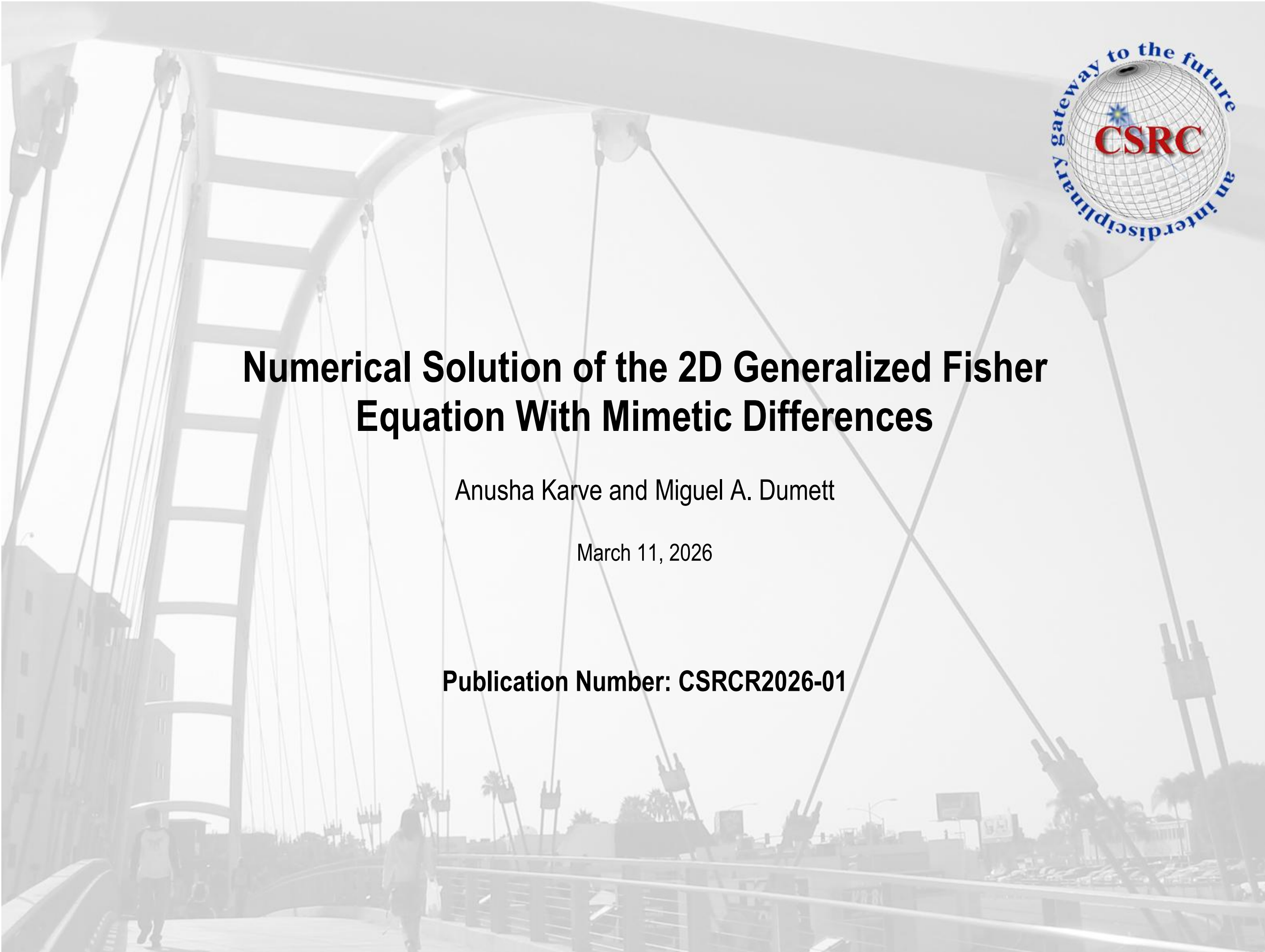

Numerical Solution of the 2D Generalized Fisher Equation With Mimetic Differences

Anusha Karve and Miguel A. Dumett

March 11, 2026

Publication Number: CSRCR2026-01

Computational Science & Engineering Faculty and Students Research Articles

Database Powered by the Computational Science Research Center Computing Group & Visualization Lab

COMPUTATIONAL SCIENCE & ENGINEERING

San Diego State University

Computational Science Research Center
College of Sciences
5500 Campanile Drive
San Diego, CA 92182-1245
(619) 594-3430

Numerical Solution of the 2D Generalized Fisher Equation With Mimetic Differences

Anusha Karve* and Miguel A. Dumett †‡

March 11, 2026

Abstract

This report documents the numerical solution of the two-dimensional generalized Fisher equation using mimetic difference operators from the MOLE library in MATLAB. An exact manufactured solution is used to validate the implementation. Spatial derivatives are discretized with MOLE's second-order mimetic Laplacian (`lap2D` with accuracy order $k = 2$), and time integration is carried out using the method of lines with `ode15s`. Numerical experiments confirm second-order spatial convergence in both L_∞ and L_2 norms.

1 Introduction

Reaction–diffusion equations arise in a wide range of scientific and engineering applications, including population dynamics, chemical kinetics, biological pattern formation, and fluid transport phenomena. These models describe the combined effects of spatial diffusion and nonlinear reaction mechanisms. Among the classical examples of such models is the Fisher equation [1], originally introduced to describe the propagation of advantageous genes in a population. The equation combines a diffusion term with a logistic growth term, producing nonlinear wave-like behavior.

In this document, we consider the two-dimensional generalized Fisher equation posed in a square domain. The governing equation consists of a Laplacian

*Computational Science Master Program at San Diego State University (akarve0363@sdsu.edu).

†Editor: Jose E. Castillo

‡Computational Science Research Center at San Diego State University (mdumett@sdsu.edu).

operator representing diffusion and a nonlinear reaction term of logistic type. Due to the presence of nonlinearities and spatial derivatives in multiple dimensions, analytical solutions are generally not available, and reliable numerical methods are required.

Mimetic difference methods [2, 3] aim to preserve important structural properties of the underlying continuous differential operators at the discrete level. In particular, mimetic operators are constructed so that discrete analogs satisfy key integral identities of vector calculus. These properties can improve robustness of the discretization, especially near boundaries where standard finite difference stencils require ad hoc modifications.

The objective of this project is to implement a mimetic discretization of the two-dimensional generalized Fisher equation using the MOLE library in MATLAB and to verify the numerical accuracy of the method. A manufactured exact solution is introduced, and the forcing term is constructed accordingly to allow a direct comparison between the numerical and analytical solutions. The performance of the method is assessed through error norms and a convergence study over multiple grid resolutions.

2 Mathematical Formulation

We consider the two-dimensional generalized Fisher equation defined on a square domain $\Omega = [0, \pi] \times [0, \pi]$ over the time interval $t \in [0, 1]$. The governing equation is given by

$$u_t = \Delta u + u(1 - u) + f(x, y, t), \quad (x, y) \in \Omega,\ t > 0, \tag{1}$$

where $\Delta u = u_{xx} + u_{yy}$ denotes the two-dimensional Laplacian operator, and $u(1 - u)$ represents a nonlinear logistic reaction term.

The problem is equipped with homogeneous Dirichlet boundary conditions

$$u(x, y, t) = 0, \quad (x, y) \in \partial\Omega,\ t \geq 0, \tag{2}$$

and the initial condition

$$u(x, y, 0) = 0, \quad (x, y) \in \Omega. \tag{3}$$

To verify the numerical implementation, a manufactured exact solution is introduced:

$$u_{\text{exact}}(x, y, t) = \sin(x) \sin(y) \sin(t). \tag{4}$$

It is straightforward to verify that the manufactured solution satisfies the homogeneous Dirichlet boundary conditions since

$$u_{\text{exact}}(0,y,t) = u_{\text{exact}}(\pi,y,t) = u_{\text{exact}}(x,0,t) = u_{\text{exact}}(x,\pi,t) = 0,$$

due to the sine factors. In addition, $u_{\text{exact}}(x,y,0) = 0$, so the initial condition is satisfied.

A short derivation of the forcing term is included for completeness. First compute

$$u_t = \sin(x)\sin(y)\cos(t), \quad u_{xx} = -\sin(x)\sin(y)\sin(t), \quad u_{yy} = -\sin(x)\sin(y)\sin(t),$$

so that

$$\Delta u = u_{xx} + u_{yy} = -2\sin(x)\sin(y)\sin(t).$$

Substituting into $u_t = \Delta u + u(1-u) + f$ gives

$$f = u_t - \Delta u - u(1-u).$$

With $u = \sin(x)\sin(y)\sin(t)$,

$$u_t - \Delta u = \sin(x)\sin(y)\big(\cos(t) + 2\sin(t)\big),$$

and

$$u(1-u) = \sin(x)\sin(y)\sin(t) - \sin^2(x)\sin^2(y)\sin^2(t).$$

Therefore,

$$f(x,y,t) = \sin(x)\sin(y)\left[\cos(t) + \sin(t) + \sin(x)\sin(y)\sin^2(t)\right], \tag{5}$$

which matches the forcing used in the implementation.

Since the manufactured solution is smooth (infinitely differentiable) on $\Omega \times [0,1]$, the regularity assumptions typically required for second-order spatial convergence are satisfied.

3 Numerical Method

3.1 Spatial Discretization with MOLE Mimetic Operators

The domain $\Omega = [0,\pi] \times [0,\pi]$ is discretized using a uniform grid with spacing h in both spatial directions:

$$x_i = ih, \quad y_j = jh, \quad i,j = 0,1,\ldots,N, \qquad h = \frac{\pi}{N}.$$

Spatial derivatives are approximated using the MOLE library [5, 4]. In particular, the discrete Laplacian is constructed via the MOLE routine

```
lap2D(k,nx,dx,ny,dy),
```

with accuracy order $k = 2$. The resulting operator is a sparse matrix that provides a second-order mimetic approximation of the continuous Laplacian.
Remark (mimetic vs. classical finite differences). Although a second-order Laplacian has a familiar five-point structure in the interior, MOLE's mimetic operators are constructed to satisfy discrete analogs of key continuous identities (e.g., symmetry and discrete integration-by-parts/Green-type relations) and to provide a principled treatment near boundaries. In particular, the near-boundary rows are not obtained by a naive stencil truncation; they are designed to preserve mimetic properties while maintaining the target accuracy order ($k = 2$ here). In this project, these properties are validated through a manufactured-solution convergence study rather than being proven analytically.

A representative mimetic property is the discrete divergence theorem (a Green-type identity) satisfied by the mimetic gradient G and divergence D operators under the discrete weighted inner products Q and P:

$$h\langle DV, F\rangle_Q + h\langle V, GF\rangle_P = (-1, 0, \ldots, 0, 1). \tag{6}$$

This high-order approximation represents the discrete analog of the continuous Gauss's divergence theorem and is a fundamental structural property preserved by the mimetic discretization.

For homogeneous Dirichlet conditions, the discrete Laplacian is expected to be negative semidefinite, reflecting the dissipative nature of diffusion. This implies that the eigenvalues of L are real and non-positive, which is consistent with decay of modes under the heat operator and is one reason implicit time integration is appropriate for refined grids.

3.2 Boundary Treatment in the Implementation

Homogeneous Dirichlet boundary conditions are imposed on all four boundaries:

$$u(x, y, t) = 0, \quad (x, y) \in \partial\Omega.$$

In the implementation, the full grid (including boundary nodes) is retained in the state vector $U(t)$. Boundary conditions are enforced using MOLE's scalar boundary condition utilities (e.g., `addScalarBC2D`) and by setting the time derivatives at boundary nodes to zero in the ODE right-hand side. This ensures boundary values remain fixed throughout time integration while keeping the system size consistent with the full grid representation.

3.3 Semi-Discrete System (Method of Lines)

Let $U(t) \in \mathbb{R}^{(N+1)^2}$ denote the lexicographically ordered vector of grid values on the full grid. After spatial discretization, the PDE becomes a nonlinear ODE system:

$$\frac{dU}{dt} = LU + U \odot (1 - U) + F(t), \tag{7}$$

where $\odot$ denotes elementwise multiplication and $F(t)$ is obtained by evaluating the manufactured forcing function at each grid point and time.

Boundary entries of $\frac{dU}{dt}$ are set to zero so that boundary values remain at 0 for all t.

3.4 Time Integration and Solver Settings

The semi-discrete nonlinear system is integrated over $t \in [0, 1]$ using MATLAB's stiff ODE solver `ode15s`. This choice is motivated by the diffusive term, whose stiffness scales like h^{-2} as the grid is refined. The solver tolerances were set to

$$\texttt{RelTol} = 10^{-8}, \qquad \texttt{AbsTol} = 10^{-10},$$

so that for the tested grid resolutions the spatial discretization error dominates the overall error.

The Jacobian of the semi-discrete system has the structure

$$J(U) = L + \text{diag}(1 - 2U),$$

where L is the sparse mimetic Laplacian matrix and

$$\text{diag}(1 - 2U)$$

denotes the diagonal matrix whose entries are $1 - 2U_{i,j}$ at each grid node. Thus the Jacobian consists of a sparse symmetric diffusion operator plus a diagonal nonlinear reaction contribution.

Because L has a five-point-type sparsity pattern and the reaction term contributes only diagonal entries, the overall Jacobian remains sparse. This structure is favorable for implicit solvers such as `ode15s`, which internally exploit sparse linear algebra when approximating or factorizing the Jacobian.

Implementation Details (Reproducibility)

The MATLAB implementation uses MOLE's `lap2D` with accuracy order $k = 2$. The grid sizes tested were $N = 20, 40, 80$ with $h = \pi/N$. Time integration was performed using `ode15s` on $[0, 1]$ with `RelTol`$= 10^{-8}$ and `AbsTol`$= 10^{-10}$. The forcing term was evaluated pointwise using the manufactured solution so that the exact solution satisfies the PDE.

4 Algorithm

The computational procedure for solving the two-dimensional generalized Fisher equation can be summarized as follows:

1. **Grid Generation:** Choose N and construct a uniform grid on $\Omega = [0, \pi] \times [0, \pi]$ with spacing $h = \pi/N$.
2. **Mimetic Laplacian Construction:** Use MOLE to build the second-order mimetic Laplacian matrix L via `lap2D(k,nx,dx,ny,dy)` with $k = 2$.
3. **Boundary Condition Setup:** Configure homogeneous Dirichlet boundary conditions using MOLE boundary utilities (e.g., `addScalarBC2D`) and enforce them by setting boundary time derivatives to zero.
4. **Initialization:** Set $U(0)$ from the initial condition $u(x, y, 0) = 0$.
5. **Time Integration:** Solve the nonlinear ODE system with `ode15s` over $t \in [0, 1]$.
6. **Error Evaluation:** Compute L_∞ and L_2 error norms at $t = 1$ by comparing the numerical solution to $u_{\text{exact}}(x, y, 1)$.
7. **Convergence Study:** Repeat for $N = 20, 40, 80$ and then compute observed convergence rates.

5 Numerical Results

Numerical experiments were performed for three spatial resolutions, namely $N = 20$, 40, and 80 grid cells in each spatial direction. The solution was computed up to final time $t = 1$, and discrete error norms were evaluated by comparison with the manufactured exact solution.

5.1 Error Analysis

The discrete L_∞ and L_2 norms are defined by

$$\|e\|_\infty = \max_{i,j} |u_{i,j}^{\text{num}} - u_{i,j}^{\text{exact}}|, \tag{8}$$

$$\|e\|_2 = \left(h^2 \sum_{i,j} |u_{i,j}^{\text{num}} - u_{i,j}^{\text{exact}}|^2 \right)^{1/2}. \tag{9}$$

The computed errors and observed convergence rates are summarized in Table 1.

N	L_∞ Error	Rate	L_2 Error	Rate
20	9.305358×10^{-4}	–	1.405638×10^{-3}	–
40	2.427941×10^{-4}	1.94	3.804873×10^{-4}	1.88
80	6.184170×10^{-5}	1.97	9.897467×10^{-5}	1.94

Table 1: Discrete error norms and observed convergence rates.

5.2 Convergence Study

As the grid resolution is doubled, the error decreases approximately by a factor of four. The observed convergence rates are close to two, indicating second-order spatial accuracy.

Figure 1 shows the log–log convergence plot of the error norms. Both L_∞ and L_2 errors exhibit approximately linear behavior with slope close to two, confirming $O(h^2)$ convergence.

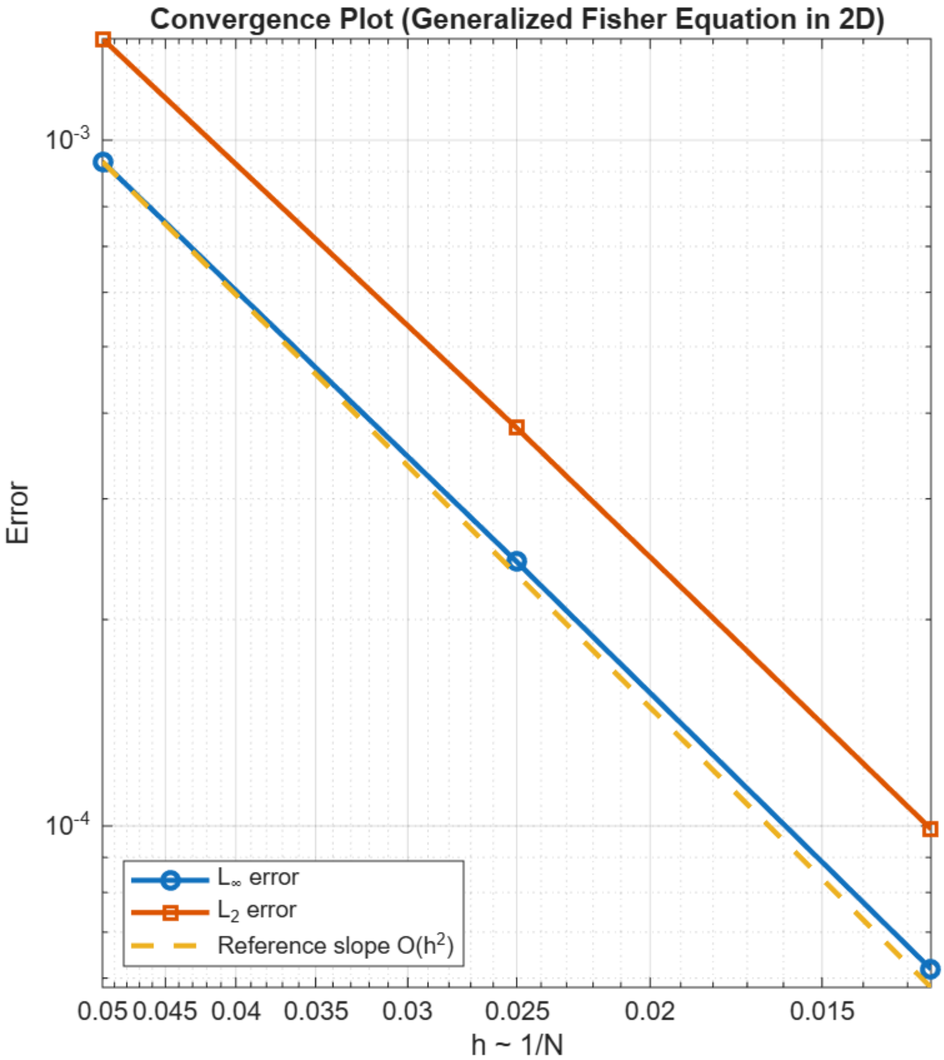

Figure 1: Log–log convergence plot of the L_∞ and L_2 errors versus grid spacing $h \sim 1/N$.

Figure 2 shows the numerical solution at $t = 1$, while Figure 3 shows the manufactured exact solution. The two surfaces match well. The pointwise error surface is given in Figure 4. This further confirms that the numerical approximation is highly accurate. From the error surface, it can be noted that the maximum error occurs in the interior region where the magnitude of the solution is the highest. The errors are still low at the boundaries because of the Dirichlet conditions.

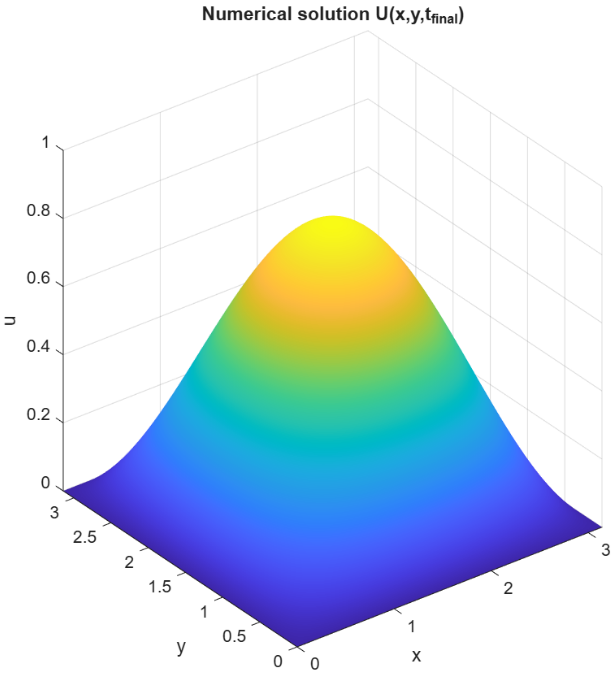

Figure 2: Numerical solution $U(x, y, t)$ at $t = 1$ for $N = 80$.

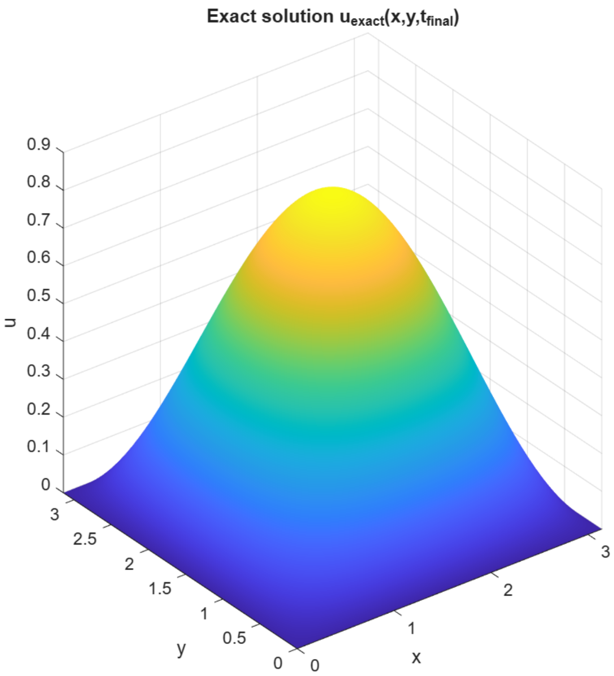

Figure 3: Manufactured exact solution $u_{\text{exact}}(x, y, t)$ at $t = 1$.

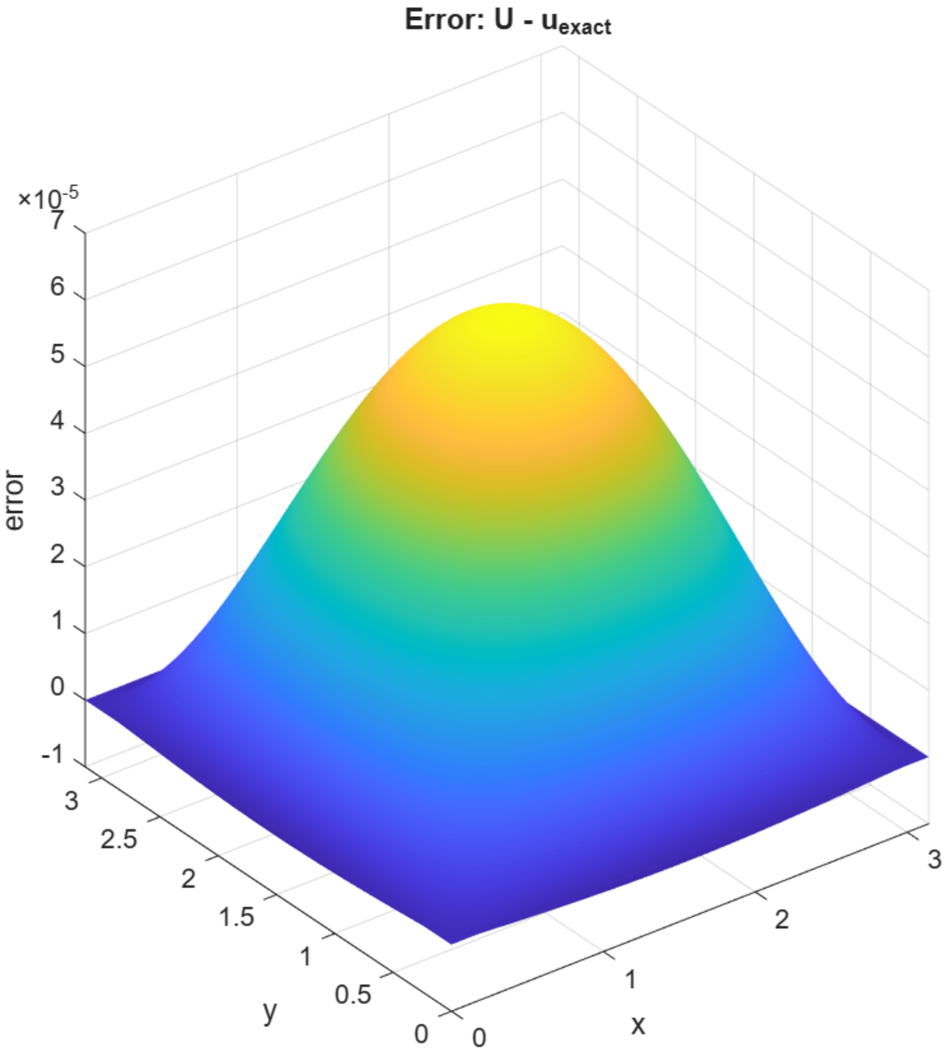

Figure 4: Pointwise error $U(x, y, 1) - u_{\text{exact}}(x, y, 1)$ for $N = 80$.

6 Discussion

The numerical results show that the mimetic difference method implemented with MOLE yields a second-order accurate approximation for the two-dimensional generalized Fisher equation. These rates are consistent with the expected behavior of the second-order mimetic Laplacian used in `lap2D`.

The smooth decay of the L_∞ and L_2 error norms as the grid is refined is a further indicator of consistency and *empirical* stability for the tested resolutions and solver settings.

The Kronecker-product structure underlying MOLE's two-dimensional operator construction preserves separability of the continuous Laplacian and produces a sparse and efficient discrete operator.

Homogeneous Dirichlet boundary conditions were enforced directly at the discrete level by fixing boundary values and setting boundary derivatives to zero in the ODE system. The absence of spurious oscillations and the close agreement between numerical and exact surfaces indicate correct boundary enforcement.

Using `ode15s` provides a robust implicit time integrator for the stiff semi-discrete system; in our experiments it produced stable time stepping and allowed the spatial convergence behavior to be observed without time-step restrictions dominating the error.

Computational Considerations and Extensions

For a grid with N cells per spatial direction, the number of degrees of freedom is $(N+1)^2$, and the discrete Laplacian matrix has a sparse five-point-type coupling structure, so the number of nonzero entries scales linearly with the number of grid nodes. Consequently, the overall computational cost grows rapidly with refinement and motivates the use of sparse linear algebra in implicit time integration.

The diffusive term introduces stiffness that scales like h^{-2}, which explains why explicit time stepping would require very small time steps for stability on refined grids. The implicit `ode15s` solver is therefore appropriate for efficiently handling stiffness while maintaining accuracy.

A natural extension of this work is to test higher-order mimetic operators available in MOLE (larger k values) and compare accuracy versus cost. Another extension is to compare the mimetic Laplacian to the classical five-point finite difference Laplacian on the same manufactured solution and assess differences in boundary behavior and robustness.

7 Conclusion

In this report, the two-dimensional generalized Fisher equation was solved numerically using mimetic difference operators implemented through the MOLE library in MATLAB. The spatial discretization was carried out using MOLE's second-order mimetic Laplacian operator `lap2D` with $k = 2$, and homogeneous Dirichlet boundary conditions were enforced on the full grid by fixing boundary values throughout the time integration.

The resulting semi-discrete nonlinear system was integrated using the method of lines with MATLAB's `ode15s` solver. An exact manufactured solution was introduced to verify the correctness, and the forcing term was derived accordingly.

A convergence study over $N = 20, 40, 80$ demonstrated approximately second-order spatial accuracy in both L_∞ and L_2 norms. These results validate the implementation and show that the mimetic framework provides a reliable and systematically convergent approach for nonlinear reaction–diffusion problems in two dimensions. The results demonstrate that the mimetic discretization preserves essential structural properties of the continuous operators while achieving the expected theoretical accuracy.

References

[1] R. A. Fisher, *The wave of advance of advantageous genes*, Annals of Eugenics, 7 (1937), pp. 355–369.

[2] J. E. Castillo and R. Grone, *A matrix analysis approach to higher-order approximations for divergence and gradient operators*, SIAM Journal on Matrix Analysis and Applications, 25 (2003), pp. 128–142.

[3] J. Corbino and J. E. Castillo, *High-order mimetic difference operators on staggered grids*, Computational Science Research Center Technical Report, San Diego State University.

[4] J. Corbino, M. Dumett, and J. E. Castillo, *MOLE: Mimetic Operators Library Enhanced*, Journal of Open Source Software, 9(99), 6288, `https://joss.theoj.org/papers/10.21105/joss.06288`

[5] MOLE Library (Mimetic Operators Library Enhanced), Computational Science Research Center, San Diego State University. Available at: `https://github.com/csrc-sdsu/mole`