1 f.v, \tag{20}$$

which is the weak formulation of the one dimensional advection-Diffusion equation. Substituting Eq.10 and Eq.11 in the weak formulation of the equation Eq.20, we have:

$$\int_0^1 \frac{\partial}{\partial t}\sum_{i=1}^{N-1} u_i\phi_i.\sum_{j=1}^{N-1} v_j\phi_j + \int_0^1 a\frac{\partial}{\partial x}\left(\sum_{i=1}^{N-1} u_i\phi_i\right).\sum_{j=1}^{N-1} v_j\phi_j$$
$$= -D\int_0^1 \frac{\partial}{\partial x}\sum_{i=1}^{N-1} u_i\phi_i.\frac{\partial}{\partial x}\sum_{j=1}^{N-1} v_j\phi_j + \int_0^1 f.\sum_{j=1}^{N-1} v_j\phi_j.$$

which then implies,

$$\sum_{j=1}^{N-1} v_j\left(\frac{\partial}{\partial t}\sum_{i=1}^{N-1} u_i\int_0^1 \phi_i.\phi_j + a\sum_{i=1}^{N-1} u_i\int_0^1 \phi_i'.\phi_j\right) = \sum_{j=1}^{N-1} v_j\left(-D\sum_{i=1}^{N-1} u_i\int_0^1 \phi_i'.\phi_j' + \int_0^1 f.\phi_j\right). \tag{21}$$

That is

$$\frac{\partial}{\partial t}\sum_{i=1}^{N-1} u_i\int_0^1 \phi_i.\phi_j + a\sum_{i=1}^{N-1} u_i\int_0^1 \phi_i'.\phi_j = -D\sum_{i=1}^{N-1} u_i\int_0^1 \phi_i'.\phi_j' + \int_0^1 f.\phi_j.$$

In a matrix form it can be written as:

$$M\dot{U} + aBU + DAU = F, \tag{22}$$

where $M$ is the mass matrix with entries given in Eq. 15, $B$ is a matrix with entries given in Eq. 13, $A$ is the stiffness matrix given in Eq. 19 and $F$ is a load vector given in Eq. 25.

## Assembly of the Load Vector in $1D$

The right-hand-side, load vector of Eq. 20 contains an integral over a function $f(x)$. In general, exactly computing this integral is very difficult, so another numerical approximation is required. We can use a well known integration rule composite simpson rule to approximate these integration whose formula (for more information you can see, (?)) is given in Eq. 23, by selecting a set of distinct $N$ nodes in the interval $[a,b]$ with $h = \frac{b-a}{N}$, $x_i = a+ih$ for each $i = 0, 1, \ldots, N$:

$$\int_a^b f(x)dx = \frac{h}{3}\left(f(x_0) + 2\sum_{j=1}^{\frac{N}{2}-1} f(x_{2j}) + 4\sum_{j=1}^{\frac{N}{2}} f(x_{2j-1}) + f(x_N)\right),\ j = 1, 2, ..., (\frac{N}{2}) - 1. \tag{23}$$

Using the another quadrature rule for simplicity, for instance, using the Trapezoidal rule, (Larson and Bengzon, 2010)

$$\int_a^b f(x)dx = \frac{f(a)+f(b)}{2}(b-a), \tag{24}$$

we have

$$\begin{aligned}
f_i &= \int_I f\phi_i dx, \\
&= \int_{x_{i-1}}^{x_{i+1}} f\phi_i dx, \\
&= \int_{x_{i-1}}^{x_i} f\phi_i dx + \int_{x_i}^{x_{i+1}} f\phi_i dx, \\
&\approx \frac{f(x_{i-1})\phi_i(x_{i-1}) + f(x_i)\phi_i(x_i)}{2} h_i + \frac{f(x_{i+1})\phi_i(x_{i+1}) + f(x_i)\phi_i(x_i)}{2} h_{i+1}, \\
&= \frac{0 + f(x_i)}{2} h_i + \frac{f(x_i)+0}{2} h_{i+1}, \\
&= f(x_i)\left(\frac{h_i}{2} + \frac{h_{i+1}}{2}\right).
\end{aligned}$$

Now using this trapezoidal method, the approximate load vector takes the form

$$F = \begin{pmatrix} f(x_0)\frac{h_1}{2} \\ f(x_1)\left(\frac{h_1+h_2}{2}\right) \\ f(x_2)\left(\frac{h_2+h_3}{2}\right) \\ \vdots \\ f(x_{n-1})\left(\frac{h_{n-1}+h_n}{2}\right) \\ f(x_n)\frac{h_n}{2} \end{pmatrix}. \tag{25}$$

Splitting $F$ into a sum over the elements yields the $n$ global element load vectors $F^{\Omega_e}$:

$$F = \begin{pmatrix} f(x_0) \\ f(x_1) \\ \\ \\ \\ \end{pmatrix}\frac{h_1}{2} + \begin{pmatrix} \\ f(x_1) \\ f(x_2) \\ \\ \\ \end{pmatrix}\frac{h_2}{2} + \begin{pmatrix} \\ \\ f(x_2) \\ f(x_3) \\ \\ \end{pmatrix}\frac{h_3}{2} + \ldots + \begin{pmatrix} \\ \\ \\ f(x_{n-1}) \\ f(x_n) \end{pmatrix}\frac{h_n}{2}$$

i.e., $F = F^{\Omega_1} + F^{\Omega_2} + \ldots + F^{\Omega_n}$. Each vector $F^{\Omega_e}, e = 1, 2, \ldots, n$, is formally derived by restricting the integration to element $\Omega_e$.

Now the system of equation, that is Eq.22 can be written in the form:

$$M\dot{U} + (aB + DA)\,U = F, \tag{26}$$

which is a simple system of ordinary differential equations. For solving this system of Ode's, we have to use a Matlab soft ware in which it has a number of tools for numerically solving ordinary differential equations. We would focus on the back ward Euler method to descritize time.

## FEM implementation of the Two Dimensional AD equation

The $2D$ advection diffusion equation with the same and constant velocity and diffusion term is given by

$u_t + a(u_x + u_y) = D(u_{xx} + u_{yy}) + f,$
$(x, y) \in \Omega = [0, 1],$

with homogeneous boundary conditions. That is

$$u_t + a\nabla u = D\nabla^2 u + f.$$

Now to find a weak formulation for this $2D$ equation, we multiply both sides of the equation with a test function $v = v(x,y) \in V$ which satisfies the boundary conditions.

$$u_t.v + a\nabla u.v = D\nabla^2 u.v + fv.$$

Here integrating this over the domain $\Omega$ yields the following:

$$\int_\Omega (u_t.v + a\nabla u.v) = \int_\Omega \left(D\nabla^2 u.v + fv\right).$$

We see from the 2D FEM of Poisson equation (using Gauss theorem and the transformation of a surface integral to a line integral) that $\int_\Omega v\nabla^2 u = -\int_\Omega \nabla v\nabla u$, and hence we get:

$$\int_\Omega (u_t.v + a\nabla u.v) = -D\int_\Omega (\nabla v\nabla u) + \int_\Omega fv.$$

$$\int_\Omega u_t.v = -\int_\Omega (D\nabla v\nabla u + a\nabla u.v) + \int_\Omega fv. \tag{27}$$

Here Eq. 27 is the weak formulation and it can be simplified as

$$(u_t, v) = l(u,v) + (f,v)\ \forall v \in V, \tag{28}$$

where $(u_t, v) = \int_\Omega u_t.v$ and $l(u,v) = -\int_\Omega (D\nabla v\nabla u + a\nabla u.v)$.

Given a FE space $V$, with $\phi_i(x,y),\ i = 1, 2, \ldots, N$ denoting a set of basis functions for $V$, we seek the FE solution of form

$$u_h(x,y,t) = \sum_{j=1}^{N} u_j(t)\phi_j(x,y). \tag{29}$$

And taking the test function $v(x,y)$ as a linear combination of basis functions

$$v(x,y) = \sum_{j=1}^{N} v_j\phi_j, \text{ with } v_j \text{ are constants.} \tag{30}$$

Substituting this expression (29 and 30) into eq.28, we obtain

$$\left(\sum_{j=1}^{N} u_j'(t)\phi_j(x,y), \sum_{j=1}^{N} v_j\phi_j\right) = l\left(\sum_{j=1}^{N} u_j(t)\phi_j(x,y), \sum_{j=1}^{N} v_j\phi_j\right) + (f, \sum_{j=1}^{N} v_j\phi_j). \tag{31}$$

Then we get the linear system of ordinary differential equations in the $u_j(t)$ as:

$$\left(u_1'(t)\phi_1(x,y) + \sum_{j=2}^{N} u_j'(t)\phi_j(x,y), \sum_{j=1}^{N} v_j\phi_j\right) = l\left(u_1(t)\phi_1(x,y) + \sum_{j=1}^{N} u_j(t)\phi_j(x,y), \sum_{j=1}^{N} v_j\phi_j(x,y)\right) +$$

$$(f, \sum_{j=1}^{N} v_j\phi_j(x,y)),$$

$$\left(u_1'(t)\phi_1, \sum_{j=1}^{N} v_j\phi_j\right) + \left(\sum_{j=2}^{N} u_j'(t)\phi_j, \sum_{j=1}^{N} v_j\phi_j\right) = l\left(u_1(t)\phi_1, \sum_{j=1}^{N} v_j\phi_j\right) +$$

$$l\left(\sum_{j=1}^{N} u_j(t)\phi_j, \sum_{j=1}^{N} v_j\phi_j\right) + (f, \sum_{j=1}^{N} v_j\phi_j),$$

$$\sum_{j=1}^{N}\left(u_j'(t)\phi_j, \sum_{j=1}^{N} v_j\phi_j\right) = \sum_{j=1}^{N} l\left(u_j(t)\phi_j, \sum_{j=1}^{N} v_j\phi_j\right) + (f, \sum_{j=1}^{N} v_j\phi_j),$$

$$\sum_{j=1}^{N}\sum_{j=1}^{N}\left(u_j'(t)\phi_j, v_j\phi_j\right) = \sum_{j=1}^{N}\sum_{j=1}^{N} l\left(u_j(t)\phi_j, v_j\phi_j\right) + \sum_{j=1}^{N}(f, v_j\phi_j).$$

Since $v_j$'s are constants we have also that:

$$\sum_{j=1}^{N} v_j \sum_{j=1}^{N}(\phi_j, \phi_j)\, u_j'(t) = \sum_{j=1}^{N} v_j \sum_{j=1}^{N} l\,(\phi_j, \phi_j)\, u_j(t) + \sum_{j=1}^{N} v_j(f, \phi_j).$$

The corresponding problem can therefore be expressed as

$$V^T M\dot{U} = V^T AU + V^T F.$$

That is

$$M\dot{U} = AU + F. \tag{32}$$

$$\text{Where, } M = \begin{pmatrix} (\phi_1,\phi_1) & (\phi_1,\phi_2) & \dots & (\phi_1,\phi_N) \\ (\phi_2,\phi_1) & (\phi_2,\phi_2) & \dots & (\phi_2,\phi_N) \\ \vdots & \vdots & \vdots & \\ (\phi_N,\phi_1) & (\phi_N,\phi_2) & \dots & (\phi_N,\phi_N) \end{pmatrix},$$

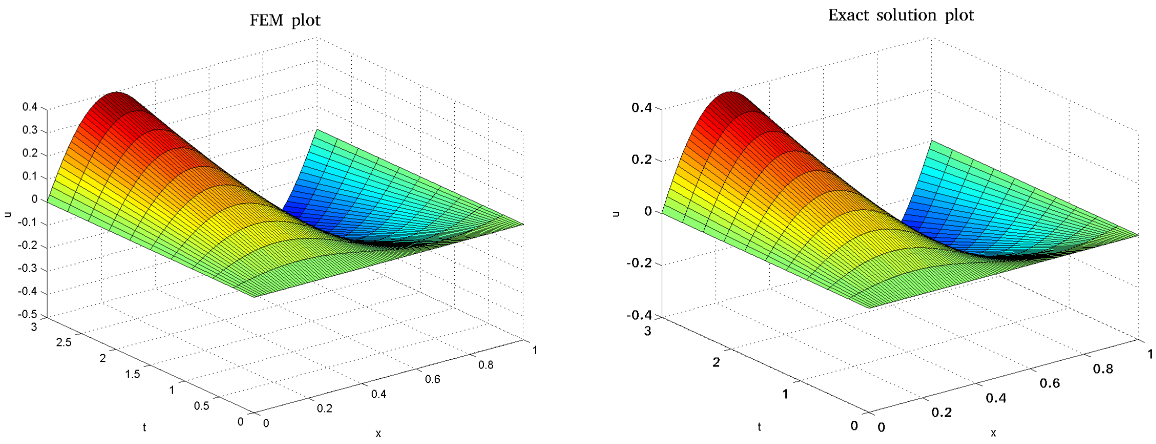
$$A = \begin{pmatrix} l(\phi_1,\phi_1) & l(\phi_1,\phi_2) & \dots & l(\phi_1,\phi_N) \\ l(\phi_2,\phi_1) & l(\phi_2,\phi_2) & \dots & l(\phi_2,\phi_N) \\ \vdots & \vdots & \vdots & \\ l(\phi_N,\phi_1) & l(\phi_N,\phi_2) & \dots & l(\phi_N,\phi_N) \end{pmatrix}, \text{ and } F = \begin{pmatrix} (f,\phi_1) \\ (f,\phi_2) \\ \vdots \\ (f,\phi_N) \end{pmatrix}.$$

There are many methods to solve the above problem involving the system of first order ODE. We can use FD methods that will descritize in time by using Explicit Euler method, Implicit Euler method or the Crank-Nicolson method, by considering an appropriate initial condition, (Johnson, 2012). But for this paper in the two dimensional case, we can use the ODE Suite in Matlab which is the Matlab build in system of ODE solver, ODE15I.

## NUMERICAL RESULTS AND DISCUSSION

In this section, we are compared for the advection-diffusion equations with an exact solution for the given finite element methods and then we solve the equation with out knowing the exact solution. The comparison is carried out by means of computed solutions for a wide range of characteristic parameters. Linear elements are employed at the discretization in case of one-dimensional problem and bilinear elements in case of two-dimensional problems. Dirichlet and general boundary conditions are considered with different initial conditions and different size of computational domain.

First, consider the advection equation $u_t + au_x = f$ with sources and sinks function, $f(x,t) = \cos(\pi x) - at\pi\sin(\pi x)(x - x^2) + t\cos(\pi x)(1 - 2x)$ and the velocity parameter $a = 3$, with homogeneous boundary conditions. The FM solution using Matlab with a back ward Euler descritization in time is given in Fig. 3(a), with $N = 100$ nodes. To saw our error the exact solution for this equation is $u(x,t) = t\cos(\pi x)(x - x^2)$ and its graph is Fig. 3(b). The error of this equation is an order of $10^{-4}$. If we increase the number of nodes from $N = 100$ to $N = 1000$, then our numerical solution becomes more accurate and we saw that the error is an order of $10^{-7}$ by modifying $h = 0.001$ from the algorithm.

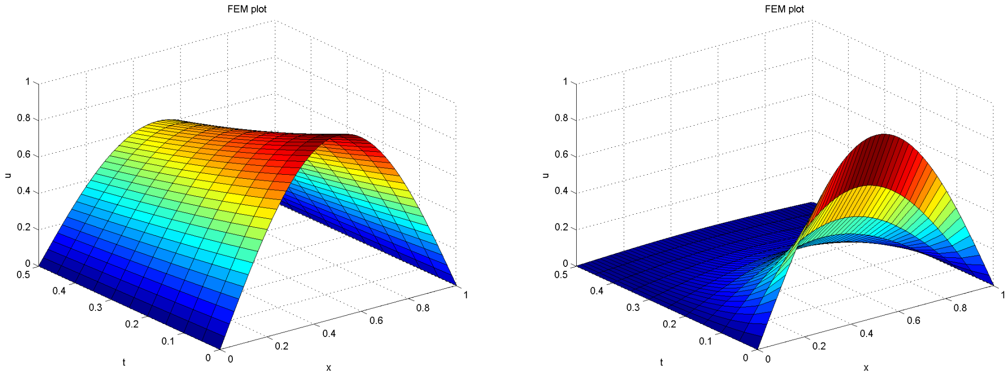



(a) FEM solution (b) Exact solution

Figure 3: The Matlab implementation of the advection dominated equation with $f(x,t) = \cos(\pi x) - at\pi\sin(\pi x)(x - x^2) + t\cos(\pi x)(1 - 2x)$.

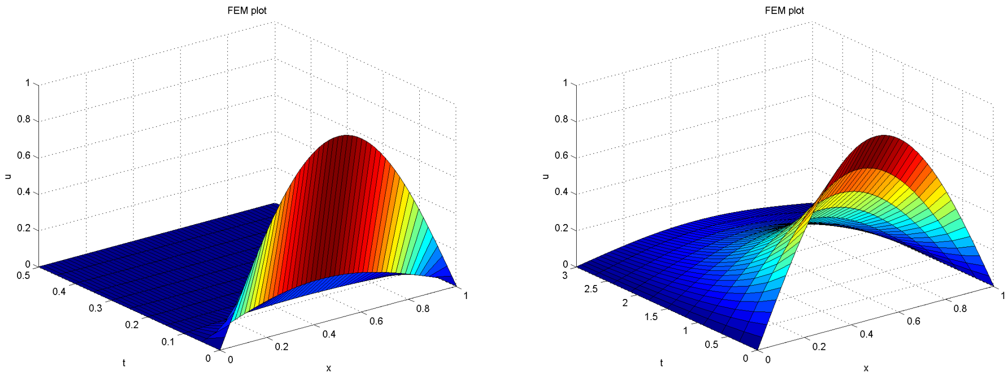
In (Bergara, 2011), there is a diffusion ($u_t = Du_{xx}$) example solved with finite difference methods, and let we solve that equation with the finite element method. He solves the diffusion (heat) equation by using initial condition $u(x,0) = \sin(\pi x)$ and with homogeneous Dirichlet boundary conditions in the interval $0 \leq x \leq 1$. If we consider different values for final time and diffusion coefficient we get the following simulations (Fig. 4 to Fig. 6) with a similar descritization of space ($x$) in to 50 nodes.

When we saw the results of those figures (Fig. 4 to Fig. 6) we can observe the following. If we use a small amount of diffusion coefficient, then it has a mall diffusion process and if we consider relatively large diffusion coefficient, the diffusion process is faster. Hence our finite element method is reasonable and accurate.




(a) FEM sol. with $D = 0.1$ and $t_f = 0.5$ (b) FEM sol. with $D = 1$ and $t_f = 0.5$

Figure 4: Solution of the diffusion equation using $D = 0.1$ and $D = 1$ for a constant final time of $t_f = 0.5$.

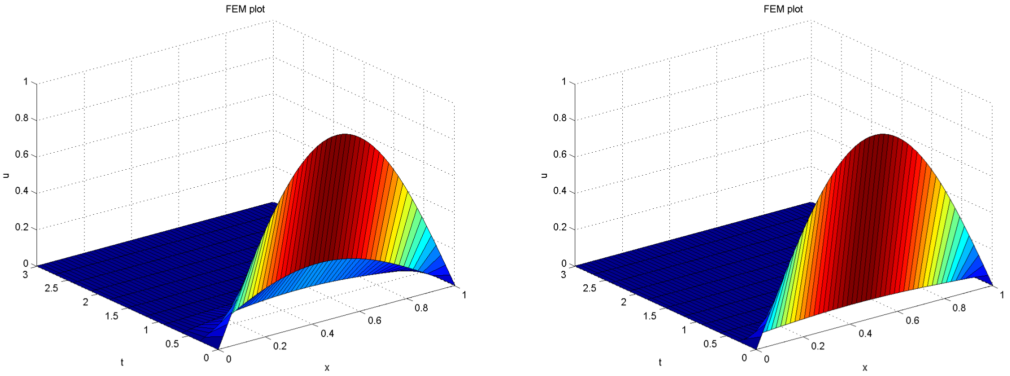



(a) FEM sol. with $D = 10$ and $t_f = 0.5$ (b) FEM sol. with $D = 0.1$ and $t_f = 3$

Figure 5: Solution of the diffusion equation using $D = 10$ with $t_f = 0.5$ and $D = 0.1$ for final time of $t_f = 3$.

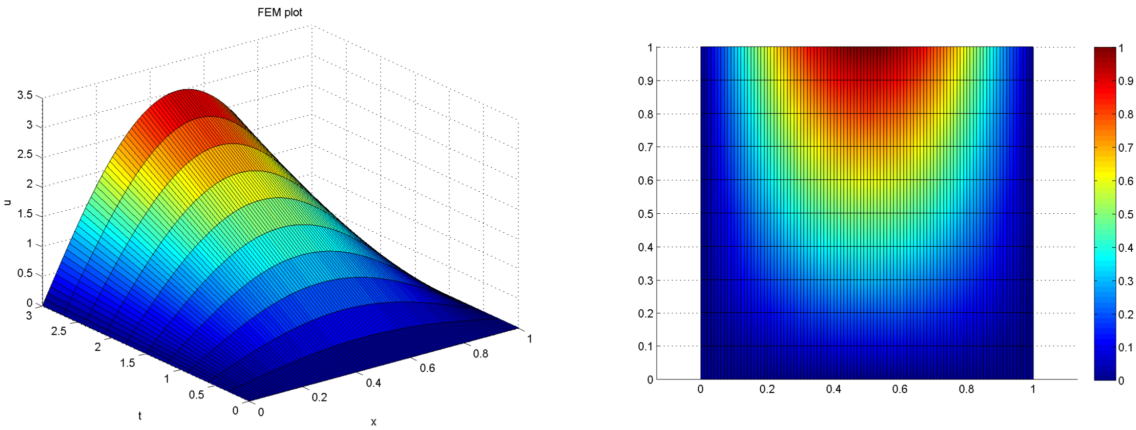
Let we know consider the ADE $u_t + au_x - Du_{xx} = f$ with $f(x,t) = \sin(\pi x)\,(1 + D\pi^2 t) + a\pi t\cos(\pi x)$, $u(x,0) = 0,\ u(0,t) = u(1,t) = 0$. By using the forward Euler descritization in time the Matlab implementation with $a = 0.03$ and $D = 10$ for this problem is given in Fig. 7. The peclet's number $Pe = \frac{al}{D} = 0.003 \ll 1$ which is diffusion dominated for this choice of $a$ and $D$.

Since we implements zero flux concentrations of the pollutants in the boundaries, the total mass of the pollution should remain constant. These boundaries physically correspond to a system where the species is enclosed inside a mesh that it cannot penetrate, however, the mesh allows the fluids to diffuse through the flow field. We see that all masses eventually concentrate along the domain as we have seen in Fig. 7. This steady state corresponds to diffusive and advective fluxes balancing each other. The flow carries additional mass towards the domain, but the density gradient limits how much more mass can be deposited.




(a) FEM sol. with $D = 1$ and $t_f = 3$ (b) FEM sol. with $D = 10$ and $t_f = 3$

Figure 6: Solution of the diffusion equation using $D = 1$ and $D = 10$ for a constant final time of $t_f = 0.5$.



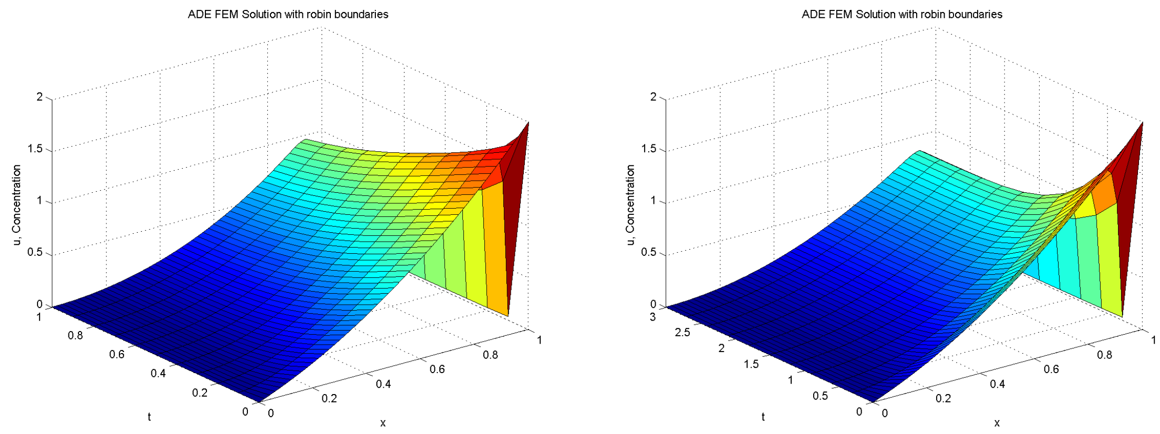

(a) FEM solution (b) Bar plot of the solution

Figure 7: Implementation of $1D$ AD equation with the source function$f(x,t) = sin(\pi x)\,(1 + D\pi^2 t) + a\pi t\cos(\pi x)$ and homogeneous Dirichlet boundary conditions and $a = 0.03,\ D = 10$, in which the peclet's number $Pe = \frac{al}{D} = 0.003 \ll 1$ which is diffusion dominated.

We see know on an arbitrary input concentration by using Neumann and Robin boundary conditions. Here, the problem is undetermined because of an unknown input concentration and hence unknown exit concentration and depends on the parameters. Authors of previous works on problems of this type have done on a known exit concentration by assuming a continuous and constant concentration at the flow boundary of Dirichlet type. This yields we to consider a problem by forcing the flow boundaries using Neumann and Robin boundary conditions. This conditions yields the flow of the concentration to move freely. Here the velocity term and the diffusion coefficient highly affects the flow of the concentration. We saw it by giving a source function $f(x,t) = a(2x+1) - 2D - 1,\ 0 \le x \le 1,\ 0 \le t \le t_f$ with a robin boundary condition $Du_x(0,t) = k_0(u(0,t) + g_0$ and $-Du_x(1,t) = k_1(u(1,t) + g_1$ and by considering different robin boundary parameters.

Let we change the velocity term and the diffusion coefficient for the given source term using $D = 0.02$ and $a = 1$, $(g_0, g_1) = (0.02, 0.06)$. The solution for this conditions is given in Fig. 8(a) using a total time of $t_f = 1$ and 8(b) using a total time of $t_f = 3$. This is an advection dominated example in which the movement of the concentration is faster in advection terms relative to diffusion.

2



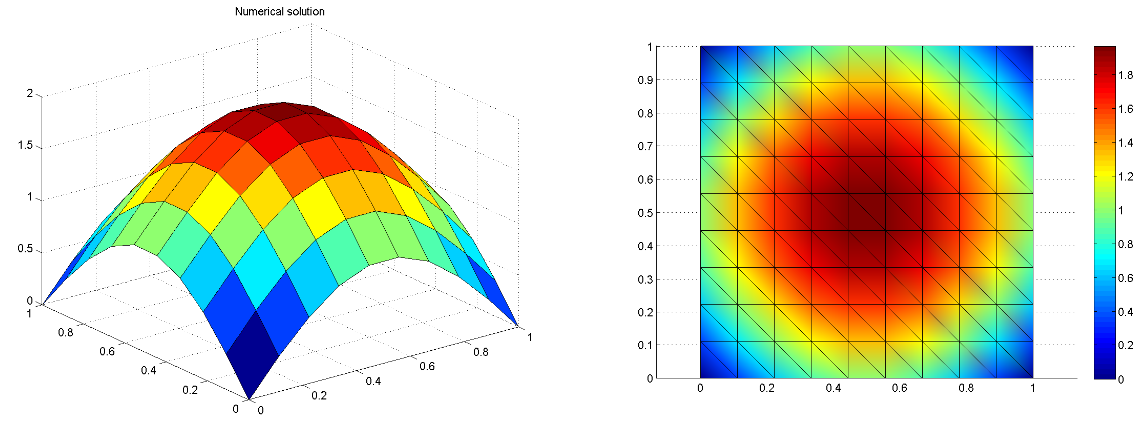

(a) FEM solution with $t_f = 1$

(b) FEM solution with $t_f = 3$

Figure 8: Numerical solution using FEM for the advection diffusion equation with $f(x,t) = a(2x+1) - 2D - 1$ and robin boundary conditions, with a diffusion coefficient $D = 0.02$ and velocity term $a = 1$ with time $t_f = 1$ for (a) and $t_f = 3$ for (b).

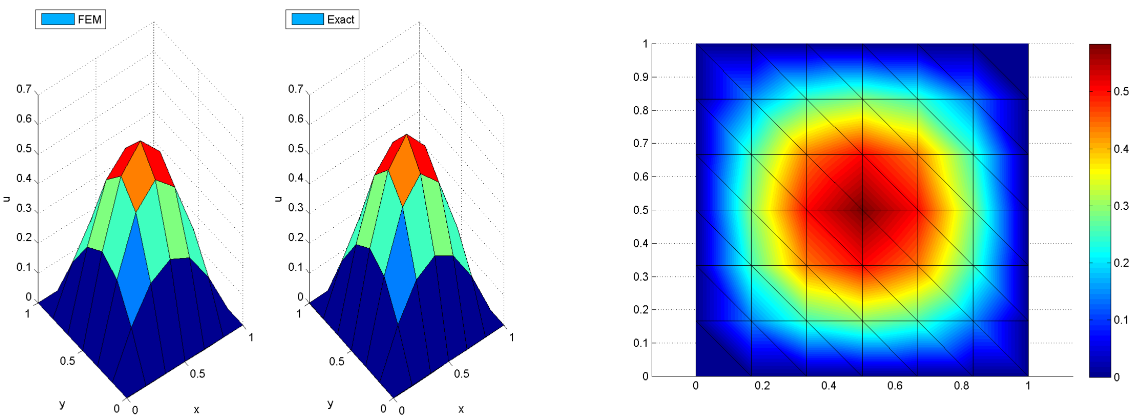
Let we consider the $2D$ steady diffusion equation with a source function given as $f(x,y) = D\pi^2(\sin(\pi x) + \sin(\pi y))$ to implement the $2D$ problem. The FEM numerical simulation of the PDE is given in Fig. 9.

In order to consider the finite element method of the time dependent two dimensional equations, we use the unsteady $2D$ diffusion equation. Let we test it with a known exact function and use the source terms and boundary conditions from that function. Let $u(t,x,y) = e^{-t}\sin(\pi x)\sin(\pi y)$ be the given exact solution for the diffusion equation $u_t - D\Delta u = f$, then our source function $f$ becomes $f(t,x,y) = (2D\pi^2 - 1)e^{-t}\sin(\pi x)\sin(\pi y)$. Now by using the ODE15I, ODE solver to integrate for time in the final finite element method descritization the solutions for the equation is given in Fig. 10. The surface plots of those figures are the exact solutions, the finite element method solutions and the color bar of the finite element solution to view its properties.




(a) FEM solution (b) Bar plot of the solution

Figure 9: The numerical simulation using FEM for the steady $2D$ diffusion equation with $f(x, y) = D\pi^2(\sin(\pi x) + \sin(\pi y))$, with a diffusion coefficient $D = 0.5$.

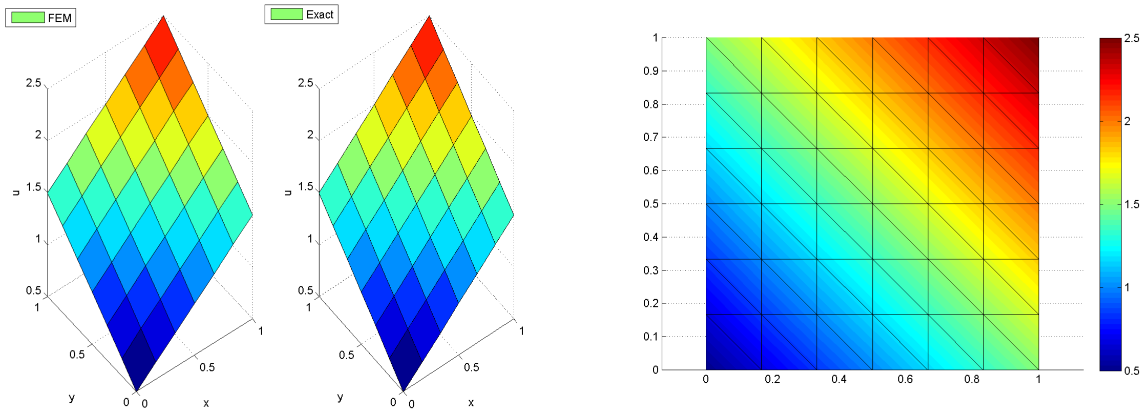



(a) FEM & exact sol (b) Colour bar of the FEM solution

Figure 10: The surface of time dependent two dimensional diffusion equation with $D = 100$ with its color bar plot for its source term $f$, $f(t, x, y) = (2D\pi^2 - 1)e^{-t} \sin(\pi x) \sin(\pi y)$.

If we consider another linear test example with exact solution $u(x, y, t) = x + y + t$, then its source function becomes $f(x, y, t) = 1$. The finite element solution for this $2D$ unsteady diffusion equation with its exact solution and color bar is given in Fig. 11 by ODE solver ODE15I.

The numerical results in all the above mentioned discussions using the finite element method are almost closed to the analytical solution in the case of the test examples. This shows us that the finite element method is one of the best numerical method to solve any differential equations numerically in any type of geometries.




(a) FEM & exact sol

(b) Colour bar of the FEM solution

Figure 11: The surface of time dependent two dimensional diffusion equation with $D = 1$ with its color bar plot for its source term $f$, $f(t, x, y) = 1$.

## CONCLUSION AND RECOMMENDATION

In this paper, we study the finite element solution of the advection-diffusion equation. In the finite element analysis, we approximate a function defined in a domain, $\Omega$, with a set of orthogonal basis functions with coefficients corresponding to the functional values at some node points. We deal with the numerical simulation of the advection-diffusion equation using the finite element method scheme in space and the backward Euler method in time. The main focus of the paper has been on the variational formulation techniques for the solution of the discrete Galerkin method and the other hand on computational analysis of different one and two dimensional PDE's. We have done numerical simulations for the above mentioned equations by considering the technique and using different test examples. The solution for the values at the nodes for the partial differential equations can be obtained by solving a linear system of equations involving the inversion of the sparse matrices.

We have solved the advection-diffusion equation for both one dimensional and two dimensional cases with the weighted residual (Galerkin) method of finite elements with constant velocity term and diffusion coefficient. These statements are supported by our test numerical investigations for the one and two dimensional Poisson equation, the advection equation, diffusion equation and the advection-diffusion equation. We have also seen the simulations of the equations by using the language of technical computing called Matlab and for time dependent equation using backward (implicit) Euler finite difference method for one dimensional problem and the built-in function ode15i to solve the system of ode's for two dimensional equations.

The generalization of the proposed Galerkin method to the three-dimensional advection diffusion is obtained within the current framework and an interested body will be done on this approach by including the reaction term. To obtain the solutions to this problems, the method can be extended to the least square finite element method, and another interested body will also be done on this approach. One can also extend this method using variable velocities and diffusion coefficients which vary with the time and space of the given dimension. The finite element method descretizes only in space and the time descritization is based on the finite difference methods. Hence the finite element method needs human power still now for the descritization process of time. So this is a very interesting area to do researches in the future to include the time descritization in the method.

## Acknowledgments

The core of this work was completed during the master study at Hawassa University and presented to former School of Mathematics and Statistical Science (currently Department of Mathematics) in partial fulfillment of the requirement for the Degree of Master. The authers would like to thank the Department of Mathematics at the University of Hawassa University. This work of has been partially supported by Hawassa University.

## References

Ahsan M. 2012. Numerical solution of the advection-diffusion equation using Laplace transform finite analytical method. *Int. J. River Basin Manag.* **10**(**2**): 177- 188.

Aragonés L., Pagán J.I., López I., Navarro- González F.J. and Villacampa Y. 2019. Galerkin's formulation of the finite elements method to obtain the depth of closure. *Sci. Total Environ.* **660**: 1256-1263.

Bajellan A.A.F. 2015. Computation of the convection diffusion equation by the fourth order compact finite difference method. Doctoral dissertation, Izmir Institute of Technology, Turkey.

Bergara A. 2011. Finite Difference Numerical Methods of Partial Differential Equations in Finance with MATLAB. *Master and Banca*. New York, Springer.

Brenner S.C., Scott L.R., & Scott L.R. 2008. *The mathematical theory of finite element methods. Vol , pp* 263-291. Springer, New York.

Donea J. and Huerta A. 2003. *Finite element methods for flow problems*. John Wiley & Sons.

Hundsdorfer W.1996. Numerical solution of advection-diffusion-reaction equations: lecture notes for Ph. D. course, 1996, Thomas Stieltjes Institute. *Department of Numerical Mathematics [NM]*, (9603).

Johnson C. 2012. *Numerical solution of partial differential equations by the finite element method*. Courier Corporation.

Langtangen H.P. 2003. *Computational partial differential equations: numerical methods and diffpack programming* (Vol. 2). Berlin: Springer.

Larson M.G. and Bengzon F. 2010. The finite element method: theory, implementation, and practice. *Texts in Computational Science and Engineering*, *10*, 23-44.

Lian Y., Ying Y., Tang S., Lin S., Wagner G.J. and Liu W.K. 2016. A Petrov–Galerkin finite element method for the fractional advection– diffusion equation. *Comput. Methods Appl. Mech. Eng.*, **309**: 388-410.

Lima S.A., Kamrujjaman M. and Islam M.S. 2021. Numerical solution of convection–diffusion–reaction equations by a finite element method with error correlation. *AIP Advances* **11**(**8**): 085225.

Pochai N. and Deepana R. 2011. A numerical computation of water quality measurement in a uniform channel using a finite difference method. *Procedia Eng.* **8**: 85-88.

Quarteroni A. and Quarteroni S. 2009. *Numerical models for differential problems* (Vol. 2). Milan: Springer.

Szymkiewicz R. and Gąsiorowski D. 2021. Adaptive method for the solution of 1D and 2D advection–diffusion equations used in environmental engineering. *J. Hydroinformatics* **23**(**6**): 1290-1311.

Yang W.Y., Cao W., Chung T.S. and Morris J. 2005. Applied Numerical Methods Using MATLAB. A John Wiley & Sons. *Hoboken, NJ.*