MIE 1210: Computational Fluid Mechanics and Heat Transfer, Assignment 3: 2D Covection-Diffusion Problem

Declan Bracken[1,*]

[1]*Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, ON M5R 0A3, Canada*

(Dated: November 17th, 2023)

I. INTRODUCTION

A. Finite Volume Discretization

The purpose of this assignment was the construction of a 2-dimensional finite volume computational fluid dynamics solver. The following derivation of the finite volume method applied to a convection-diffusion problem follows the works at [1] expanded to 2 dimensions. As the third assignment in the course, this project has been constrained to solving for the distribution of a state variable ϕ representing temperature using a predefined, steady-state velocity field.

$$\nabla \cdot (\rho \mathbf{u} \phi) = \nabla \cdot (\Gamma \nabla \phi) + S_\phi \tag{1}$$

Where ϕ is the state variable representing temperature, $\mathbf{u}$ is the velocity field, Γ is a constant coefficient describing the thermal conductance of the domain's material, ρ is the density of the fluid, and S_ϕ is the source term. Using the finite volume method, a control volume integration is performed on equation (1), yielding;

$$\int_A \mathbf{n} \cdot (\rho \mathbf{u} \phi)\, dA = \int_A \mathbf{n} \cdot (\Gamma \nabla \phi)\, dA + \int_{CV} S_\phi\, dV. \tag{2}$$

In this equation, the right hand side represents the transport of the variable ϕ through diffusive means along with it's creation or destruction in the volume. The left hand side of (2) represents the transport of the state variable through convective means. The flow must also satisfy the continuity equation:

$$\nabla \cdot (\rho \mathbf{u}) = 0 \tag{3}$$

For a 2D problem, and if we drop the source term for simplicity, this integration yields the partial differential equation:

$$(\rho u A \phi)_e - (\rho u A \phi)_w - (\rho v A \phi)_n + (\rho v A \phi)_s = \left[\Gamma A \frac{\partial \phi}{\partial x}\right]_e - \left[\Gamma A \frac{\partial \phi}{\partial x}\right]_w - \left[\Gamma A \frac{\partial \phi}{\partial y}\right]_n + \left[\Gamma A \frac{\partial \phi}{\partial y}\right]_s, \tag{4}$$

with continuity equation;

$$(\rho u A)_e - (\rho u A)_w + (\rho v A)_n - (\rho v A)_s = 0. \tag{5}$$

Where the subscripts n, s, e, and w correspond to the cardinal directions of the 2d grid denoting north, south, east and west when viewing the plane. It is then useful to define convective and diffusive coefficients for the discretization separately:

$$F_e = (\rho u)_e \qquad F_w = (\rho u)_w \tag{6}$$

$$D_e = \frac{\Gamma_e}{\delta x_e} \qquad D_w = \frac{\Gamma_w}{\delta x_w} \tag{7}$$

$$F_n = (\rho v)_n \qquad F_s = (\rho v)_s \tag{8}$$

$$D_n = \frac{\Gamma_n}{\delta y_n} \qquad D_s = \frac{\Gamma_s}{\delta y_s} \tag{9}$$

Where the F terms are convective fluxes and the D terms are diffusive fluxes per unit area in any direction.

To continue with the discretization, we must decide on an advection scheme to calculate the gradients $\frac{\partial \phi}{\partial x}$ and $\frac{\partial \phi}{\partial y}$. We will select between 2 schemes: central difference and first order upwind.

B. Central Difference Scheme

The central difference scheme weighs neighbouring node's equally, and uses a linear interpolation between the central node p and it's neighbor. Implementing central difference on equation (4) with the coefficients calculated using equations (6) though (9), we get;

$$\begin{aligned}
&\frac{F_e}{2}(\phi_P + \phi_E) - \frac{F_w}{2}(\phi_W + \phi_P) \\
&\quad - \frac{F_n}{2}(\phi_P + \phi_N) + \frac{F_s}{2}(\phi_S + \phi_P) = \\
&\quad D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W) \\
&\quad - D_n(\phi_P - \phi_N) + D_s(\phi_S - \phi_P).
\end{aligned} \tag{10}$$

By grouping like ϕ terms, we can rearrange equation (10) so that we can extract coefficients for each of the neighboring temperatures, leading to:

$$a_P \phi_P = a_W \phi_W + a_E \phi_E + a_S \phi_S + a_N \phi_N \tag{11}$$

a_W	a_E	a_S	a_N
$D_w + \frac{F_w}{2}$	$D_e - \frac{F_e}{2}$	$D_s + \frac{F_s}{2}$	$D_n - \frac{F_n}{2}$
$a_P = a_W + a_E + a_S + a_N + (F_e - F_w) + (F_s - F_n)$			

Table I: Coefficients for the discretized transport equation in 2D.

For boundary conditions, we'd add an additional source term to equation (11) and to a_p in table III.

* declan.bracken@mail.utoronto.ca

C. Upwind Scheme

The central difference scheme works well for a pure diffusion problem because it weighs each cardinal direction equally in how the state variable is distributed. In the case of diffusion this is true, however, applying the same scheme to convection could cause problems. This is because with vectorized convection terms, transport is mostly only occurring in the direction of the velocity field. We use the term 'transportiveness' to describe an advection scheme's capability to represent directional transport by weighing the upstream (or upwind) coefficients more than downwind.

For the upwind scheme, we adopt a gradient discretization which is flow direction dependant. Take a 1D example from [1]. For a positive (rightward) flow, ϕ_e becomes ϕ_p and ϕ_w becomes ϕ_W, effectively shifting flux faces towards the upstream direction, yielding:

$$F_e\phi_P - F_w\phi_W = D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W) \tag{12}$$

If the flow is negative (leftward) however, then the process would be reversed, and ϕ_w that would become ϕ_p, while ϕ_e becomes ϕ_E:

$$F_e\phi_E - F_w\phi_P = D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W) \tag{13}$$

Incorporating the same process of grouping like terms will yield neighboring coefficients in the same way as it did for central difference, but this time there is flow direction dependence:

Condition	a_W, a_s	a_e, a_n
$F_w > 0, F_e > 0$	$D_w + F_w$	D_e
$F_w < 0, F_e < 0$	D_w	$D_e - F_e$
$F_s > 0, F_n > 0$	$D_s + F_s$	D_n
$F_s < 0, F_n < 0$	D_s	$D_n - F_n$

Table II: Coefficients for discretized transport equations considering flow direction.

When populating a dependency matrix, it's inefficient to use if-statements to constantly check flow direction in order to adjust the coefficient, which is why we can use a *max* function to check flow direction and adjust coefficients accordingly:

a_W	$D_w + \max(F_w, 0)$
a_E	$D_e + \max(0, -F_e)$
a_S	$D_s + \max(F_s, 0)$
a_N	$D_n + \max(0, -F_n)$

Table III: Coefficient calculations for 2D discretized transport equations.

II. RESULTS AND DISCUSSION

The simulation is setup with the following conditions: Fixed boundary conditions on all 4 walls of the equidistant domain. The temperature at the north and west walls are maintained at $T_{north}, T_{west} = 100$, while the south and east walls are maintained at $T_{south}, T_{east} = 0$. The dimensions of the domain are $L_x, L_y = 1$, and 2 different velocity fields will be analyzed for variable grid size, as well as variable thermal permeability Γ.

The system of linear equations is solved using a loose-general minimum residual solver (lgmres) with a tolerance of 10^{-10} and an incomplete LU preconditioner with the same tolerance as the solver. It was found that minimum residual methods functioned generally better than conjugate gradient methods for certain applications.

A. $\Gamma = 0$ Case

When the thermal permeability coefficient Γ governing diffusion in the fluid is set to 0, theoretically there should be no diffusion in the simulation's steady state. Due to the phenomenon of false diffusion, however, we expect some observable diffusion between the hot and cold sides of the domain. To test this hypothesis, a constant velocity field is initialized with values $(u_x, u_y) = (2, 2)$. This will result in a flow direction which runs parallel to the diagonal where the hot and cold domain's meet through the center of the grid.

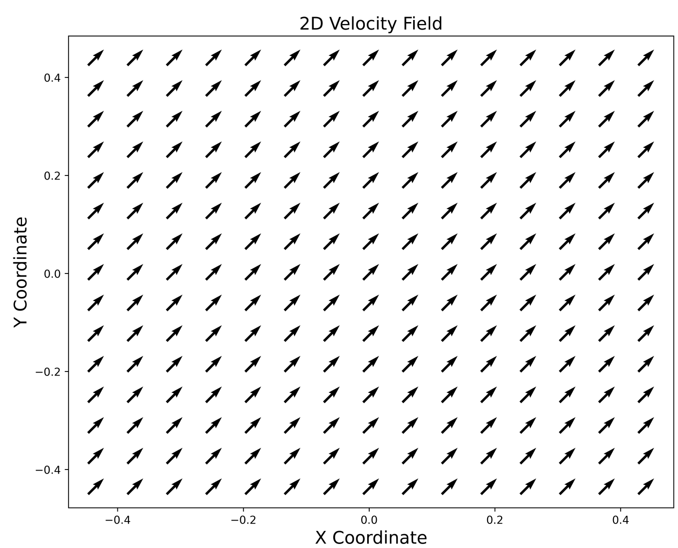

Figure 1: Velocity field with $(u, v) = (2, 2)$.

With this velocity field, both the upwind and central difference schemes are tested. The central difference scheme, however, does not converge to a solution.

This is due to the boundedness conditions of central difference. Take, for example, table II. When the flow is unidirectional ($F_e, F_w > 0$) and the convection coefficient is greater than diffusion $F_e > D_e, F_w > D_w$, than the east coefficient will be negative, which violates one of the requirements for boundedness. In order for the central dif-

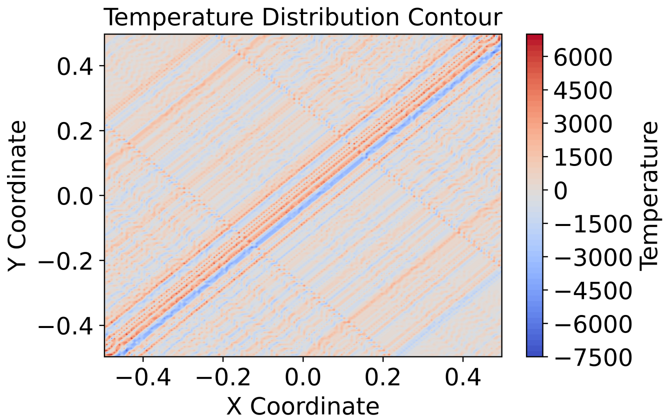

Figure 2: Temperature distribution for 160×160 grid nodes using the central difference scheme. 0 diffusion and constant velocity field $(u, v) = (2, 2)$.

ference scheme to be bounded, the convection and diffusion terms must abide by the inequality,

$$F_e/D_e = Pe_e < 2, \tag{14}$$

where Pe_e is known as the Peclet number. Maintaining that the Peclet number be lower than 2 is required so that the solution for the central difference scheme remain stable and bounded. For a 0-diffusion scenario, the Peclet number will always be infinite, and therefore the solution is completely unbounded, and will not converge.

Utilizing the upwind scheme, however, solves this boundedness problem by virtue of changing the coefficient calculation depending on flow direction. The coefficients for the upwind scheme will always be positive, leading to a solution which converges even without diffusion terms.

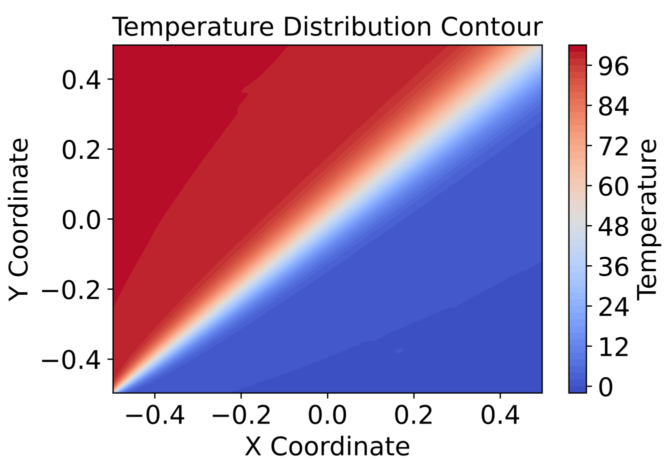

Figure 3: Temperature distribution for 160×160 grid nodes using upwind scheme. 0 diffusion and constant velocity field $(u, v) = (2, 2)$. The solution did not converge.

As seen in figure 3, the upwind scheme converges to a predictable solution; 2 domains, one hot and one cold, separated by a smooth but steep transition through it's center. Of note is how this transition broadens as it approaches the north-east corner of grid. This is a property of the false diffusion induced by the discretization of convection terms in the finite volume method.

False diffusion is a non-physical numerical artifact. As the flow moves from the southwest to the northeast, it continuously sweeps higher temperature values from the boundaries into the domain. Due to the upwind bias, the high temperatures at the northwest and southwest are advected inwards and diagonally, creating a broad transition zone that widens as it reaches the northeast corner.

We can plot the temperature across the north-west to south-east diagonal to see how this false diffusion is affected by the grid size

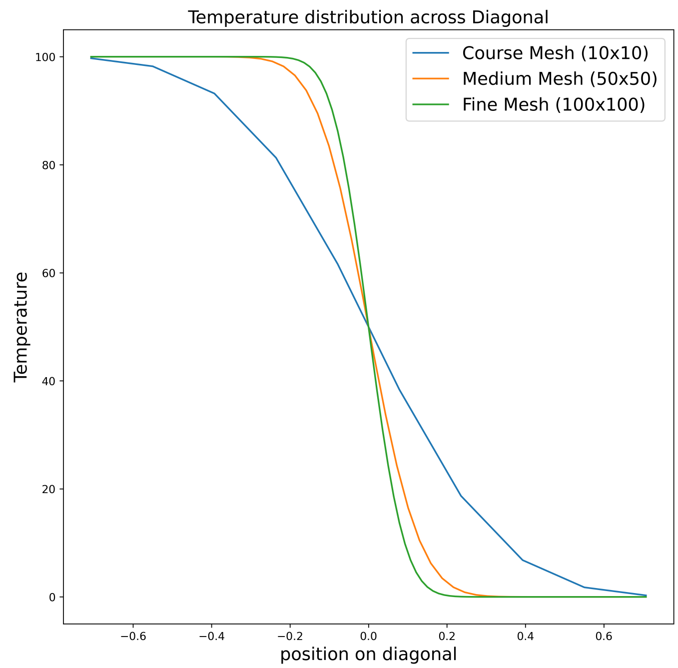

Figure 4: 0 diffusion temperature distribution across north-west to south-east diagonal using upwind for grid sizes 10×10, 50×50, and 100×100.

These results match reference [1]'s figure 5.15 well.

B. $\Gamma = 5$ Case

Now we may explore the results while using a thermal permeability coefficient of $\Gamma = 5$, bringing diffusion into the domain. The same steps as before are undertaken with a velocity field of $(u, v) = (2, 2)$, but this time the result using central difference can be analyzed as the Peclet number will be lower.

Analyzing figures 5 and 6, we notice very little difference between the advection schemes when diffusion is at play. That said, the difference between the cases without diffusion and with diffusion are stark. With diffusion enabled, there's a far broader, smoother blending between the temperatures at each end of the domain, and the false diffusion induced by the convective terms is negligible by comparison.

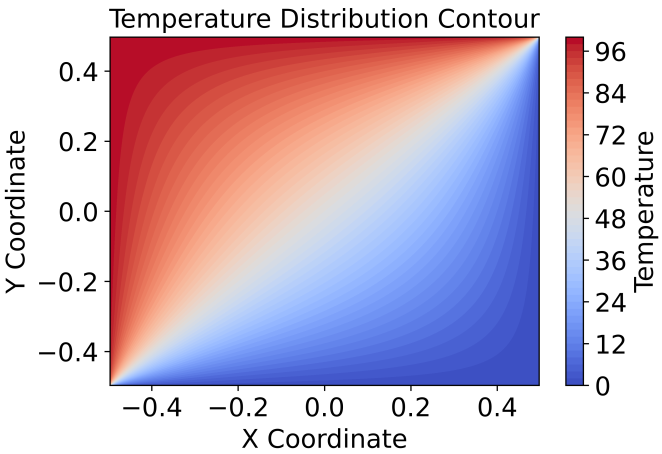

Figure 5: $\Gamma = 5$ temperature distribution using the upwind scheme with 160×160 grid nodes.

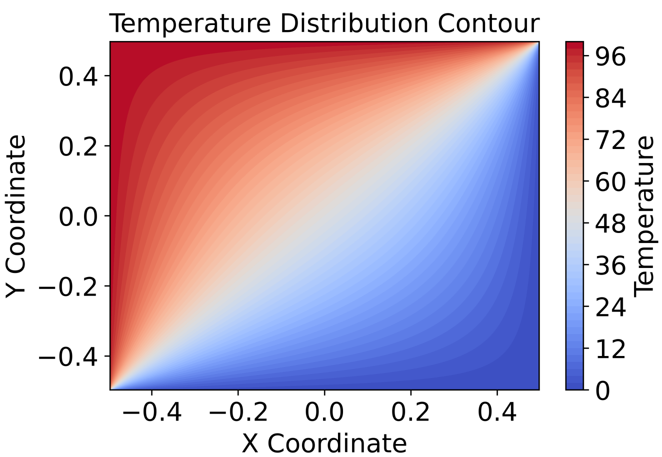

Figure 6: $\Gamma = 5$ temperature distribution using the central difference scheme with 160×160 grid nodes.

The numerical stability issue has also been fixed, as utilizing diffusion coefficients brings back the boundedness of the central difference method, no longer having neighboring terms in the coefficient matrix be negative in the east or north directions.

C. Circular Velocity Pattern

Having explored the effectiveness of upwind vs central difference for constant velocity streams with and without diffusion, we'll not examine the performance of either scheme on a variable velocity field. Specifically we will create a circular velocity function which maps the following equations according to grid position:

$$r = \sqrt{x^2 + y^2} \tag{15}$$
$$\theta = \arctan 2(y, x) \tag{16}$$
$$u = -r \sin\theta \tag{17}$$
$$v = r \cos\theta \tag{18}$$

This velocity distribution creates a circular pattern, with increased speed proportional to the distance from the center of the grid.

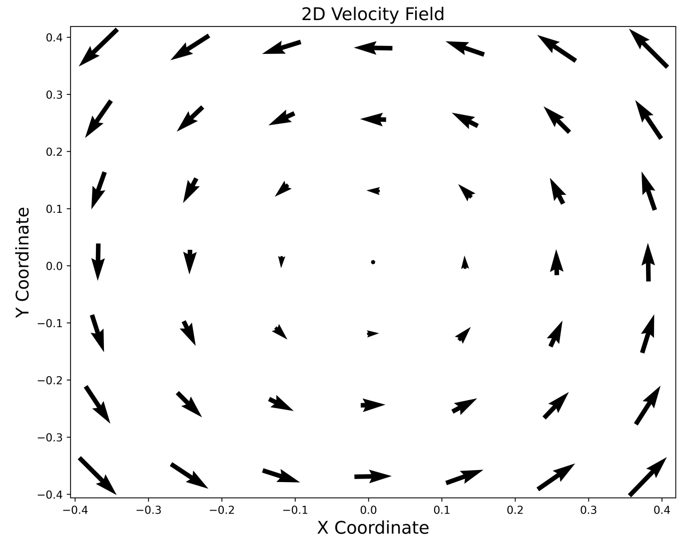

Figure 7: Circular velocity field implementing equations (17) and (18).

When applying either the central difference or upwind schemes to solve this velocity field, and the permeability of $\Gamma = 5$, results appear to be very similar, just as they were when using a constant velocity field.

Visually, there is no difference between either 320×320 node plot, and the same can be said for courser simulations using 80×80 or 160×160, but numerically there are characteristic changes which can be observed by calculating the order of convergence using either scheme.

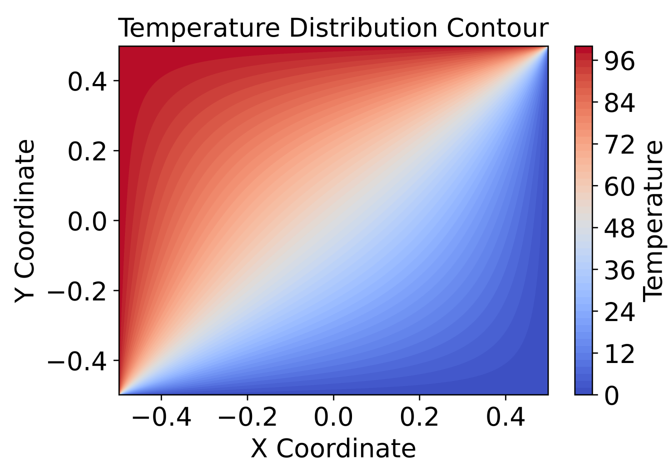

Figure 8: Temperature distribution using the upwind scheme on a 320×320 node grid and a circular velocity distribution.

The order of convergence for an equidistant mesh can be calculated using equation 19;

$$0 \approx \frac{\log(|e_c|/|e_f|)}{\log(h_c/h_f)}, \tag{19}$$

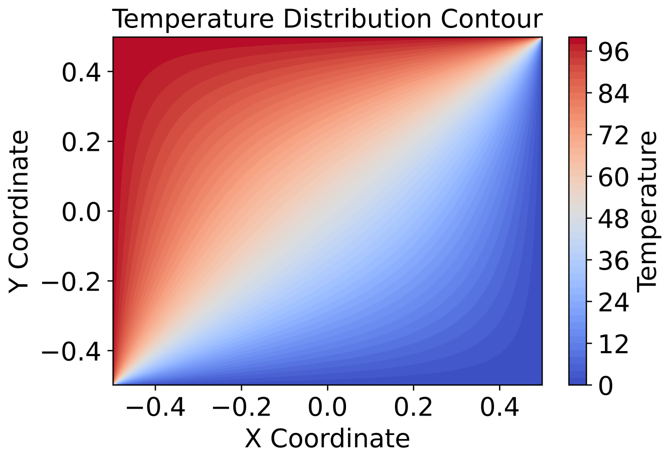

Figure 9: Temperature distribution using the central difference scheme on a 320×320 node grid and a circular velocity distribution.

Where e_c and e_f are the error norms for a course and a medium mesh (with respect to a fine mesh) and h_c and h_f are the spacing of the course and medium mesh respectively. The error norms can be calculated according to equation (20);

$$|e| = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\phi_f - \phi_c)^2}, \tag{20}$$

where ϕ_f is the temperature distribution for the finest grid (estimating the analytical solution) at 320×320, and ϕ_c is either the course or the medium temperature distribution. Since the meshes don't have the same sampling points, an interpolation and extrapolation step is performed so that the course and medium meshes are sampled at the same locations as the fine mesh.

Mesh Size	**Upwind Error**	**Central Diff Error**
320x320 (fine)	Analytical Solution	Analytical Solution
160x160 (medium)	0.305256	0.305736
80x80 (coarse)	0.410890	0.411133
Order of Convergence:	0.4287326	0.427321

Table IV: Errors and Order of Convergence for Different Mesh Sizes

As can be seen in table IV, the order of convergence using the upwind scheme is slightly higher (about 0.001) than that of the central difference scheme. This difference is fairly small, nearly negligible in fact, but does indicate that using the upwind scheme may not require as high a grid resolution in order to achieve good results when compared to central difference.

III. CONCLUSIONS

In this assignment, we added convection to the existing 2D diffusion problem solved in the previous assignment. By implementing a first-order linear upwind scheme rather than only central difference, the solver was able to converge even without diffusion. Apart from this improved boundedness and transportiveness achieved through upwind, the results with diffusion between the 2 advection schemes were very similar, no matter the velocity field examined. The small differences in solutions observed between schemes can be seen through order of convergence calculations using 3 grid sizes. The upwind scheme was found to have a slightly higher convergence rate when compared to central difference.

[1] H. K. Versteeg and W. Malalasekera, *An Introduction to Computational Fluid Dynamics: THE FINITE VOLUME METHOD*, 2nd ed. Pearson Education Limited, 2007.