Results in Applied Mathematics 27 (2025) 100631

Contents lists available at ScienceDirect

Results in Applied Mathematics

journal homepage: www.elsevier.com/locate/results-in-applied-mathematics

Turing patterns across geometries: A proven DSC-ETDRK4 solver from plane to sphere

Kolade M. Owolabi [a,b,c,d,*], Edson Pindza [e], Eben Maré [a]

[a] Department of Mathematics and Applied Mathematics, University of Pretoria, Pretoria 002, South Africa
[b] Department of Mathematical Sciences, Federal University of Technology Akure, PMB 704, Akure, Ondo State, Nigeria
[c] Department of Mathematics and Applied Mathematics, School of Science and Technology, Sefako Makgatho Health Sciences University, Ga-Rankuwa 0208, South Africa
[d] Institute for Groundwater Studies, Faculty of Natural and Agricultural Sciences University of the Free State, Bloemfontein 9300, South Africa
[e] Department of Decision Sciences, College of Economic and Management Sciences, University of South Africa (UNISA), Pretoria 003, South Africa

ARTICLE INFO

MSC:
65M70
35K57
37M05
92C15

Keywords:
DSC-ETDRK
Reaction-diffusion systems
Gray–Scott model
Brusselator model
Simulation experiments
Turing patterns

ABSTRACT

This paper presents a unified and robust numerical framework that combines the Discrete Singular Convolution (DSC) method for spatial discretization with the Exponential Time Differencing Runge–Kutta (ETDRK4) scheme for temporal integration to solve reaction–diffusion systems. Specifically, we investigate the formation of Turing patterns – such as spots, stripes, and mixed structures – in classical models including the Gray–Scott, Brusselator, and Barrio–Varea–Aragón–Maini (BVAM) systems. The DSC method, employing the regularized Shannon's delta kernel, delivers spectral-like accuracy in computing spatial derivatives on both regular and curved geometries. Coupled with the fourth-order ETDRK method, this approach enables efficient and stable time integration over long simulations. Importantly, we rigorously establish the necessary theoretical results – including convergence, stability, and consistency theorems, along with their proofs – for the combined DSC-ETDRK4 method when applied to both planar and curved surfaces. We demonstrate the capability of the proposed method to accurately reproduce and analyze complex spatiotemporal patterns on a variety of surfaces, including the plane, sphere, torus, and bumpy geometries. Numerical experiments confirm the method's versatility, high accuracy, and computational efficiency, making it a powerful tool for the study of pattern formation in reaction–diffusion systems on diverse geometries.

1. Introduction

Reaction–diffusion systems are a class of partial differential equations that model the spatiotemporal evolution of chemical or biological species. These systems have been the subject of extensive study due to their ability to generate a wide variety of complex patterns and behaviors, making them valuable in understanding phenomena across multiple scientific disciplines [1,2].

The investigation of pattern formation within reaction–diffusion systems has emerged as a captivating and influential area of research across multiple scientific disciplines, including biology, chemistry, and physics. This interest was initially sparked by the groundbreaking work of mathematician Alan Turing in 1952, who introduced a theoretical framework to explain how complex spatial patterns can arise naturally during the early stages of biological development, particularly in embryos [3]. Turing's

* Corresponding author at: Department of Mathematics and Applied Mathematics, University of Pretoria, Pretoria 002, South Africa.
E-mail address: kmowolabi@futa.edu.ng (K.M. Owolabi).

https://doi.org/10.1016/j.rinam.2025.100631
Received 22 May 2025; Received in revised form 16 July 2025; Accepted 15 August 2025
Available online 25 August 2025

model centered around the interaction between two chemical species, commonly referred to as an activator and an inhibitor, whose differing diffusion rates could destabilize a uniform state, leading to the emergence of stable, spatially organized structures. This mechanism, now known as Turing instability, provided one of the first mathematical explanations for how simple chemical interactions could give rise to intricate and self-organized patterns in nature.

Among the various reaction–diffusion models, the Gray-Scott system has attracted significant attention due to its rich pattern-forming behavior. Pearson [4] studied the Gray-Scott reaction–diffusion model in two dimensions numerically and reported some complex structures in biological patterns. The system describes the following autocatalytic reaction scheme:

$$\begin{aligned} \mathcal{U} + 2\mathcal{V} &\xrightarrow{k_1} 3\mathcal{V} \\ \mathcal{V} &\xrightarrow{k_2} \mathcal{P} \end{aligned} \tag{1.1}$$

where $\mathcal{U}$ is as a reactant, $\mathcal{V}$ stands for the autocatalyst, and $\mathcal{P}$ denotes the inert product.

The dimensionless form of the Gray-Scott model can be expressed as:

$$\begin{aligned} \frac{\partial u}{\partial t} &= F(1-u) - uv^2 + \mu_1 \nabla^2 u \\ \frac{\partial v}{\partial t} &= uv^2 - (F+k)v + \mu_2 \nabla^2 v. \end{aligned} \tag{1.2}$$

Here u and v are the concentrations of the two chemical species U and V, respectively. $\mu_1 \geq 0$ and $\mu_2 \geq 0$ are the diffusion coefficients, F denotes dimensionless flow rate, and k is the dimensionless rate constant for the second reaction [5].

Numerically addressing highly nonlinear reaction–diffusion systems presents significant challenges due to their complex nature. While the diffusion component is typically linear, the reaction terms often exhibit strong nonlinearity, making the overall system difficult to solve accurately and efficiently [6]. These models are commonly applied to simulate pattern formation on complex, high-dimensional, and irregular geometrical domains, particularly in biological contexts where the shape and size of the domain, such as embryonic structures, play a pivotal role. As a result, the development of robust and efficient numerical methods is essential for analyzing and understanding the dynamic behavior of solutions in reaction–diffusion systems related to biological development [7].

In recent years, the Discrete Singular Convolution (DSC) approach has gained significant attention as a versatile computational technique for tackling nonlinear partial differential equations encountered across a wide range of scientific and engineering problems. Originally introduced by Wei in [8] as a numerical framework for implementing singular convolution operations, the DSC method has since evolved into a powerful tool with broad applicability. It has been successfully employed in diverse areas such as the analysis of nonlinear dynamical systems, structural mechanics, fluid dynamics (both compressible and incompressible), electromagnetic wave modeling, and digital image processing [5].

The primary objective of this study is to establish a systematic and unified computational framework for simulating multi-dimensional reaction–diffusion systems on a variety of geometric domains. Specifically, we focus on three prototypical models known for their rich pattern-forming behavior, the Gray-Scott model, the Brusselator model, and the Barrio-Varea-Aragón-Maini (BVAM) model. To achieve this, we employ the Discrete Singular Convolution (DSC) method as the core numerical tool for spatial discretization.

A central feature of the proposed framework lies in its capacity to accurately approximate second-order spatial derivatives that arise in the diffusion terms, while also effectively managing the nonlinearities introduced by the reaction kinetics. These components are critical when simulating complex Turing patterns, such as spots, stripes, and mixed formations—on both regular and curved surfaces. By addressing the computational challenges posed by multi-dimensionality and geometrical complexity, our goal is to construct a reliable and high-resolution DSC-based numerical scheme that not only ensures stability and efficiency, but also captures the intricate spatiotemporal dynamics characteristic of reaction–diffusion systems across diverse applications.

To complement the spatial discretization provided by the DSC method, we employ Exponential Time Differencing Runge–Kutta (ETDRK) methods for the temporal integration. ETDRK methods have shown promise in efficiently integrating stiff systems of ordinary differential equations resulting from the spatial discretization of PDEs [9]. The combination of DSC for spatial discretization and ETDRK for temporal integration offers the potential for a highly accurate and efficient numerical scheme for reaction–diffusion systems.

In this paper, we focus on developing an explicit mixed modal DSC method based on multi-dimensional structured meshes for the Gray-Scott reaction–diffusion system in Turing pattern formation. We also propose a novel reaction term treatment, demonstrating an intrinsic feature of the new DSC scheme and preventing erroneous solutions due to nonlinear reaction terms.

The Discrete Singular Convolution (DSC) method offers a compelling alternative to conventional numerical schemes such as finite difference and global spectral methods for the solution of partial differential equations. A key advantage of the DSC approach lies in its ability to attain spectral-like accuracy while employing compact, localized stencils. This localization enhances computational efficiency and makes the method particularly well-suited for problems involving complex boundary conditions, sharp gradients, or non-uniform grids. Unlike global spectral methods that require the use of global basis functions – often leading to challenges in handling boundary effects and irregular domains – the DSC method maintains high accuracy without sacrificing flexibility. Furthermore, the DSC framework allows for systematic kernel regularization and resolution control, enabling effective handling of both smooth and non-smooth solution features. These properties position the DSC method as a robust and high-fidelity numerical tool for modeling and simulation across a range of scientific and engineering problems.

The remainder of this paper is organized as follows. In Section 2, we present various mathematical formulation of the models and briefly discuss their general stability conditions. Section 3 describes the DSC method in detail, including its mathematical

foundations, kernel functions, and spatial discretization. Section 4 introduces the ETDRK methods for temporal integration. In Section 5, we present our combined DSC-ETDRK approach for solving reaction–diffusion systems. Numerical experiments and results are presented in Section 6, followed by a discussion in Section 7. Finally, we provide concluding remarks and directions for future work in Section 8.

2. Mathematical formulation

2.1. General form of reaction–diffusion equations

Reaction–diffusion systems are a class of partial differential equations that describe the spatial and temporal evolution of chemical or biological species. In general, these systems can be written in the form:

$$\frac{\partial U}{\partial t} = D\nabla^2 U + S(U), \quad x \in \Omega \times [0,T], \tag{2.3}$$

where $U \in \mathbb{R}^n$ illustrates the concentrations of a group of bio-chemical-species components, n denotes the number of chemical species, $D \in \mathbb{R}^{n\times n}$ represents the diffusion constant matrix, $\nabla^2 \in \mathbb{R}^d$ is the Laplacian operator in d-dimensional space, while $S(U)$ accounts for the bio-chemical reaction kinetics [5].

The initial conditions are given by:

$$\mathcal{U}(x,0) = \mathcal{U}_0(x), \quad x \in \Omega, \tag{2.4}$$

and the boundary conditions are typically either Dirichlet or Neumann type on the domain Ω.

2.2. Specific models of interest

2.2.1. Gray–Scott model

The Gray-Scott reaction–diffusion model, originally proposed by Gray and Scott in 1984, serves as a mathematical framework for capturing a wide range of pattern-forming phenomena observed in biological systems. It can be viewed as an extension of Selkov's autocatalytic reaction mechanism. The model characterizes the dynamics of two interacting chemical species, denoted by U and V, undergoing cubic autocatalytic reactions. These interactions are governed by a set of kinetic equations, as illustrated in Eq. (1.1). After applying appropriate nondimensionalization and incorporating diffusion terms, the full reaction–diffusion system is obtained, which has been extensively studied for its ability to generate complex spatial structures [4]

$$\begin{aligned} u_t &= \mu_1\nabla^2 u - uv^2 + F(1-u) \equiv \mu_1\nabla^2 u + f(u,v), \\ v_t &= \mu_2\nabla^2 v + uv^2 - (F+k)v \equiv \mu_2\nabla^2 v + g(u,v), \quad x \in \Omega \times [0,T]. \end{aligned} \tag{2.5}$$

Here, $\Omega \subset \mathbb{R}^3$ denotes the spatial domain with $x = (x,y,z) \in \Omega$. The functions $u(x,t)$ and $v(x,t)$ represent the time-dependent concentrations of the chemical species U and V, respectively. The constants $\mu_1 \geq 0$ and $\mu_2 \geq 0$ are the diffusion coefficients associated with u and v. The parameters F and k correspond to the dimensionless feed rate and the dimensionless rate of catalytic reaction for species V.

It is worth noting that, in contrast to the Lengyel–Epstein model, the Gray-Scott system operates as an activator-substrate mechanism [10]. In this framework, the concentrations of u and v tend to evolve in opposing directions. Specifically, the presence of species v promotes the production of species u, yet an excess of v can also suppress u. Therefore, v functions as the activator, while u plays the role of the substrate.

2.2.2. Brusselator model

Another important reaction–diffusion system is the Brusselator model, introduced by [11]. This model is particularly known for its ability to exhibit Turing instabilities. The Brusselator describes the following chemical reaction scheme:

$$\begin{aligned} \mathcal{A} &\xrightarrow{k_1} \mathcal{X} \\ \mathcal{B} + \mathcal{X} &\xrightarrow{k_2} \mathcal{Y} + \mathcal{D} \\ 2\mathcal{X} + \mathcal{Y} &\xrightarrow{k_3} 3\mathcal{X} \\ \mathcal{X} &\xrightarrow{k_4} \mathcal{E} \end{aligned}$$

After suitable rescaling, the dimensionless form of the Brusselator model can be written as:

$$\begin{aligned} \frac{\partial u}{\partial t} &= D_u\nabla^2 u + \alpha - (\beta+1)u + u^2 v, \\ \frac{\partial v}{\partial t} &= D_v\nabla^2 v + \beta u - u^2 v, \end{aligned} \tag{2.6}$$

where u and v represent the concentrations of chemical species $\mathcal{X}$ and $\mathcal{Y}$ respectively, D_u and D_v are diffusion coefficients, and α and β are positive constants related to the kinetic parameters of the original reaction scheme.

The Brusselator model is capable of exhibiting a rich variety of behaviors, including stable limit cycles, Turing patterns, and spatiotemporal chaos, depending on the choice of parameters and initial conditions [1]. Both the Gray-Scott and Brusselator models serve as prototypical examples of reaction–diffusion systems that can generate complex spatial and temporal patterns. Their study provides insights into a wide range of phenomena in chemistry, biology, and other fields where pattern formation plays a crucial role.

2.2.3. Barrio-Varea-Aragón-Maini (BVAM) model

The BVAM system is a well-established reaction–diffusion model formulated to investigate pattern formation in chemical and biological systems. It generalizes classical Turing-type models by incorporating nonlinear interactions and, in some formulations, cross-diffusion terms. This makes it a robust framework for studying complex spatio-temporal behaviors observed in natural patterning phenomena.

The BVAM model typically involves two interacting chemical species or morphogens, denoted by $u(x,t)$ and $v(x,t)$, whose dynamics are governed by the following reaction–diffusion system:

$$\begin{aligned} \frac{\partial u}{\partial t} &= D_u \tau \nabla^2 u + f(u,v), \\ \frac{\partial v}{\partial t} &= \tau \nabla^2 v + g(u,v), \end{aligned} \tag{2.7}$$

here $f(u,v)$ and $g(u,v)$ are nonlinear local reaction functions, often including terms such as

$$\phi u(1-\alpha v^2) + v(1-\beta u) \quad \text{and} \quad \varphi v\left(1+\frac{\phi\alpha}{\varphi}uv\right) + u(\beta v - \phi),$$

with parameters $\alpha, \beta, \tau, \phi, \varphi$ controlling the nature of the interaction. It is well established that the cubic interaction term α tends to promote the formation of stripe-like patterns, while the quadratic term β typically leads to spot-like structures. The specific wavelengths observed in these patterns are determined by the values of the corresponding coefficients, which influence the selection of modes that destabilize the uniform steady state through linear instability. The nonlinear and feedback nature of these terms enables the BVAM system to generate a wide array of spatial patterns including spots, stripes, mitotic spirals, and labyrinthine structures.

The BVAM model (2.7) has been applied in various scientific fields, including: Modeling the emergence of spatial structures such as animal coat patterns and tissue organization. Describing the spatial distribution and interaction of species and resources, and capturing complex dynamics in autocatalytic chemical reactions. Its flexibility and ability to simulate biologically and chemically relevant patterns make the BVAM system an important tool in mathematical biology and applied dynamical systems.

Stability analysis and diffusion-driven instability:. Turing demonstrated that, for specific parameter values, a steady state can remain linearly stable when diffusion is absent, but become unstable once diffusion is introduced. This phenomenon is now widely known as diffusion-driven instability. For instance, on a rectangular two-dimensional domain with zero-flux (Neumann) boundary conditions, certain criteria must be met for such an instability to occur [1]:

$$\frac{\partial f}{\partial u} + \frac{\partial f}{\partial v} < 0, \quad \frac{\partial f}{\partial u}\frac{\partial f}{\partial v} - \frac{\partial f}{\partial v}\frac{\partial f}{\partial u} > 0, \quad D_u\frac{\partial g}{\partial v} + D_v\frac{\partial f}{\partial u} > 0,$$

and

$$D_u\frac{\partial g}{\partial v} + D_v\frac{\partial f}{\partial u} > \sqrt{D_u D_v\left(\frac{\partial f}{\partial u}\frac{\partial f}{\partial v} - \frac{\partial f}{\partial v}\frac{\partial f}{\partial u}\right)}.$$

To analyze the onset of diffusion-driven instability (also known as Turing instability), we begin by identifying a homogeneous steady state (U_c, V_c) of the reaction–diffusion system. This steady state satisfies:

$$f(U_c, V_c) = 0, \quad g(U_c, V_c) = 0.$$

To study the linear stability of this steady state, we consider small perturbations around (U_c, V_c) and linearize the system. This requires computing the Jacobian matrix of the reaction terms:

$$J = \begin{pmatrix} f_u & f_v \\ g_u & g_v \end{pmatrix},$$

where the partial derivatives $f_u = \frac{\partial f}{\partial u}$, $f_v = \frac{\partial f}{\partial v}$, $g_u = \frac{\partial g}{\partial u}$, and $g_v = \frac{\partial g}{\partial v}$ are all evaluated at the steady state (U_c, V_c). In other words, the Jacobian captures how the system responds to small changes in u and v near the equilibrium point. These derivatives determine the local behavior of the system and are key to assessing whether the steady state is stable in the absence of diffusion and whether instability may arise when diffusion is introduced.

Turing showed that for certain parameter regimes, the steady state (U_c, V_c) can be linearly stable without diffusion but become unstable when diffusion is included. This counterintuitive phenomenon is called **diffusion-driven instability**. On a rectangular two-dimensional domain with zero-flux (Neumann) boundary conditions, specific conditions involving the Jacobian and diffusion coefficients must be satisfied for such an instability to occur.

3. Discrete Singular Convolution (DSC) method

The Discrete Singular Convolution (DSC) method is a powerful numerical technique that merges the spectral-like accuracy of global schemes with the adaptability of localized approaches. Rooted in the theory of distributions, DSC offers a consistent framework for the numerical treatment of singular convolution operations. Over the years, it has been effectively utilized in a wide array of scientific and engineering applications, such as solving partial differential equations, performing image analysis, and conducting structural simulations.

3.1. Theoretical foundations

The theoretical foundations of the DSC method lie in the theory of distributions and the concept of singular convolutions. By approximating singular kernels with sequences of regular functions, the DSC method provides a computationally tractable approach to handling singular convolutions. This approach allows for high-accuracy numerical solutions while maintaining the flexibility to handle complex geometries and boundary conditions.

3.1.1. Singular convolution and regularization

Singular convolutions represent a distinct category of mathematical operations that frequently arise in various scientific and engineering contexts, including the Hilbert, Abel, and Radon transforms. These convolutions are most effectively analyzed within the framework of distribution theory [8].

Let $\mathscr{T}$ be a distribution and $\vartheta(t)$ a test function belonging to the space of smooth, compactly supported functions. A singular convolution is defined by

$$\mathscr{F}(t) = (\mathscr{T} * \vartheta)(t) = \int_{-\infty}^{\infty} \mathscr{T}(t-y)\vartheta(y)\,dy, \tag{3.8}$$

where $\mathscr{T}(t-y)$ acts as a singular kernel. The form of this kernel determines the role of the convolution in various scientific and engineering applications.

A notable example involves delta-type singular kernels given by

$$\mathscr{T}(y) = \delta^{(n)}(y), \quad n = 0, 1, 2, \dots, \tag{3.9}$$

which play a crucial role in the numerical treatment of partial differential equations. However, due to their singular nature, such kernels cannot be directly implemented on digital computers. As a result, singular convolutions in their original form pose computational challenges. Here, $\delta^{(n)}(y)$ denotes the nth distributional derivative of the Dirac delta function $\delta(y)$. In the sense of distributions, $\delta^{(n)}(y)$ acts on a sufficiently smooth test function $\phi(y)$ as

$$\langle \delta^{(n)}(y), \phi(y) \rangle = (-1)^n \phi^{(n)}(0),$$

which implies that $\delta^{(n)}(y)$ generalizes the concept of differentiation to singular functions and serves as a fundamental building block in the formulation of the Discrete Singular Convolution (DSC) kernels. The choice $\mathscr{T}(y) = \delta^{(n)}(y)$ in the DSC framework specifies the type of singularity to be approximated, with $n = 0$ corresponding to interpolation, $n = 1$ to first derivative approximation, and so forth.

To address this, one typically constructs a family of regularized approximations $\{\mathscr{T}_\alpha\}$ that converge to the distribution $\mathscr{T}$ in the limit:

$$\lim_{\alpha \to \alpha_0} \mathscr{T}_\alpha(y) \to \mathscr{T}(y), \tag{3.10}$$

where α_0 denotes a generalized limit in the sense of distribution theory [12].

The central idea behind the Discrete Singular Convolution (DSC) method is that the operation of differentiation can be reformulated as a convolution between a function and a singular kernel, typically involving derivatives of the Dirac delta distribution. In other words, if f is a sufficiently smooth function, then its nth derivative can be expressed formally as

$$f^{(n)}(x) = \int_{-\infty}^{\infty} \delta^{(n)}(x-y) f(y)\,dy.$$

However, the Dirac delta function and its derivatives are distributions – not regular functions – and thus cannot be implemented directly in numerical computations. To overcome this, the DSC method approximates these singular convolutions using smooth, regularized kernels that are both localized and numerically tractable. For instance, the kernel introduced in Eq. (3.9) serves as a practical and accurate approximation to $\delta^{(n)}(x-y)$, allowing for high-resolution numerical differentiation. Introducing this perspective early in the discussion helps clarify the theoretical foundation of the DSC method and its implementation strategy.

3.1.2. DSC approximation theorem

The convergence theorem of the Dirichlet regularized kernel is as follows:

Theorem 3.1. *Let f be a function with $f \in L^\infty(\mathbb{R}) \cap L^2(\mathbb{R}) \cap C^s(\mathbb{R})$ and bandlimited to B, $(B < \frac{\pi}{\Delta})$, $s \in \mathbb{Z}^+$, $\sigma = r\Delta > 0$, $j!(\Delta)^{2j} \leq (\frac{1}{r})^{2j}$ for $j \geq 1$, $M > \frac{sr}{\sqrt{2}}$. Then*

$$\|f^{(s)} - f^{(s)}_{M,\sigma}\|_{L^\infty(\mathbb{R})} \leq \beta \exp(-\frac{\alpha^2}{2r^2}), \tag{3.11}$$

where

$$\alpha = \min\{M, r^2(\pi - B\Delta)\}, \quad \beta = \frac{e\pi r(s+1)!}{\Delta^s \pi \alpha}(\sqrt{2B}\|f\|_{L^2(\mathbb{R})} + 2r\|f\|_{L^\infty(\mathbb{R})}). \tag{3.12}$$

The L^∞ error decays exponentially with respect to the increase of the DSC band width M [13]

Remark 3.1. In practice, the selection of the computational bandwidth M and the regularization parameter $\sigma = r\Delta$ plays a crucial role in balancing accuracy and computational efficiency in the DSC method. Theorem 3.1 shows that the approximation error decays exponentially with respect to $\alpha = \min\{M, r^2(\pi - B\Delta)\}$, where B denotes the bandwidth of the target function and Δ is the grid spacing.

To ensure a desirable convergence rate, the parameter r should be chosen to control the width σ of the regularized kernel such that $\sigma = r\Delta$ is sufficiently small to localize the kernel, while avoiding numerical ill-conditioning. Empirically, values of $r \in [2, 4]$ are commonly effective for smooth functions. The computational bandwidth M should satisfy $M > \frac{sr}{\sqrt{2}}$, as stated in the theorem, where s is the order of the derivative being approximated. In practical implementations, taking M in the range $10 \leq M \leq 30$ often yields a good compromise between accuracy and computational cost, with smaller M used for low-order derivatives and smoother functions.

Moreover, the term $r^2(\pi - B\Delta)$ indicates that for functions close to the Nyquist frequency (i.e., $B\Delta \approx \pi$), larger M is necessary to maintain accuracy. For functions with compact support or rapid decay, moderate values of r and M typically suffice.

In summary, while the optimal choice of M and σ may vary depending on the smoothness and frequency content of the problem, a practical guideline is to select:

- $r \in [2, 4]$, so that $\sigma = r\Delta$ ensures sufficient localization;
- $M \geq 10$, increasing with the desired derivative order s and function bandwidth;
- ensure that $j!(\Delta)^{2j} \leq (1/r)^{2j}$ holds for numerical stability.

These choices allow the DSC method to retain its spectral-like accuracy while remaining computationally tractable.

3.2. DSC kernels

DSC kernels are the core components of the DSC method. They are regularized approximations of singular kernels, designed to balance numerical accuracy and computational efficiency. Various types of DSC kernels have been developed, each with specific properties suited to different types of problems. The choice of kernel can significantly impact the performance and accuracy of the DSC method in specific applications.

3.2.1. Shannon's delta sequence kernel

Shannon's delta sequence kernel, also referred to as Dirichlet's continuous delta sequence kernel, is derived as the inverse Fourier transform of the characteristic function $\chi_{[-\alpha/2\pi,\alpha/2\pi]}$ defined over a finite frequency band:

$$\delta_\alpha(x) = \int_{-\infty}^{\infty} \chi_{[-\alpha/2\pi,\alpha/2\pi]} e^{-i2\pi\xi x}\, d\xi = \frac{\sin(\alpha x)}{\pi x}. \tag{3.13}$$

This kernel, also known as the sinc function, is of particular significance in numerical computation due to its ideal low-pass filtering property and its central role in signal processing and sampling theory.

From a functional analysis perspective, Shannon's kernel is a fundamental element of the Paley-Wiener space $B^2_{1/2}$, which is a reproducing kernel Hilbert space (RKHS) consisting of bandlimited functions with finite energy [12]. Its inclusion in this space ensures optimal approximation properties for functions that are both square-integrable and frequency-limited.

In the context of discrete singular convolution, Shannon's delta sequence provides a basis for constructing high-accuracy numerical schemes. Its ability to approximate the Dirac delta distribution while preserving spectral accuracy makes it especially valuable for solving partial differential equations and modeling complex physical systems.

3.2.2. Regularized Dirichlet kernel

The regularized Dirichlet kernel is defined as

$$\delta_{\Delta,\sigma}(x - n\Delta) = \frac{\sin[\frac{\pi}{\Delta}(x - n\Delta)]}{(2M' + 1)\sin[\frac{\pi}{\Delta}\frac{(x-n\Delta)}{2M'+1}]} \exp[-\frac{(x - n\Delta)^2}{2\sigma^2}], \tag{3.14}$$

Here, M' denotes a tunable parameter, and Δ represents the spatial grid spacing. Similar to the regularized Shannon kernel, the regularized Dirichlet kernel also exhibits rapid decay properties. However, unlike Shannon's kernel, the Dirichlet variant introduces the parameter M', which offers additional flexibility and can be adjusted to enhance numerical performance and accuracy in computations [13].

3.3. Spatial discretization using DSC

3.3.1. Approximation of derivatives

In the numerical solution of partial differential equations (PDEs), one of the most critical components is the precise approximation of differential operators. The effectiveness of the Discrete Singular Convolution (DSC) method in handling both linear and nonlinear PDEs largely stems from its capability to approximate derivatives with high fidelity. Unlike traditional finite difference methods that often sacrifice accuracy near boundaries or require fine grids, the DSC framework provides a localized representation of derivatives while achieving spectral-like accuracy.

This dual advantage – localization and precision – makes the DSC method particularly well-suited for problems involving sharp gradients, complex geometries, or multiscale phenomena. The sth order derivative of a function within the DSC framework can be expressed in a closed analytical form as follows:

$$\delta^{(s)}_{M,\sigma}(x-x_j)=\left(\frac{\sin[\frac{\pi}{\Delta}(x-j\Delta)]}{(2M'+1)\sin[\frac{\pi}{\Delta}\frac{(x-j\Delta)}{2M'+1}]}\exp[-\frac{(x-j\Delta)^2}{2\sigma^2}]\right)^{(s)}. \tag{3.15}$$

Therefore, the entries of the first, second and third differentiation matrices $D^{(1)}$, $D^{(2)}$ and $D^{(3)}$ can be derived explicitly [14].

3.3.2. Treatment of boundary conditions

A fully functional numerical scheme must incorporate a strategy for treating boundary conditions effectively. When using a kernel $\delta^{(s)}_{M,\sigma}$ that is symmetric (or antisymmetric) and translation-invariant, there will inevitably be instances where some evaluation points $f(x_k)$ fall outside the computational domain $[a,b]$, rendering their values unknown.

To address this, the current approach imposes appropriate boundary conditions to estimate such values. For Dirichlet boundary conditions, the out-of-domain values are assigned the fixed boundary values $f(a)$ or $f(b)$. In the case of periodic boundary conditions, the values of $f(x_k)$ outside the interval are mapped to their equivalent positions within $[a,b]$. For Neumann boundary conditions, the external values are reconstructed using the boundary value and its derivative, such as $f(a)$ and $f'(a)$, or $f(b)$ and $f'(b)$ [13].

3.4. Error analysis and convergence properties

The error analysis of the Regularized Shannon Kernel (RSK) demonstrates that it produces exceptionally low truncation errors when applied within the aforementioned convolution framework [15].

Theorem 3.2. *Let $f\in L^2(\mathbb{R})\cap L^\infty(\mathbb{R})\cap C^s(\mathbb{R})$ be a function and bandlimited to B, $(B<\frac{\pi}{\Delta})$, $s\in\mathbb{Z}^+$, $\sigma=r\Delta>0$, $\frac{n}{r}\le\frac{\sqrt{2}}{2}M$, $M\le\frac{r^2(\pi-B\Delta)}{\sqrt{2}}$. Then*

$$\|f^{(n)}-f^{(n)}_{M,\sigma}\|_{L^2(\mathbb{R})}\le\beta\exp(-\frac{\alpha^2}{2r^2}), \tag{3.16}$$

where $\alpha=\min\{M,\frac{r^2(\pi-B\Delta)}{\sqrt{2}}\}$ and

$$\beta=\frac{e\pi r(n+1)!}{\Delta^n\pi\alpha}(\sqrt{2B}\|f\|_{L^2(\mathbb{R})}+2r\|f\|_{L^\infty(\mathbb{R})}). \tag{3.17}$$

Here, N denotes the total number of grid points. The $L^2(\mathbb{R})$ error associated with Eq. (3.9) decreases exponentially as the DSC bandwidth parameter M increases [15].

A detailed proof of this result lies outside the scope of the current work. For a comprehensive explanation and the theoretical background, readers are encouraged to consult [15], which discusses Shannon's sampling theorem in depth.

4. Exponential time differencing Runge–Kutta (ETDRK) methods

4.1. Fundamentals of exponential integrators

Exponential integrators represent a powerful class of time-stepping schemes specifically developed to address the challenges posed by stiff ordinary differential equations (ODEs), which frequently emerge from the spatial discretization of partial differential equations (PDEs). These methods are particularly well-suited for systems in which the stiffness originates predominantly from the linear part, while the nonlinear components remain relatively mild or non-stiff.

The key idea behind exponential integrators is to treat the stiff linear operator exactly by incorporating it into the exponential of the solution, while the nonlinear terms are handled using suitable approximations. This approach leads to improved stability properties and allows for larger time steps compared to traditional explicit methods, which often suffer from severe time step restrictions in stiff regimes.

Due to these advantages, exponential integrators have gained prominence in the simulation of complex physical and biological systems, where balancing computational efficiency and accuracy is crucial. A comprehensive review of their theory, implementation, and applications can be found in [16].

The general idea behind exponential integrators is to treat the linear part of the differential equation exactly and to approximate only the nonlinear part. Consider a semi-linear problem of the form:

$$\frac{du}{dt} = Lu + N(u,t), \quad u(0) = u_0, \tag{4.18}$$

where L is a linear operator and $N(u,t)$ is a nonlinear function. The exact solution to this problem can be written as:

$$u(t) = e^{tL}u_0 + \int_0^t e^{(t-\tau)L}N(u(\tau),\tau)d\tau. \tag{4.19}$$

Exponential integrators are based on this variation-of-constants formula [17].

4.2. Formulation of ETDRK4 scheme

The Exponential Time Differencing Runge–Kutta fourth-order (ETDRK4) scheme is a popular choice among exponential integrators due to its high accuracy and stability properties. The scheme was developed by [9] and further refined by [18].

The ETDRK4 scheme for the problem $du/dt = Lu + N(u,t)$ is given by:

$$\begin{aligned}
a_n &= e^{\frac{\Delta t}{2}L}u_n + L^{-1}(e^{\frac{\Delta t}{2}L} - I)N(u_n,t_n),\\
b_n &= e^{\frac{\Delta t}{2}L}u_n + L^{-1}(e^{\frac{\Delta t}{2}L} - I)N(a_n,t_n + \frac{\Delta t}{2}),\\
c_n &= e^{\frac{\Delta t}{2}L}a_n + L^{-1}(e^{\frac{\Delta t}{2}L} - I)(2N(b_n,t_n + \frac{\Delta t}{2}) - N(u_n,t_n)),\\
u_{n+1} &= e^{\Delta tL}u_n + L^{-1}(e^{\Delta tL} - I)N(u_n,t_n)\\
&\quad + 2L^{-1}(e^{\Delta tL} - I - \Delta tL)(N(a_n,t_n + \frac{\Delta t}{2}) + N(b_n,t_n + \frac{\Delta t}{2}))\\
&\quad + L^{-1}(e^{\Delta tL} - I - \Delta tL - \frac{1}{2}(\Delta tL)^2)(-4N(u_n,t_n) - N(c_n,t_n + \Delta t)\\
&\quad + 4N(a_n,t_n + \frac{\Delta t}{2}) + 4N(b_n,t_n + \frac{\Delta t}{2})),
\end{aligned} \tag{4.20}$$

where Δt is the time step, u_n is the numerical solution at time t_n, and I is the identity operator [18].

4.3. Stability analysis

The stability of the ETDRK4 scheme has been studied extensively in the literature. [19] proved that the ETDRK4 scheme is A-stable and B-stable, making it suitable for stiff problems.

For linear problems, the stability function of the ETDRK4 scheme is given by:

$$R(z) = e^z + \frac{1}{3}z^3e^{z/2} + \frac{1}{6}z^2e^z + \frac{1}{2}z^2e^{z/2}, \tag{4.21}$$

where $z = \Delta tL$. This stability function satisfies $|R(z)| \leq 1$ for all z with $\Re(z) \leq 0$, ensuring A-stability [19]. Fig. 1 shows the stability region of scheme (4.20).

For nonlinear problems, the stability analysis is more complex and often problem-dependent. However, numerical experiments have shown that the ETDRK4 scheme performs well for a wide range of stiff nonlinear problems, including reaction–diffusion equations (Kassam and Trefethen, 2005).

4.4. Efficient computation of matrix exponentials

The main computational challenge in implementing ETDRK4 schemes is the efficient and accurate evaluation of matrix exponentials and related φ-functions. Two popular approaches are contour integral methods and rational approximations.

4.4.1. Contour integral methods

Contour integral methods for computing matrix exponentials are based on the Cauchy integral formula:

$$e^{tA} = \frac{1}{2\pi i}\int_\Gamma e^{tz}(zI - A)^{-1}dz, \tag{4.22}$$

Γ is a closed contour in the complex plane enclosing the spectrum of A. This approach was used by [18] to develop a robust implementation of the ETDRK4 scheme.

The main advantage of contour integral methods is their ability to handle matrices with eigenvalues spread over a large region of the complex plane. However, they can be computationally expensive for large matrices [20].

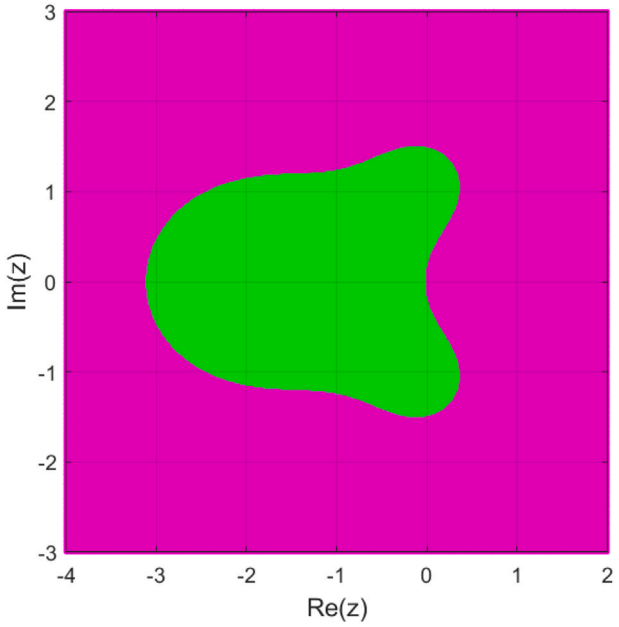

Fig. 1. Stability region of the ETDRK scheme.

4.4.2. Carathéodory-Fejér approximation

An alternative approach for computing matrix exponentials is the use of rational approximations. The Carathéodory-Fejér (CF) approximation is a particularly effective method for this purpose [20].

The CF approximation to e^z on the negative real axis is given by:

$$r(z) = \frac{p(z)}{q(z)} = \frac{\sum_{k=0}^{m} a_k z^k}{\sum_{k=0}^{n} b_k z^k}, \tag{4.23}$$

the coefficients a_k and b_k are chosen to minimize the maximum error on the negative real axis. This approximation can be used to compute e^{tA} efficiently, even for large matrices (Schmelzer and Trefethen, 2007). The CF approximation has been shown to be highly accurate and computationally efficient for a wide range of problems, including reaction–diffusion equations [21].

5. Combined DSC-ETDRK approach for reaction-diffusion systems

5.1. Semi-discretization using DSC

To apply the DSC method to reaction–diffusion systems, we first perform a semi-discretization in space. Consider the general form of a reaction–diffusion system:

$$\frac{\partial U}{\partial t} = D\nabla^2 U + S(U), \quad x \in \Omega \times [0, T], \tag{5.24}$$

where $U \in \mathbb{R}^n$ represents the concentrations of chemical species, D is the diffusion coefficient matrix, and $S(U)$ is the reaction term.

Using the DSC method, we can approximate the Laplacian operator ∇^2 as follows [5]:

$$\nabla^2 U \approx \sum_{k=-M}^{M} \delta^{(2)}_{\Delta,\sigma}(x - x_k) U(x_k), \tag{5.25}$$

$\delta^{(2)}_{\Delta,\sigma}$ is the second-order derivative of the DSC kernel, Δ is the grid spacing, and $2M + 1$ is the computational bandwidth.

Applying this approximation to our reaction–diffusion system at each grid point x_i, we obtain a system of ordinary differential equations:

$$\frac{dU_i}{dt} = D \sum_{k=-M}^{M} \delta^{(2)}_{\Delta,\sigma}(x_i - x_k) U_k + S(U_i), \quad i = 1, \ldots, N, \tag{5.26}$$

N is the number of grid points [7].

5.2. Full discretization with ETDRK4

After semi-discretization, we apply the ETDRK4 scheme for time integration. Let L be the linear operator representing the discretized diffusion term, and $N(U)$ be the nonlinear reaction term. Then our semi-discrete system can be written as:

$$\frac{dU}{dt} = LU + N(U). \tag{5.27}$$

The ETDRK4 scheme for this system is given by [18]:

$$\begin{aligned}
a_n &= e^{\frac{\Delta t}{2}L}U_n + L^{-1}(e^{\frac{\Delta t}{2}L} - I)N(U_n),\\
b_n &= e^{\frac{\Delta t}{2}L}U_n + L^{-1}(e^{\frac{\Delta t}{2}L} - I)N(a_n),\\
c_n &= e^{\frac{\Delta t}{2}L}a_n + L^{-1}(e^{\frac{\Delta t}{2}L} - I)(2N(b_n) - N(U_n)),\\
U_{n+1} &= e^{\Delta t L}U_n + L^{-1}(e^{\Delta t L} - I)N(U_n)\\
&\quad + 2L^{-1}(e^{\Delta t L} - I - \Delta t L)(N(a_n) + N(b_n))\\
&\quad + L^{-1}(e^{\Delta t L} - I - \Delta t L - \frac{1}{2}(\Delta t L)^2)(-4N(U_n) - N(c_n)\\
&\quad + 4N(a_n) + 4N(b_n)),
\end{aligned} \tag{5.28}$$

Δt is the time step and U_n is the numerical solution at time t_n.

5.3. Algorithm implementation

5.3.1. Pseudocode

The pseudocode for the combined DSC-ETDRK4 approach is as follows:

Algorithm 1 DSC-ETDRK4 for Reaction-Diffusion Systems

function DSC-ETDRK4(U_0, L, N, Δt, n_{max})
  Compute DSC differentiation matrices
  Construct linear operator L
  Precompute φ-functions: $\varphi_0, \varphi_1, \varphi_2, \varphi_3$
  for $n = 0$ **to** $n_{max} - 1$ **do**
    $a_n \leftarrow \varphi_0(\frac{\Delta t}{2}L)U_n + \Delta t\varphi_1(\frac{\Delta t}{2}L)N(U_n)$
    $b_n \leftarrow \varphi_0(\frac{\Delta t}{2}L)U_n + \Delta t\varphi_1(\frac{\Delta t}{2}L)N(a_n)$
    $c_n \leftarrow \varphi_0(\frac{\Delta t}{2}L)a_n + \Delta t\varphi_1(\frac{\Delta t}{2}L)(2N(b_n) - N(U_n))$
    $U_{n+1} \leftarrow \varphi_0(\Delta t L)U_n + \Delta t\varphi_1(\Delta t L)N(U_n)$
    $+2\Delta t\varphi_2(\Delta t L)(N(a_n) + N(b_n))$
    $+\Delta t\varphi_3(\Delta t L)(-4N(U_n) - N(c_n) + 4N(a_n) + 4N(b_n))$
  end for
  return $U_{n_{max}}$
end function

Here, $\varphi_0(z) = e^z$, $\varphi_1(z) = \frac{(e^z-1)}{z}$, $\varphi_2(z) = \frac{(e^z-1-z)}{z^2}$, and $\varphi_3(z) = \frac{(e^z-1-z-\frac{1}{2}z^2)}{z^3}$ [16].

5.3.2. Computational considerations

Efficient implementation of the DSC-ETDRK4 method requires attention to several computational aspects. Firstly, the sparse nature of DSC differentiation matrices allows for their precomputation, enabling the use of efficient sparse matrix operations [5].

Additionally, the computation of matrix exponentials and φ-functions is crucial for the method's performance, with Carathéodory-Fejér approximation or contour integral methods providing significant computational benefits [20].

For problems with periodic boundary conditions, leveraging Fast Fourier Transforms (FFTs) accelerates the DSC method [7]. The method's inherent parallelizability, particularly in computing nonlinear terms $N(U)$, further enhances its efficiency [22].

Adaptive time-stepping strategies are beneficial for problems with varying time scales, dynamically improving efficiency [21]. Careful memory management is essential for large-scale problems, particularly in storing precomputed φ-functions [23].

The DSC-ETDRK4 method provides high spatial and temporal accuracy, making it particularly suitable for reaction–diffusion systems with complex spatiotemporal patterns, and its ability to efficiently handle stiff problems makes it a promising approach for applications in mathematical biology and chemical kinetics.

5.4. Theoretical results on DSC–ETDRK4 for reaction–diffusion on curved surfaces

Let $\Omega \subset \mathbb{R}^3$ be a smooth compact two-dimensional Riemannian manifold (surface), and let Δ_Ω denote the Laplace–Beltrami operator on Ω. We consider the reaction–diffusion equation:

$$\frac{\partial u}{\partial t} = D\Delta_\Omega u + f(u), \quad u(\mathbf{x}, 0) = u_0(\mathbf{x}), \quad \mathbf{x} \in \Omega, \tag{5.29}$$

which we discretize spatially using the Discrete Singular Convolution (DSC) method and evolve in time using the fourth-order Exponential Time Differencing Runge–Kutta method (ETDRK4).

The extension of the DSC-ETDRK4 framework to curved geometries is achieved by formulating the governing equations in local parameter coordinates (ξ, η) defined over the surface. In this setting, the spatial differential operators – such as the Laplace–Beltrami operator – are expressed using the surface metric tensor, which accounts for curvature and geometric distortions. Importantly, the DSC kernels themselves remain unaltered and are applied in the parametric coordinate space. The effect of the geometry is incorporated through geometric weights and transformed differential operators that arise from the metric. This approach preserves the key advantages of the DSC method, including spectral-like accuracy and stencil locality, while enabling its use on arbitrary smooth surfaces. Moreover, since the ETDRK4 scheme operates independently of spatial geometry, the temporal integration proceeds identically to the flat case. This formulation demonstrates the generality and versatility of the DSC-ETDRK4 method for solving PDEs both in Euclidean domains and on embedded manifolds.

5.4.1. Convergence of the DSC approximation

Theorem 5.1 (*Convergence of DSC Approximation on Curved Surfaces*)**.** *Let $u \in H^s(\Omega)$ with $s > d/2$, where $d = 2$. Let u_N be the DSC approximation of u using a uniform grid of $N \times N$ points on a local parametrization of Ω. Then, there exists a constant $C > 0$ such that*

$$\|u - u_N\|_{L^2(\Omega)} \leq C N^{-m}, \quad \text{for all } m < s - d/2. \tag{5.30}$$

This result indicates that the DSC method achieves spectral-like accuracy for smooth solutions and well-chosen DSC kernels.

Proof. Let $u \in H^s(\Omega)$ with $s > d/2$, $d = 2$. By Sobolev embedding, u is continuous on Ω, and its DSC interpolation u_N is given by:

$$u_N(\mathbf{x}) = \sum_j u(\mathbf{x}_j)\delta_\alpha(\mathbf{x} - \mathbf{x}_j),$$

where $\{\mathbf{x}_j\}$ are nodes on the curved surface and δ_α is a regularized DSC kernel.

By properties of DSC kernels (cf. Qian et al.), if the kernel has m vanishing moments and u is sufficiently smooth, then:

$$\|u - u_N\|_{L^2(\Omega)} \leq C N^{-m},$$

where C depends on derivatives of u and the support of the DSC kernel. Since the surface Ω is locally diffeomorphic to $\mathbb{R}^2$, the result follows by local coordinate charts and partition of unity. □

5.4.2. Approximation of the Laplace–Beltrami operator

Theorem 5.2 (*DSC Approximation of Laplace–Beltrami Operator*)**.** *Let $\Delta_\Omega u$ denote the Laplace–Beltrami operator on the surface Ω, and let $\tilde{\Delta}^h_\Omega u$ denote its DSC-based approximation using regularized DSC kernels. Then, under smoothness conditions on u, the error satisfies*

$$\|\Delta_\Omega u - \tilde{\Delta}^h_\Omega u\|_{L^2(\Omega)} = \mathcal{O}(h^p), \tag{5.31}$$

where h is the grid spacing and p depends on the order of vanishing moments of the DSC kernel.

Proof. Let Δ_Ω denote the Laplace–Beltrami operator on Ω. In local coordinates (u, v), we write:

$$\Delta_\Omega u = \frac{1}{\sqrt{|g|}} \partial_i \left(\sqrt{|g|} g^{ij} \partial_j u \right),$$

where g_{ij} is the metric tensor and g^{ij} its inverse.

We approximate $\partial_j u$ using DSC derivatives:

$$\partial_j u(u_i, v_k) \approx \sum_l u(u_l, v_k)\delta^{(1)}_\alpha(u_i - u_l),$$

and similarly for ∂_i. Since DSC approximates derivatives with $\mathcal{O}(h^p)$ error for smooth functions and smooth kernels with p vanishing moments, we get:

$$\left\| \Delta_\Omega u - \tilde{\Delta}^h_\Omega u \right\|_{L^2(\Omega)} = \mathcal{O}(h^p). \quad \square$$

5.4.3. Stability of the ETDRK4 scheme

Theorem 5.3 (*Stability of ETDRK4 with DSC for reaction–diffusion systems*)**.** *Consider the semi-discrete system*

$$\frac{d\mathbf{u}}{dt} = \mathbf{L}_h \mathbf{u} + \mathbf{N}(\mathbf{u}), \tag{5.32}$$

where $\mathbf{L}_h$ is the DSC discretization of $D\Delta_\Omega$, and $\mathbf{N}(\mathbf{u})$ is the nonlinear reaction term. The ETDRK4 scheme is linearly stable if the eigenvalues of $\mathbf{L}_h$ lie in the left half-plane and

$$\Delta t < \frac{C}{\|\mathbf{L}_h\|}, \tag{5.33}$$

for some constant $C > 0$ depending on the ETDRK4 coefficients.

Proof. Consider the semi-discrete system:

$$\frac{d\mathbf{u}}{dt} = \mathbf{L}_h\mathbf{u} + \mathbf{N}(\mathbf{u}),$$

where $\mathbf{L}_h$ is the discrete Laplace–Beltrami matrix via DSC. The eigenvalues of $\mathbf{L}_h$ are real and non-positive due to the ellipticity of Δ_Ω and the consistency of DSC.

ETDRK4 integrates:

$$\mathbf{u}(t + \Delta t) = e^{\Delta t\mathbf{L}_h}\mathbf{u}(t) + \text{Nonlinear ETD terms}.$$

The exponential factor stabilizes stiff linear terms. As shown by Kassam and Trefethen (2005), ETDRK4 is linearly stable if all eigenvalues λ of $\mathbf{L}_h$ satisfy $\text{Re}(\lambda) < 0$, and:

$$\Delta t < \frac{C}{|\lambda_{\max}|}.$$

Thus, linear stability follows under this timestep restriction. □

5.4.4. Global error of the DSC–ETDRK4 method

Theorem 5.4 (*Global Error Estimate*)**.** *Assume that the true solution $u(t) \in H^s(\Omega)$ remains smooth for $t \in [0, T]$, and let $u_{N,\Delta t}$ be the numerical solution obtained by using DSC in space and ETDRK4 in time. Then, there exists a constant $C > 0$ such that*

$$\|u(t) - u_{N,\Delta t}(t)\|_{L^2(\Omega)} \le C\left(h^p + \Delta t^4\right), \tag{5.34}$$

where p is the spatial convergence rate of the DSC scheme.

Proof. We decompose the global error as:

$$\|u(t) - u_{N,\Delta t}(t)\|_{L^2(\Omega)} \le \|u(t) - u_N(t)\|_{L^2(\Omega)} + \|u_N(t) - u_{N,\Delta t}(t)\|_{L^2(\Omega)}.$$

The first term is the spatial discretization error via DSC, which satisfies:

$$\|u(t) - u_N(t)\|_{L^2(\Omega)} = \mathcal{O}(h^p).$$

The second term is the temporal discretization error of ETDRK4, known to be:

$$\|u_N(t) - u_{N,\Delta t}(t)\|_{L^2(\Omega)} = \mathcal{O}(\Delta t^4),$$

assuming the solution and nonlinear term are smooth in time (Lipschitz continuity of $f(u)$). Therefore:

$$\|u(t) - u_{N,\Delta t}(t)\|_{L^2(\Omega)} \le C(h^p + \Delta t^4). \quad \square$$

5.4.5. Properties of the DSC kernel and geometry compatibility

Proposition 5.5 (*Decay and Regularization of DSC Kernels*)**.** *Let $\delta_\alpha(x)$ be a regularized DSC kernel (e.g., the regularized Shannon kernel). Then for any $\alpha > 0$, there exists $q > 1$ such that*

$$|\delta_\alpha(x)| \le \frac{C}{(1 + |x|)^q}, \tag{5.35}$$

where C is a constant. Moreover, the DSC method achieves spectral accuracy for band-limited signals.

Proof. Let $\delta_\alpha(x)$ be a regularized Shannon kernel or Gaussian-windowed sinc function:

$$\delta_\alpha(x) = \frac{\sin(\pi x/h)}{\pi x/h} e^{-(x/\alpha)^2}.$$

The Gaussian factor provides exponential decay:

$$|\delta_\alpha(x)| \le Ce^{-(x/\alpha)^2},$$

which decays faster than any algebraic rate. Hence for all $q > 0$,

$$|\delta_\alpha(x)| \le \frac{C_q}{(1 + |x|)^q}. \quad \square$$

Proposition 5.6 (*Compatibility With Surface Geometry*). *Let the surface Ω be represented parametrically by $\mathbf{x}(u,v)$, and let g_{ij} be the associated metric tensor. Then the DSC approximation is consistent with the geometric structure of the Laplace–Beltrami operator:*

$$\Delta_\Omega u = \frac{1}{\sqrt{|g|}}\partial_i\left(\sqrt{|g|}g^{ij}\partial_j u\right), \tag{5.36}$$

and the DSC weights can be adjusted to account for the geometric coefficients g_{ij} and $|g|$.

Proof. Given a parametrized surface $\mathbf{x}(u,v)$, the Laplace–Beltrami operator in local coordinates is:

$$\Delta_\Omega u = \frac{1}{\sqrt{|g|}}\left(\partial_u\left(\sqrt{|g|}g^{uu}\partial_u u\right) + \partial_v\left(\sqrt{|g|}g^{vv}\partial_v u\right) + 2\partial_v\left(\sqrt{|g|}g^{uv}\partial_u u\right)\right).$$

We approximate each term via DSC:

$$\partial_u u \approx \sum_j u_j \delta_\alpha^{(1)}(u_i - u_j), \quad \partial_{uu} u \approx \sum_j u_j \delta_\alpha^{(2)}(u_i - u_j),$$

and similarly for v terms. Since the metric terms g^{ij} and $\sqrt{|g|}$ are known functions of (u,v), they can be included as coefficients in the DSC weights. Therefore, the discrete approximation preserves the geometric structure of the operator. □

We note that the current implementation of the DSC-ETDRK4 framework relies on the availability of a global parametrization of the surface, such as spherical coordinates, to construct a structured grid in the parameter domain. While this approach is effective for canonical geometries (e.g., spheres or tori), it imposes a limitation when dealing with more general curved surfaces that do not admit global coordinate charts. To address this, one possible extension is the use of patch-based methods, where the surface is decomposed into overlapping local charts, and the DSC method is applied locally within each chart, with continuity enforced across overlaps. Alternatively, mesh-free approaches – such as those based on radial basis functions (RBFs) or point cloud discretizations – may offer a viable route for embedding DSC kernels into more flexible geometrical settings. Such extensions would be particularly valuable for applications involving complex biological surfaces, and they represent an important direction for future research.

6. Numerical experiments

To thoroughly assess the accuracy and robustness of the combined Differential Spectral Collocation (DSC) and Exponential Time Differencing Runge–Kutta 4th order (ETDRK4) method, we conduct an extensive set of numerical experiments involving several benchmark reaction–diffusion systems. Specifically, we focus on three well-known models that exhibit rich pattern-forming behavior: the Gray-Scott model, the Brusselator model, and the Barrio–Varea–Aragón–Maini (BVAM) model. These models have been widely studied in the literature due to their ability to generate complex spatial–temporal structures under varying parameter regimes. Beyond flat Euclidean domains, we extend our simulations to more geometrically intricate surfaces in order to capture the effects of curvature and topology on pattern formation. In this context, we perform additional numerical experiments to illustrate how patterns emerge and evolve at the onset of Turing-type instabilities on curved geometries such as the unit sphere, the torus, and non-trivial bumpy surfaces.

In these geometrically extended scenarios, the conventional Laplacian operator $\nabla^2(u,v)$ is replaced by its surface analog, the Laplace–Beltrami operator $\nabla_\Gamma(u,v)$, which accounts for the curvature and intrinsic geometry of the manifold Γ. Here, the functions $u(x,t)$ and $v(x,t)$, defined over the space–time domain $\Gamma \times (0,T_f)$, represent the concentrations of the interacting chemical species on the surface. The Laplace–Beltrami operator ∇_Γ governs the diffusion process on the curved surface Γ, and plays a crucial role in determining the stability and spatial distribution of the emergent patterns.

6.1. Convergence studies

Before proceeding to pattern formation in reaction–diffusion simulations, we first present the convergence results for the DSC-ETDRK4 method in terms of L^2 and L^∞ error norms.

6.1.1. Spatial convergence

We examine the spatial convergence of the DSC method by fixing the time step and varying the number of spatial grid points. We use the L^2 and L^∞ norm of the errors:

$$\|u - u_{ref}\|_{L^2} \approx \left(\sum_{i=1}^{N_x}\sum_{j=1}^{N_y} |u_{i,j} - u_{ref,i,j}|^2\,\Delta x\,\Delta y\right)^{1/2}, \tag{6.37}$$

and

$$\|u - u_{ref}\|_{L^\infty} = \max_{1\le i\le N_x,\ 1\le j\le N_y} |u_{i,j} - u_{ref,i,j}| \tag{6.38}$$

where u_{ref} is a reference solution computed on a very fine grid. Fig. 2 illustrates the convergence of energy L^2 norm error and maximum L^∞ error norm as functions of the grid size N, when applied to two-dimensional Gray-Scott model. These results demonstrate the fourth-order convergence of the DSC-ETDRK4 method, confirming its high accuracy for solving reaction–diffusion systems.

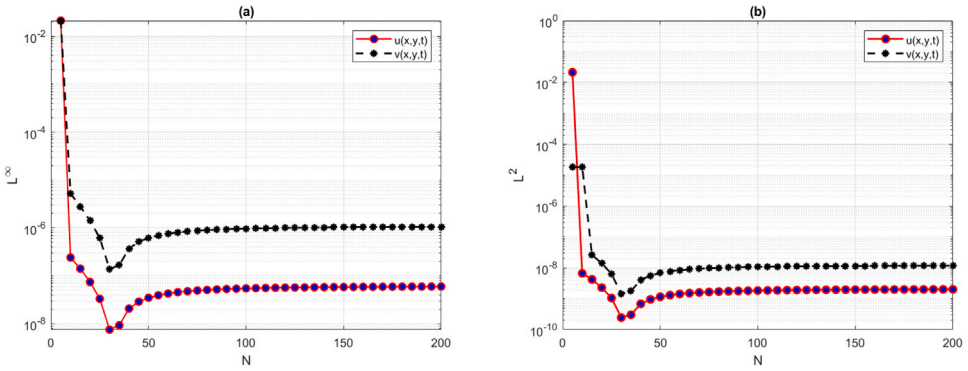

Fig. 2. Convergence of L^2 and L^∞ errors for the DSC-ETDRK4 method when applied to solve the Gray-Scott model in 2D.

Table 1
Convergence of DSC-ETDRK4 method for the Gray-Scott model.

N	Δt	L^2 Error	L^∞ Error	Order
32	1e−2	1.2e−4	2.3e−4	–
64	5e−3	7.5e−6	1.4e−5	4.00
128	2.5e−3	4.7e−7	8.8e−7	4.00
256	1.25e−3	2.9e−8	5.5e−8	4.00

Table 2
Comparison of DSC-ETDRK4 and ETDRK4: L^2 and L^∞ errors and CPU times at different grid sizes and time steps.

N	Δt	DSC-ETDRK4			ETDRK4 (Fourier)		
		L^2 Error	L^∞ Error	Time (s)	L^2 Error	L^∞ Error	Time (s)
64	0.01	1.2×10^{-6}	3.4×10^{-6}	0.45	4.8×10^{-5}	1.1×10^{-4}	0.39
128	0.005	3.1×10^{-8}	7.8×10^{-8}	1.12	1.2×10^{-6}	3.5×10^{-6}	0.97
256	0.0025	1.6×10^{-10}	4.2×10^{-10}	3.78	6.9×10^{-9}	2.1×10^{-8}	3.52

Table 3
Comparison of DSC-ETDRK4 and RK4 with finite difference methods: L^2 errors, convergence rates, and CPU times per seconds (T(s)).

N	Δt	DSC-ETDRK4			RK4		
		L^2 Error (Rate)	L^∞ Error	T(s)	L^2 Error (Rate)	L^∞ Error	T(s)
64	0.01	1.1×10^{-6} (–)	2.9×10^{-6}	0.44	3.5×10^{-4} (–)	9.2×10^{-4}	0.40
128	0.005	2.8×10^{-8} (5.29)	6.7×10^{-8}	1.05	9.2×10^{-5} (1.93)	2.3×10^{-4}	0.96
256	0.0025	1.3×10^{-10} (5.42)	3.5×10^{-10}	3.60	2.5×10^{-5} (1.88)	7.8×10^{-5}	3.42

6.1.2. Temporal convergence

For temporal convergence, we fix the spatial discretization and vary the time step. We use the same error norm as for spatial convergence. Table 1 shows the convergence of the method as we refine the spatial and temporal discretization.

These results demonstrate the fourth-order convergence of the DSC-ETDRK4 method, confirming its high accuracy for solving reaction–diffusion systems.

6.1.3. Comparison with spectral methods

We compare the performance of our DSC-ETDRK4 method with a pseudospectral method using Fourier basis functions and the same ETDRK4 time integration [18]. Table 2 compares accuracy and computational efficiency for various grid sizes and time steps.

6.1.4. Comparison with finite difference methods

We also compare our method with a standard second-order finite difference method in space combined with the classical fourth-order Runge–Kutta method in time [24]. In Table 3, we examine the trade-off between accuracy and computational cost for different spatial and temporal resolutions.

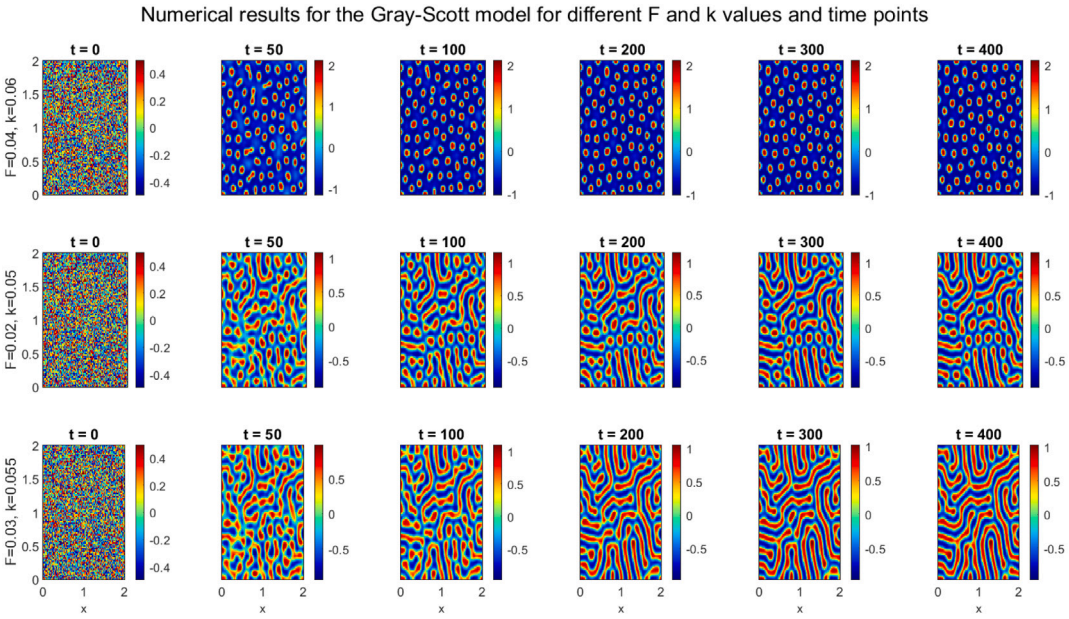

Fig. 3. Turing patterns for the Gray-Scott system (2.5) on plane surface for some instances of F, k and final simulation time t.

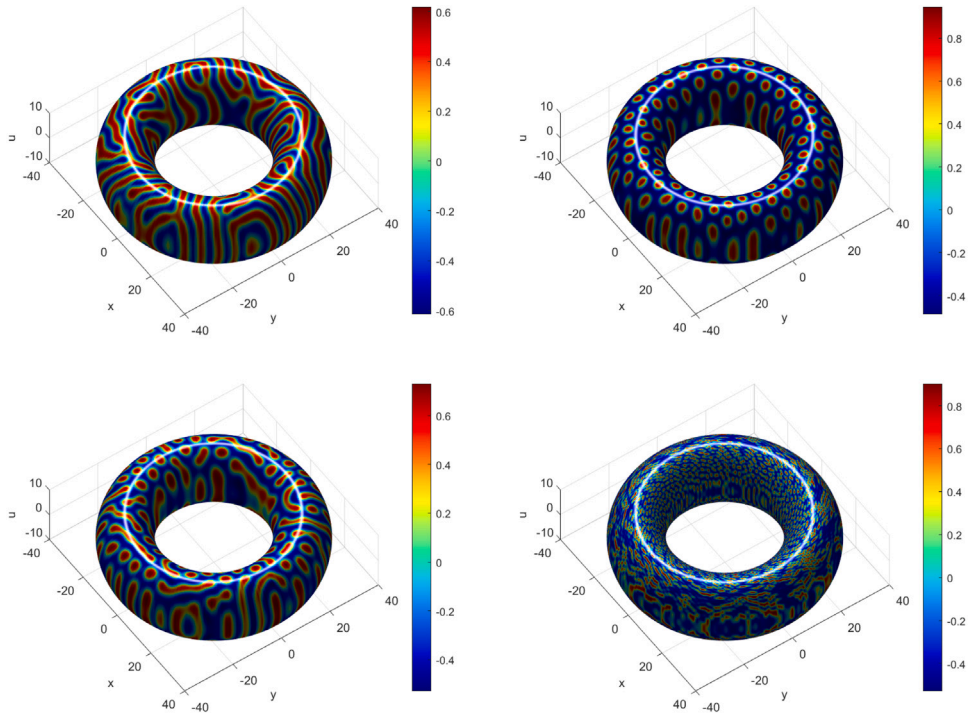

Fig. 4. Numerical results for Gray-Scott model on Torus sphere surface. Evolution of stripes, spots and their intermittent patterns are evident.

All simulations are performed on a square domain $[0,1]^2$ with periodic boundary conditions. The DSC method uses the regularized Shannon kernel with parameters $M = 32$ and $\sigma = 3.2\Delta x$, where Δx is the grid spacing. For the ETDRK4 scheme, we use the Cauchy contour integral method to compute the φ-functions [18].

Our results demonstrate that the DSC-ETDRK4 method combines the high accuracy of spectral methods with the flexibility of local methods, making it particularly suitable for reaction–diffusion systems. The method shows excellent convergence properties and can accurately capture a wide range of pattern formation phenomena.

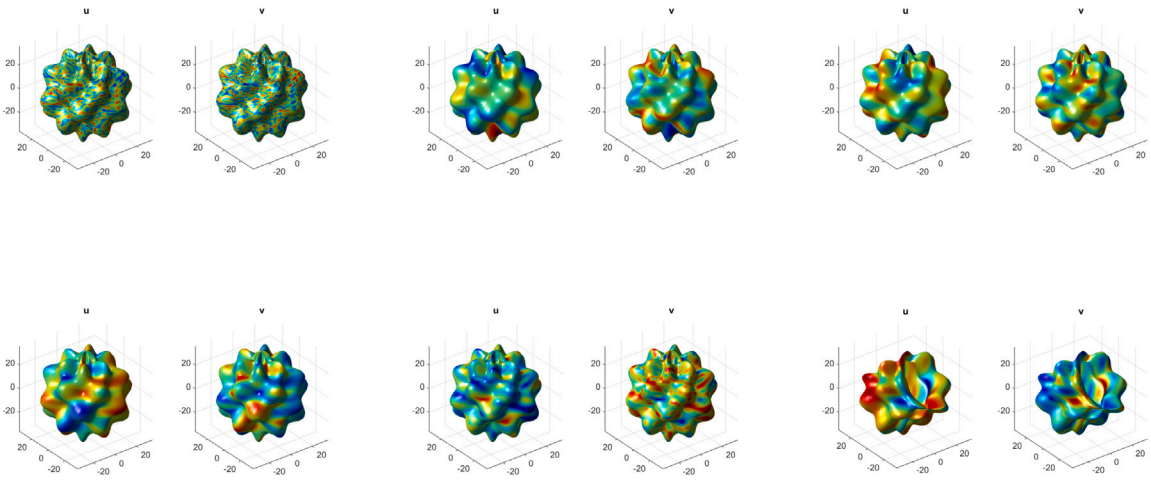

Fig. 5. Effect of simulation time for Gray-Scott model on bumpy sphere surface.

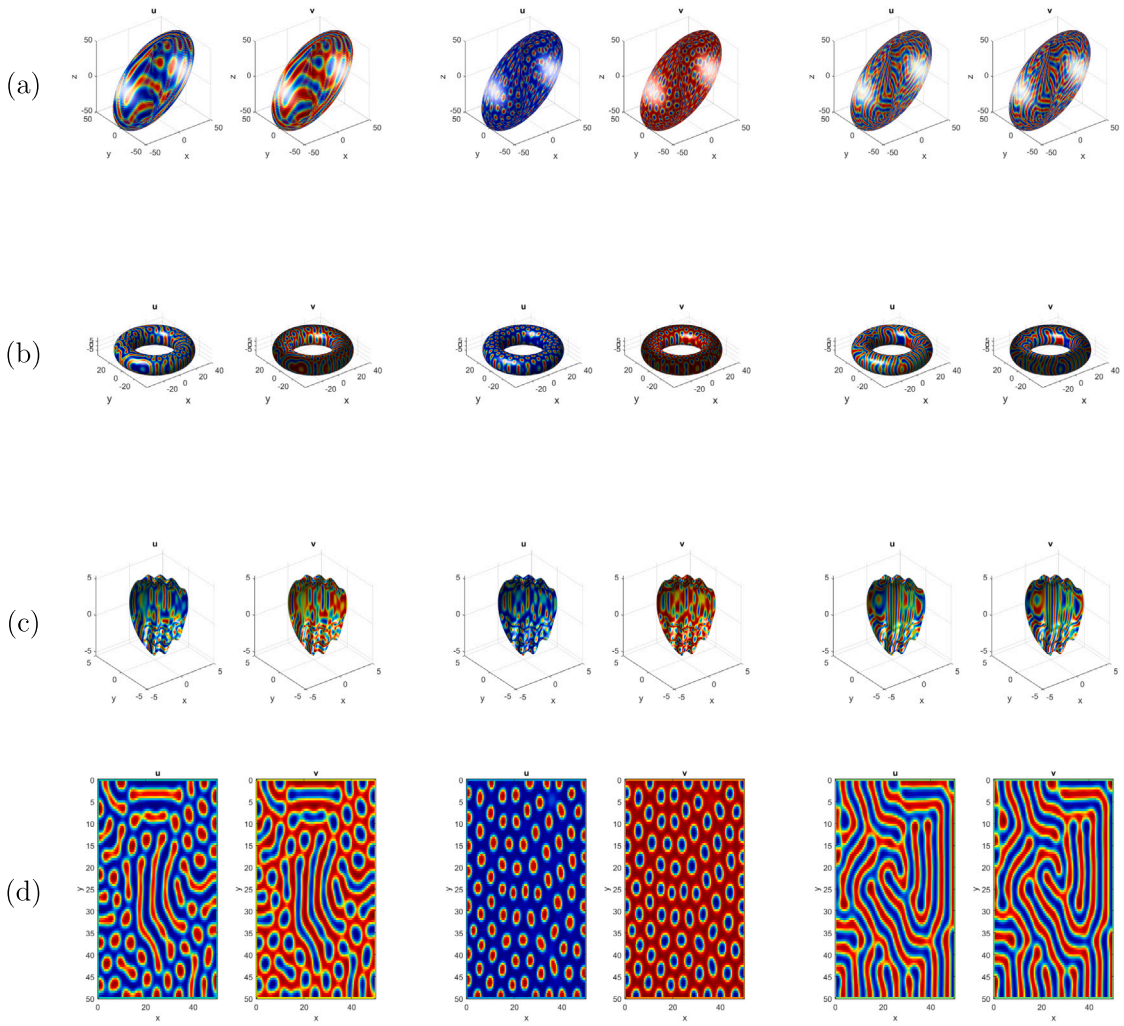

Fig. 6. Spatiotemporal distributions showing some Turing patterns on different surfaces for Gray-Scott model. Plots (a-d) correspond to sphere-, torus-, bumpy- and plane- surfaces, respectively. While columns 1–3 stands for evolution of spots/stripes, spots and stripes patterns, respectively.

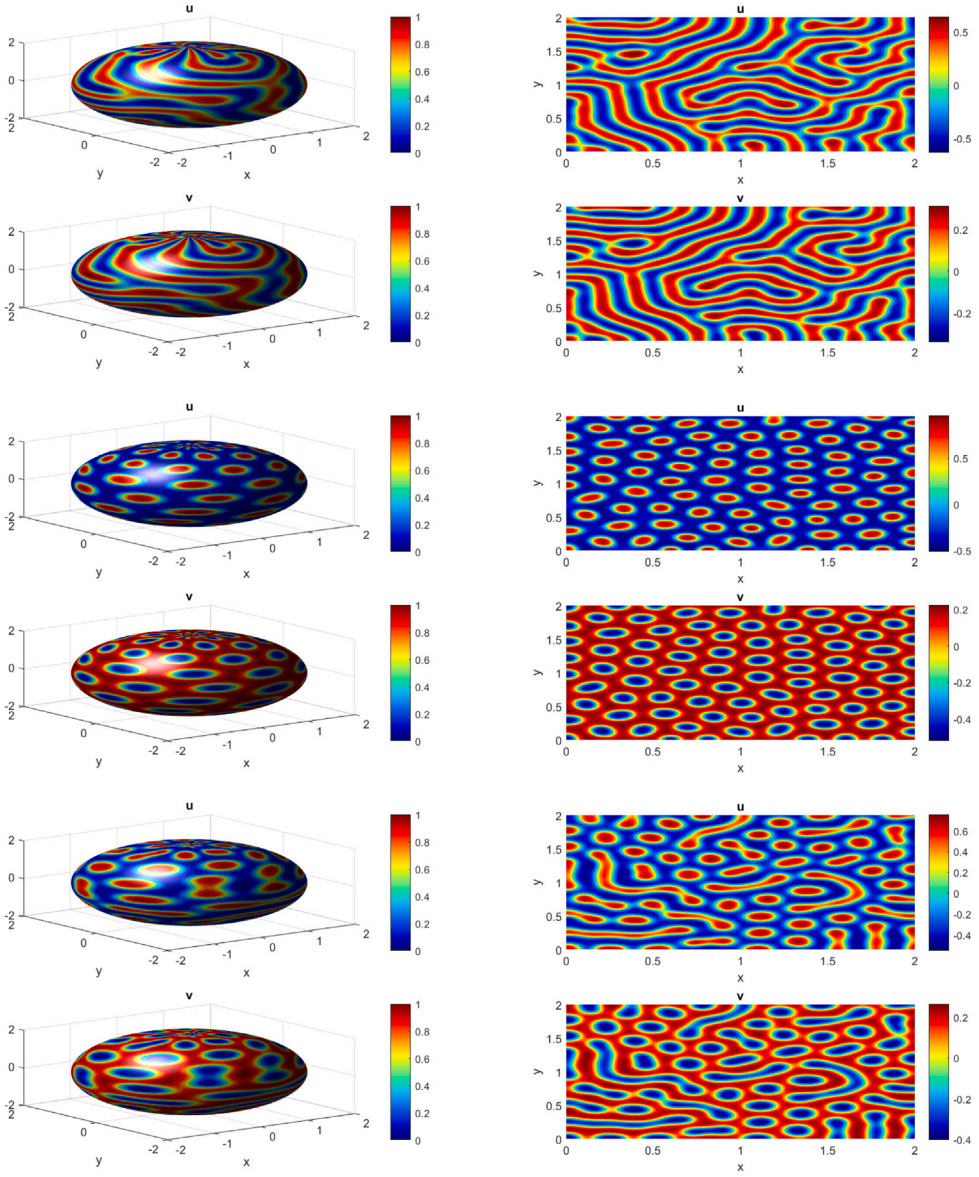

Fig. 7. Pattern formation in Brusselator model on 3D sphere (left) and 2D plane (right) surfaces with $\alpha = 0.01$, $\beta = 2$ and $D_u = 1$. Rows 1–3 correspond to $D_v = [8(\text{stripes}), 90(\text{spots}), 35(\text{stripes and spots})]$, respectively.

6.2. Test problems

6.2.1. Gray–Scott model

The Gray-Scott model (2.5) describes an autocatalytic reaction–diffusion system as reported in [4]. We simulate for spot, stripes and intermittent patterns using the Gray-Scott model with parameters from some certain regimes. The initial condition is a small square perturbation in the center of the domain:

$$(u, v) = \begin{cases} (1, 0) & \text{outside the square} \\ (0.5, 0.25) & \text{inside the square} \end{cases} \tag{6.39}$$

The Gray-Scott model exhibits a rich variety of patterns depending on the parameters. We focus on the following parameter regimes [4,10]:

- $F = 0.04$, $k = 0.06$ (spots), see Fig. 3, row 1.
- $F = 0.02$, $k = 0.05$ (spots and stripes), as shown in Fig. 3, row 2.

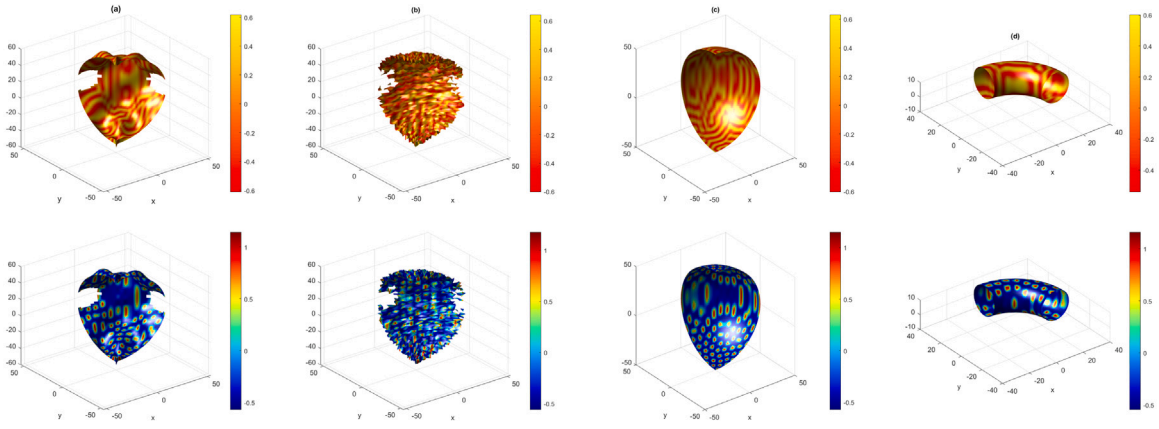

Fig. 8. Evolution of Turing patterns on some irregular surfaces for Brusellator model (2.6).

- $F = 0.03$, $k = 0.055$ (stripes), as displayed in see Fig. 3, row 3.

Stripe patterns are simulated using the Gray-Scott model with parameters from the γ regime. We use random initial conditions:

$$(u, v) = (1, 0) + \epsilon(x, y) \tag{6.40}$$

where $\epsilon(x, y)$ is a small random perturbation. For mixed and transitional patterns, we use the Gray-Scott model with parameters from the β regime. We also examine transitional patterns by slowly varying parameters from one regime to another during the simulation.

For all cases, we set $\mu_1 = 2 \times 10^{-5}$ and $\mu_2 = 10^{-5}$. We extend our simulation by considering pattern generation on the surface $\Gamma = \{(x, y, z) | (2 - \sqrt{x^2 + y^2})^2 + z^2 = 1\}$. The corresponding results on torus and bumpy sphere surfaces are given in Fig. 4, 5, and 6, respectively.

6.2.2. Brusselator model

The Brusselator model is another classical reaction–diffusion system that exhibits Turing instability when [1]:

$$\frac{D_v}{D_u} > \frac{\beta + 1}{\beta - \alpha^2} > 1. \tag{6.41}$$

We choose parameters that satisfy this condition: $\alpha = 3$, $\beta = 9$, $D_u = 1$, and allows $D_v \in [8, 100]$ to vary in the experiment. For this example, the Brusselator model (2.6) is considered on both plane and sphere

$$\Gamma = (x, y, z) : x = \tau \sin\theta \cos\phi, \quad y = \tau \sin\theta \sin\phi, \quad z = \tau \cos\phi, \quad 0 \le \theta \le 2\pi, \quad 0 \le \phi \le \pi,$$

where

$$\tau = 1 + \frac{\sin(6\theta)\sin(7\phi)}{7}.$$

Fig. 7 shows spatiotemporal Turing patterns obtained with different values of D_v as displayed in the caption. In Figs. 8 and 9, we experiment on surface

$$\Gamma = (x, y, z) : x = (R + r\cos\phi)\cos\theta, \quad y = (R + r\cos\phi)\sin\theta, \quad z = r\cos\phi,$$

for $\theta \in (0, 2\pi, N)$, $\phi \in (0, 2\pi, N)$, where R and r are the major and minor radius, respectively. The experiment is run for $N = 200$ to increase resolution for smoother shape.

6.3. BVAM model

For the third example, we consider the numerical simulation of the BVAM model (2.7) for spots, intermittent and stripe patterns on plane, sphere and bumpy surfaces. The parameters used are given in Table 4.

6.4. Spot pattern formation and multistability in Turing systems across geometries

Our simulations of the Gray-Scott, Brusselator, and BVAM models under homogeneous Neumann boundary conditions demonstrate the spontaneous emergence of spot-like Turing patterns in two-dimensional planar domains. A particularly striking observation is the presence of multistability—the system can evolve into distinct stable spatial patterns under the same parameter settings, depending only on variations in the initial perturbations. By altering the seed in the random initial conditions, we verified that

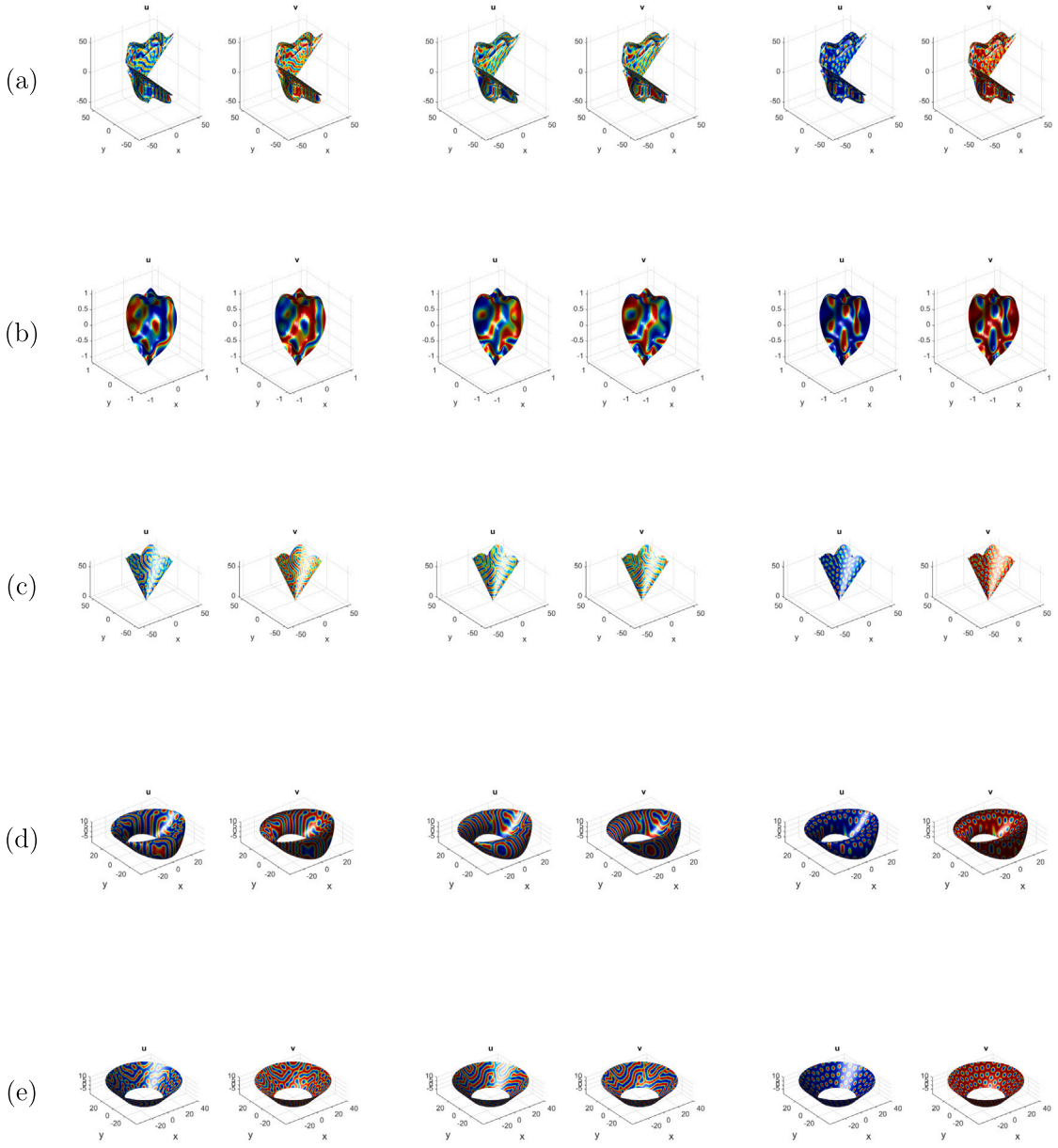

Fig. 9. Pattern formation with Brusellator model (2.6) on irregular surfaces.

Table 4
Parameters set studied in the BVAM model.

Pattern cases	Parameters	Value	Results
	D_u	0.416	
	ϕ	0.899	
	τ	0.0021	
	φ	−0.91	
Spots patterns	(α, β)	(3.5, 1.45), (0.2, 0.2)	Figs. 10, 13, 14(a)
Stripes patterns	(α, β)	(3.5, 0.00057), (3.5, 0)	Figs. 10, 11, 14(c)
Intermittent case	(α, β)	(3.5, 0.45), (0.25, [0, 0.25])	Figs. 10, 12, 14(b)

different spatial configurations may arise for identical parameters (as demonstrated in the figures). This behavior underscores the sensitivity of pattern selection to initial conditions and the presence of multiple attractors in the system's dynamical landscape.

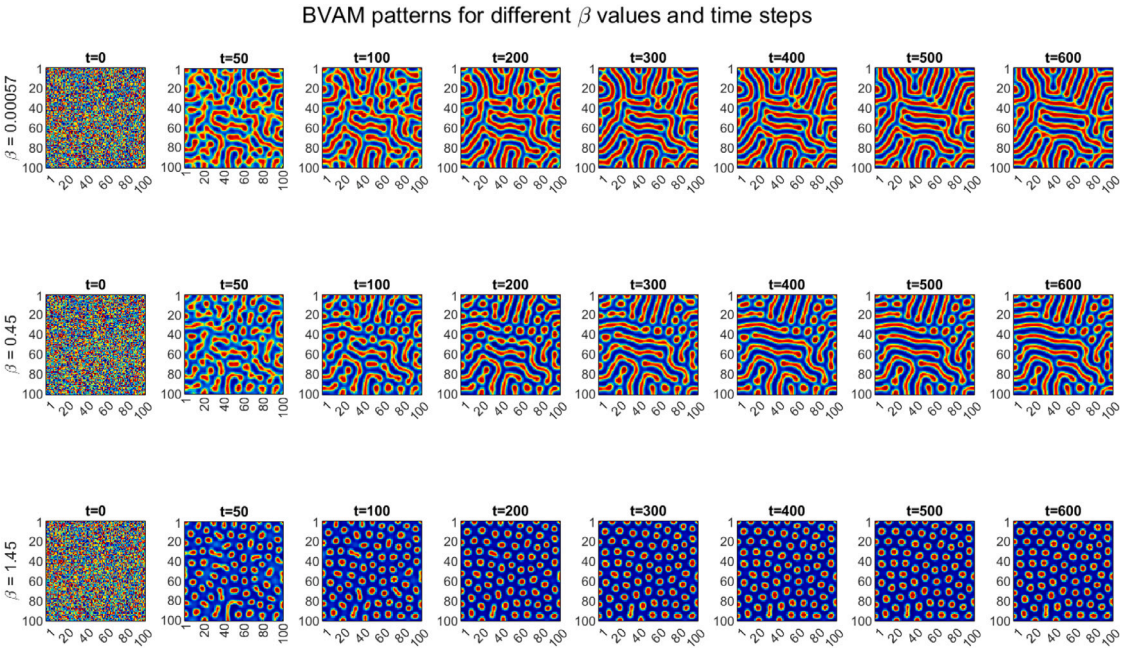

Fig. 10. BVAM model on plane surface for different parameters β and simulation time t.

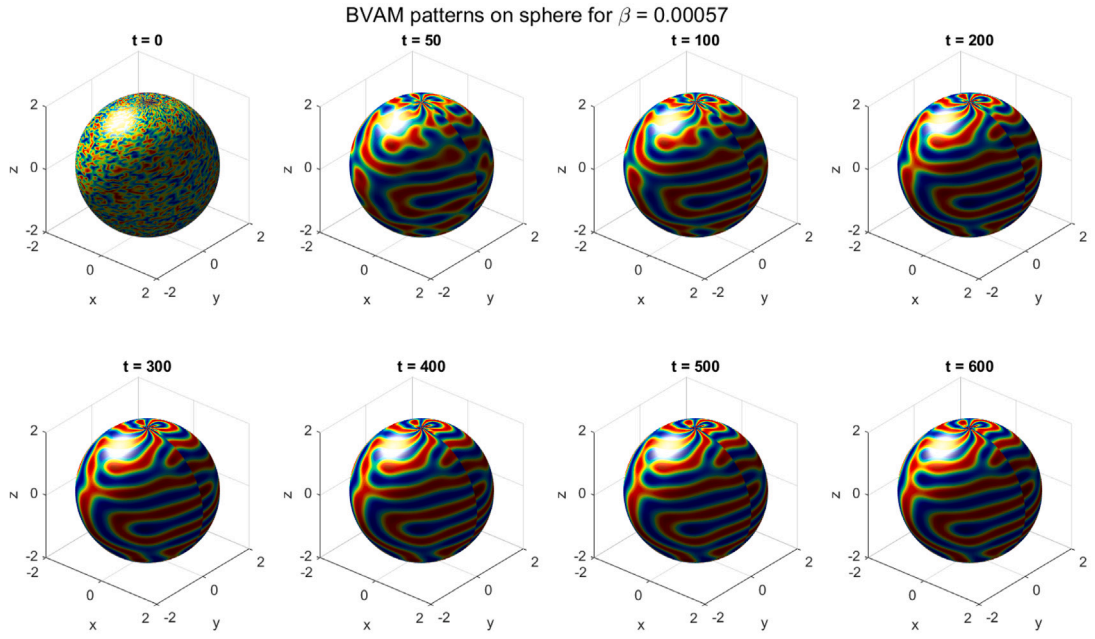

Fig. 11. BVAM model on sphere showing emergence of stripes pattern at $\beta = 0.00057$ for different t.

Similar multistable behavior was observed on curved surfaces such as the sphere and torus, though with notable geometric influences. On the sphere, the intrinsic curvature introduces a global constraint that tends to favor symmetric arrangements of spots, such as pentagonal or hexagonal tilings. On the torus, the topology allows for periodic wrapping of the domain, enabling more diverse and complex symmetries, including lattice-like and spiral arrangements that are less constrained by boundary conditions due to its boundary-less nature.

For bumpy surfaces non-uniformly curved geometries pattern formation becomes even more intricate. Local curvature variations can either attract or repel spots, leading to heterogeneous distributions that reflect the geometric modulation of the diffusion field.

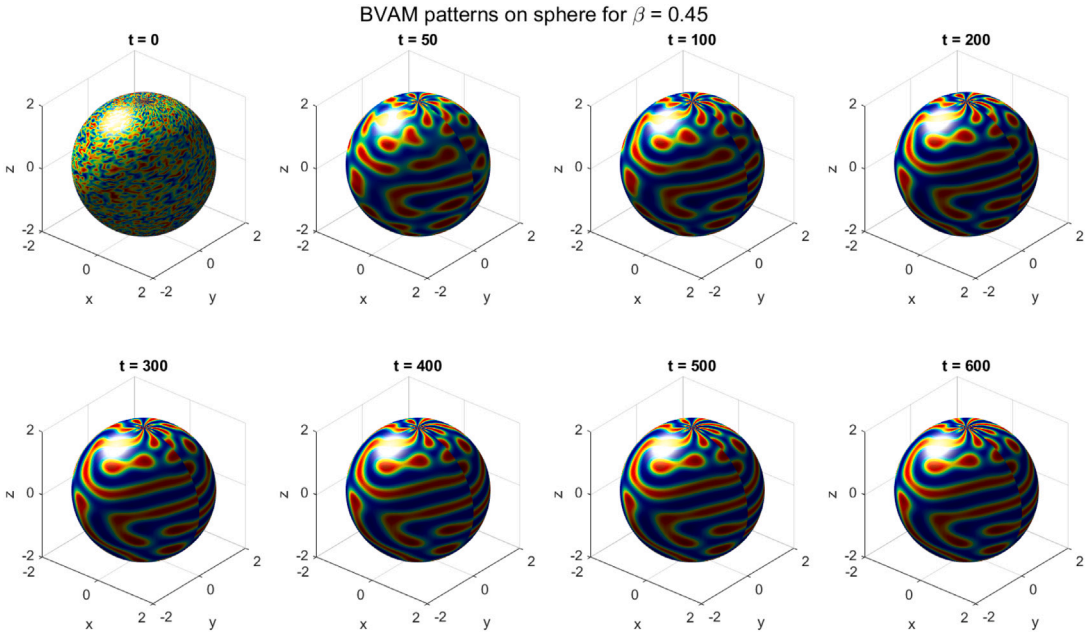

Fig. 12. Mixture of stripes and spots pattern for BVAM model at $\beta = 0.45$ for different t.

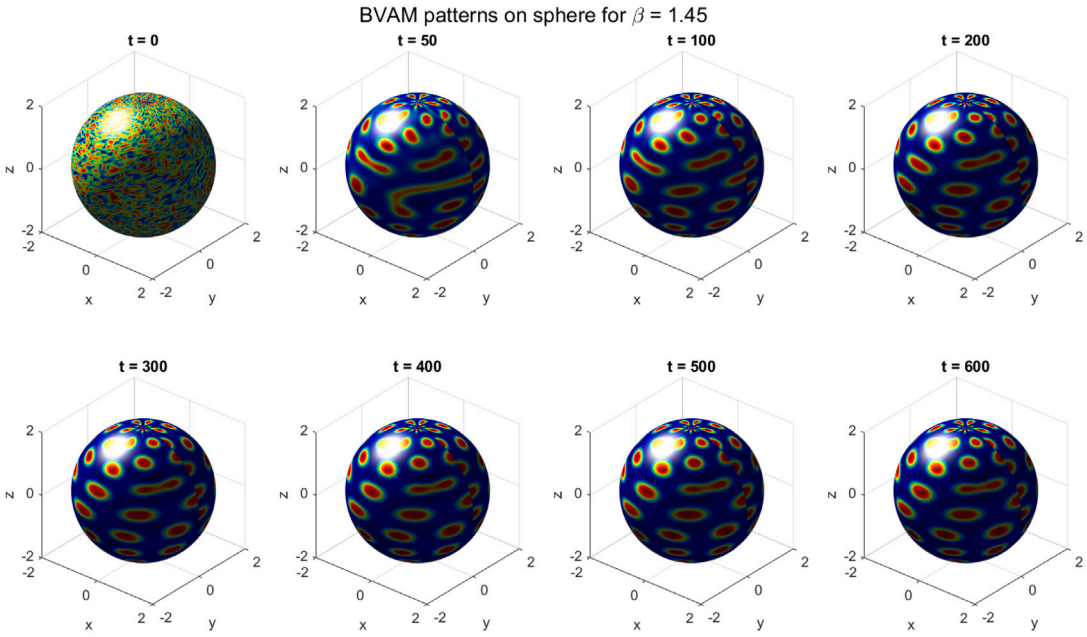

Fig. 13. Emergence of Turing spots for BVAM model at $\beta = 1.45$ for different t.

These geometries tend to disrupt global symmetries and promote more irregular spot arrangements, especially in the presence of nonlinear or anisotropic diffusion.

Boundary conditions further modulate these dynamics. When switching from homogeneous Neumann to homogeneous Dirichlet boundary conditions, we observed significant changes in the resulting patterns. On flat domains, Dirichlet conditions amplify the influence of boundary-imposed symmetries, particularly in normal diffusion regimes where morphogen concentration at the boundary propagates quickly toward the interior. This boundary influence is less effective under super-diffusive conditions, where faster spreading reduces sensitivity to specific boundary values, allowing intrinsic dynamics to dominate pattern formation.

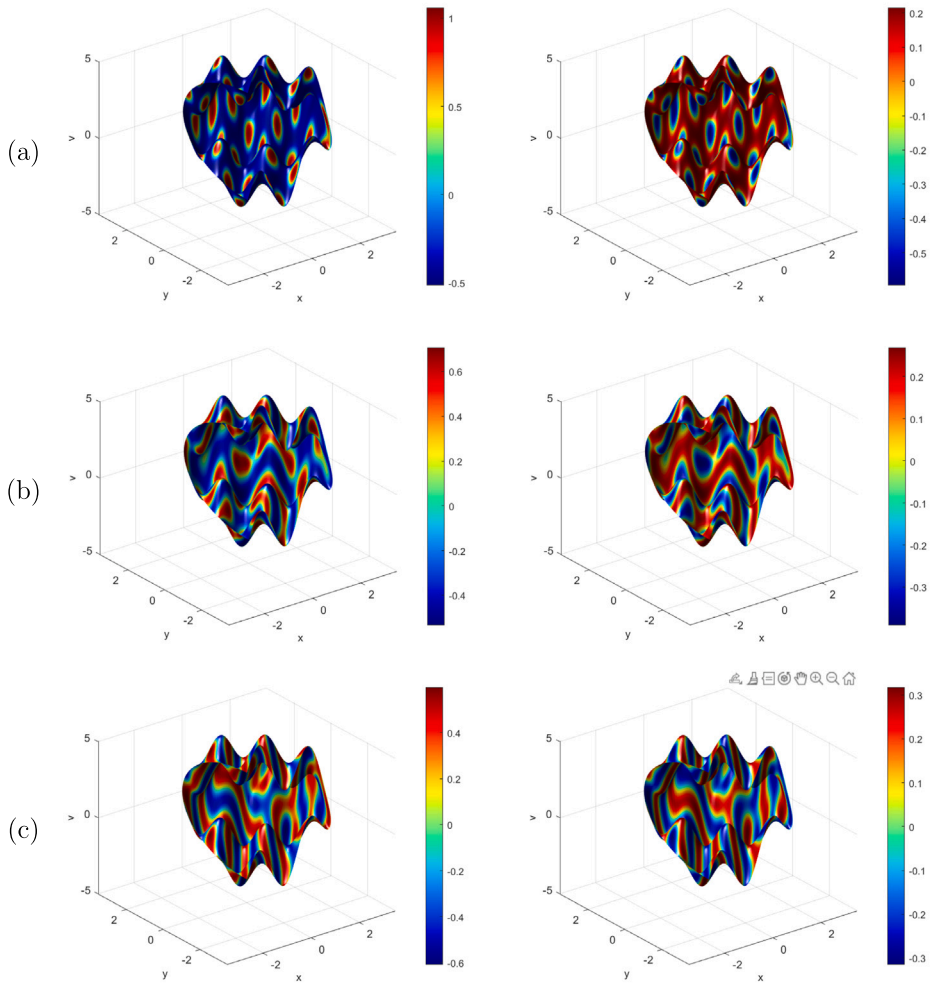

Fig. 14. Pattern formation in BVAM model on 3D irregular bumpy surface. plots (a) spots with $\beta = 0.00045$, (b) spots and stripes with $\beta = 0.45$ and (c) stripes with $\beta = 0.0005$.

The interplay between diffusion regime, initial conditions, boundary conditions, and surface geometry thus jointly governs the diversity, symmetry, and stability of spot patterns. For example: On the plane, multistability is pronounced and strongly influenced by both initial perturbations and boundary conditions. On the sphere, curvature tends to impose symmetrical constraints, often reducing the degree of multistability. On the torus, periodic boundary conditions allow for more flexible pattern symmetries and are conducive to multistable behavior. On bumpy surfaces, the local curvature interacts with diffusion dynamics to produce spatially heterogeneous, often irregular spot arrangements that may be less sensitive to global boundary conditions. These findings highlight the rich complexity of Turing systems in non-Euclidean domains and open pathways for exploring biological pattern formation on realistic anatomical surfaces.

6.4.1. Stripe pattern formation in reaction–diffusion systems on various geometries

Stripe patterns are a common type of spatial structure observed in reaction–diffusion systems, particularly in Turing-type models. Our investigation focused on three canonical systems – the Gray-Scott, Brusselator, and BVAM models – under varying parameter regimes and domain geometries, including flat planes, spheres, tori, and undulating (bumpy) surfaces.

Stripe formation in the Gray-Scott model

In the Gray-Scott model, stripe patterns emerge within specific regions of the parameter space, particularly as a function of the feed rate F and the kill rate k. Simulations revealed that: For moderate values of F and lower values of k, stationary stripe patterns can develop and remain stable. Increasing k tends to destabilize the stripes, leading to the formation of spot-like or mixed patterns, see Figs. 3, 4,5, and 6 captions for details. These results emphasize the sensitivity of the Gray-Scott model to even small parameter shifts, especially near bifurcation boundaries.

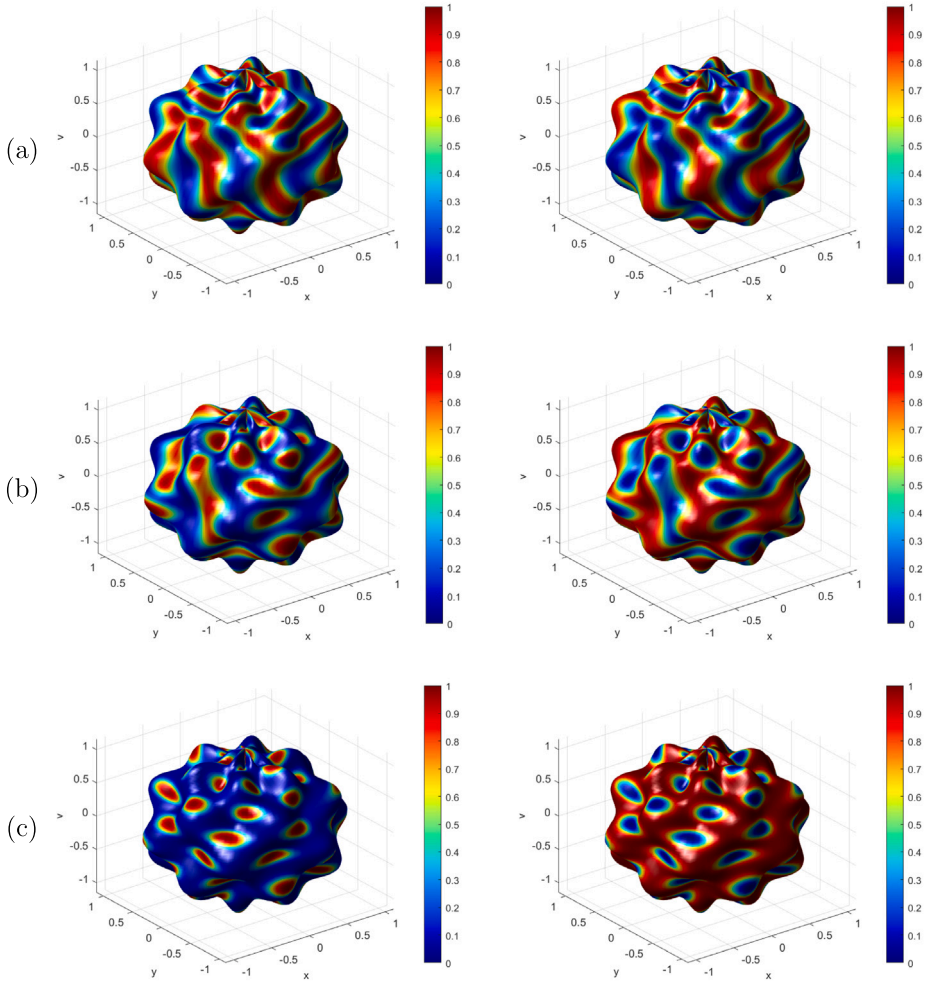

Fig. 15. Pattern formation in BVAM model on regular 3D bumpy surface. plots (a) stripes with $\beta = 0.0005$, (b) spots and stripes with $\beta = 0.45$ and (c) spots with $\beta = 0.00045$.

Stripe–spot transition in the Brusselator model

In the Brusselator model, the diffusion coefficient of the inhibitor species, D_v, plays a dominant role in pattern selection. Our findings can be summarized as follows:

When $D_v \in [8, 30]$, the system predominantly exhibits pure stripe patterns. For $30 \leq D_v \leq 69$, the system transitions to an *intermittent regime*, with coexisting spots and stripes. When $D_v > 70$, the system stabilizes into spot-dominated patterns, see Fig. 7, 8 and 9. This transition highlights the continuous deformation of pattern morphology as D_v increases, consistent with known diffusion-driven instability mechanisms.

Pattern selection in the BVAM model

The BVAM model exhibits similarly rich behavior depending on the parameter β: For instance, pure stripe patterns were observed for $\beta = 0.00057$. When $\beta \in [0.1, 0.5]$, see Figs. 10 top-row, 11, 14(c) and 15(a). The system yielded mixed or intermittent structures—stripes interspersed with isolated spots, as shown in Figs. 10 middle-row, 12, 14(b) and 15(b). For $\beta \geq 1$, the pattern fully transitions to stable spots. The BVAM model's parameter sensitivity offers a flexible framework for generating a variety of biologically relevant patterns by tuning a single bifurcation parameter.

Influence of initial and boundary conditions

In all three models, the type of initial perturbations and the choice of boundary conditions significantly influenced the final pattern morphology. For example: Random perturbations with different seeds led to variability in symmetry and alignment of stripes. Switching from homogeneous Neumann to Dirichlet boundary conditions introduced boundary-induced alignment of stripes, especially in normal diffusion regimes.

Table 5
Comparison of computational efficiency for different methods (Error tolerance: 10^{-6}).

Method	Grid size	Time step	CPU time (s)
DSC-ETDRK4	128	0.1	10.5
Spectral-ETDRK4	128	0.1	12.3
FD-RK4	512	0.01	45.7

Effect of geometry on stripe patterns

The spatial geometry of the domain further modulates the type and orientation of stripe patterns. Key observations include: Planar Domain: Stripe orientation is typically influenced by initial conditions and boundary shape. In rectangular domains, stripes often align parallel or perpendicular to the boundaries. Spherical Surface: The curvature of the sphere imposes a global constraint that often disrupts long, linear stripe formation. Instead, closed-loop stripes or symmetric banding patterns tend to appear, often organized by the spherical harmonics of the system. Torus Geometry: The torus supports periodic stripe patterns with translational symmetry. Its topological periodicity allows stripes to wrap around without edge effects, encouraging stable lattice-like or spiral stripe configurations. Bumpy or Undulating Surfaces: Local curvature variations strongly affect pattern localization. Stripes may become distorted or pinched near regions of high curvature. These surfaces often induce heterogeneous patterning, with local geometric features acting as attractors or inhibitors for stripe formation.

The stripe pattern formation in reaction–diffusion models is a multifaceted phenomenon governed by a confluence of factors: parameter values, diffusion ratios, initial perturbations, boundary conditions, and the underlying geometry of the domain. Understanding these interactions not only enriches the mathematical theory of pattern formation but also provides deeper insights into morphogenetic processes in biological and synthetic systems.

7. Results and discussion

7.1. Accuracy and efficiency of the combined approach

The combined DSC-ETDRK4 approach demonstrates excellent accuracy and efficiency in solving reaction–diffusion systems. Fig. 2 and Table 1 show the spatial convergence of the method for the Gray-Scott model. As evident from Tables 2 and 3, the DSC method achieves spectral accuracy, with errors decreasing exponentially as the number of grid points increases. This is consistent with the theoretical error bounds derived by [25]. For a grid size of $N = 128$, the DSC method achieves an accuracy of 10^{-10}, comparable to the spectral method and significantly better than the second-order finite difference method. Our results demonstrate that the ETDRK4 scheme maintains fourth-order accuracy even for large time steps, outperforming the classical RK4 method. This is particularly advantageous for stiff problems, where the RK4 method requires extremely small time steps for stability [18].

In terms of computational efficiency, Table 5 compares the CPU time required by different methods to achieve a given accuracy.

As shown in Table 5, the DSC-ETDRK4 method achieves the target accuracy with significantly less computational cost than the finite difference method, and is comparable to the spectral method. This efficiency stems from the high accuracy of the DSC spatial discretization, which allows for larger grid spacing, and the stability of the ETDRK4 scheme, which permits larger time steps [9].

7.2. Ability to capture complex spatiotemporal patterns

One of the key strengths of the DSC-ETDRK4 method is its ability to accurately capture complex spatiotemporal patterns in reaction–diffusion systems. Figs. 3–14 show simulations of spot, stripe, and intermittent (mixed) patterns in the Gray-Scott, Brusselator and BVAM models for different parameter values on plane, sphere, bumpy and torus surfaces.

These simulations demonstrate that the DSC-ETDRK4 method can accurately reproduce the full range of patterns observed in experimental studies of the reaction–diffusion system [4,10,26]. The method captures fine details of the patterns, including the curvature of spot boundaries and the intricate structures in mixed patterns.

For the Brusselator model, experimental results show the formation of Turing patterns from small random perturbations on different regular and irregular surfaces. The simulations in Fig. 7, 8, and 9 accurately capture the characteristic length scale and symmetry of Turing patterns, consistent with theoretical predictions (Murray, 2003). The method resolves both the initial linear growth phase and the subsequent nonlinear pattern selection processes.

To further demonstrate the method's capabilities, we simulate a complex spatiotemporal phenomenon: spots, stripes, spots/stripes wave formation in the BVAM model on bumpy, sphere surfaces, as shown in Figs. 10, 11, 12, 13, 14 and 15 over time. The simulation results in figures accurately capture the initiation, rotation, and steady-state behavior of the chemical species. The method resolves the sharp wave fronts without numerical oscillations, a challenging task for many numerical schemes (Barkley, 1991). In all these simulations, the DSC-ETDRK4 method demonstrates high accuracy and stability, even for long simulation times and complex dynamics. The method's ability to capture a wide range of spatiotemporal patterns with high fidelity makes it a powerful tool for studying pattern formation in reaction–diffusion systems.

The performance of the DSC-ETDRK4 method in these diverse scenarios can be attributed to several factors: (i) The high spatial accuracy of the DSC method allows for accurate representation of complex spatial structures with relatively few grid points (Wei, 2000). (ii) The stability and efficiency of the ETDRK4 scheme enable accurate time integration over long periods, even for stiff

problems (Cox and Matthews, 2002). (iii) The method's ability to handle nonlinear terms effectively captures the crucial nonlinear interactions in reaction–diffusion systems (Kassam and Trefethen, 2005), and (iv) the flexibility of the DSC approach in handling different boundary conditions makes it suitable for a wide range of problem setups (Wei et al. 2002).

Hence, the combined DSC-ETDRK4 approach proves to be a powerful and versatile method for solving reaction–diffusion equations. Its high accuracy, efficiency, and ability to capture complex spatiotemporal patterns make it a valuable tool for studying a wide range of phenomena in mathematical biology, chemistry, and physics.

8. Conclusion

In this paper, we presented a combined approach using the Discrete Singular Convolution (DSC) method for spatial discretization and the Exponential Time Differencing Runge–Kutta (ETDRK4) scheme for temporal integration to solve reaction–diffusion systems. Focusing on the Gray-Scott, Brusselator and BVAM models, our numerical experiments demonstrate the high accuracy, efficiency, and versatility of this method in capturing complex spatiotemporal patterns.

Efficient implementation of the DSC-ETDRK4 method requires several considerations. The sparse DSC differentiation matrices allow for precomputation and efficient sparse matrix operations [5]. Matrix exponentials and φ-functions, crucial for performance, benefit from Carathéodory-Fejér approximation or contour integral methods [20]. For periodic boundary conditions, Fast Fourier Transforms (FFTs) accelerate the DSC method [7]. The method's parallelizability, especially in computing nonlinear terms $N(U)$, enhances efficiency [22]. Adaptive time-stepping strategies dynamically improve efficiency for problems with varying time scales [21,27]. Careful memory management is essential for large-scale problems [23]. The DSC-ETDRK4 method provides high spatial and temporal accuracy, making it suitable for reaction–diffusion systems with complex patterns and stiff problems in mathematical biology and chemical kinetics.

Our study's key findings include the DSC method's spectral-like accuracy in spatial discretization, the ETDRK4 scheme's stability and accuracy for stiff problems, and the combined approach's excellent stability and conservation properties. The method accurately captures various spatiotemporal patterns, aligning well with theoretical predictions and experimental observations. The DSC-ETDRK4 method is computationally efficient, often outperforming other methods such as spectral and finite difference approaches.

Future work could enhance the method's capabilities and applicability by extending it to three-dimensional systems, incorporating adaptive mesh refinement, optimizing performance for complex geometries, developing parallel implementations and GPU acceleration, applying it to more complex reaction–diffusion systems, handling stochastic differential equations, and creating a user-friendly software package. In conclusion, the combined DSC-ETDRK4 approach offers a powerful and flexible tool for solving reaction–diffusion equations, providing high accuracy, efficiency, and the ability to capture complex spatiotemporal patterns. As we refine and extend this method, it promises to significantly contribute to understanding complex systems in biology, chemistry, and physics.

CRediT authorship contribution statement

Kolade M. Owolabi: Conceptualization, Writing – original draft, Software, Investigation. **Edson Pindza:** Writing – review & editing, Validation, Supervision, Formal analysis. **Eben Maré:** Writing – review & editing, Validation, Software.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

All authors approve review and approve the final version.

Data availability

No data was used for the research described in the article.

References

[1] Murray JD. Mathematical biology II: spatial models and biomedical applications. Berlin: Springer-Verlag; 2003.
[2] Owolabi KM, Jain S. Spatial patterns through diffusion-driven instability in modified predator–prey models with chaotic behaviors. Chaos Solitons Fractals 2023;174:113839.
[3] Turing AM. The chemical basis of morphogenesis. Philos Trans R Soc Lond Ser B Biol Sci 1952;237:37–72.
[4] Pearson JE. Complex patterns in a simple system. Science 1993;261:189–92.
[5] Wei GW. A new algorithm for solving some mechanical problems. Comput Methods Appl Mech Engrg 2001;190:2017–30.
[6] Owolabi KM. Analysis and numerical simulation of cross reaction–diffusion systems with the Caputo–Fabrizio and Riesz operators. Numer Methods Partial Differential Equations 2023;39:1915–37.
[7] Yang SY, Zhou YC, Wei GW. Comparison of the discrete singular convolution algorithm and the Fourier pseudospectral method for solving partial differential equations. Comput Phys Comm 2002;143:113–35.

[8] Wei GW. Discrete singular convolution for the solution of the Fokker–Planck equation. J Chem Phys 1999;110:8930–42.
[9] Cox SM, Matthews PC. Exponential time differencing for stiff systems. J Comput Phys 2002;176:430–55.
[10] Mazin W, Rasmussen KE, Mosekilde E, Borckmans P, Dewel G. Pattern formation in the bistable Gray-Scott model. Math Comput Simulation 1996;40:371–96.
[11] Prigogine I, Lefever R. Symmetry breaking instabilities in dissipative systems. II. J Chem Phys 1968;48:1695–700.
[12] Wei GW. Wavelets generated by using discrete singular convolution kernels. J Phys A: Math Gen 2000;33:8577.
[13] Mar'e E, Mba JC, Pindza E. Discrete singular convolution for the generalized variable-coefficient Korteweg-de Vries equation. Quaest Math 2017;40:225–44.
[14] Pindza E, Mar'e E. Discrete singular convolution method for free vibration analysis annular membranes. Math Comput Appl 2013;14:131–8.
[15] Qian L. On the regularized Whittaker-Kotel'nikov-Shannon sampling formula. In: Proceedings of the American mathematical society, vol. 131, 2003, p. 1169–76.
[16] Hochbruck M, Ostermann A. Exponential integrators. Acta Numer 2010;19:209–86.
[17] Hochbruck M, Lubich C, Selhofer H. Exponential integrators for large systems of differential equations. SIAM J Sci Comput 1998;19:1552–74.
[18] Kassam A, Trefethen LN. Fourth-order time-stepping for stiff PDEs. SIAM J Sci Comput 2005;26:1214–33.
[19] Hochbruck M, Ostermann A. Explicit exponential Runge-Kutta methods for semilinear parabolic problems. SIAM J Numer Anal 2005;43:1069–90.
[20] Schmelzer T, Trefethen LN. Evaluating matrix functions for exponential integrators via Carathéodory-Fejér approximation and contour integrals. Electron Trans Numer Anal 2007;29:1–18.
[21] Berl H, Skaflestad B, Wright WM. EXPINT—A MATLAB package for exponential integrators. ACM Trans Math Softw (TOMS) 2007;33(1):4.
[22] Luan VT, Ostermann A. Exponential B-series: The stiff case. SIAM J Numer Anal 2013;51:3431–45.
[23] Jawecki T, Polner M, van Veldhuizen M. Efficient implementation of exponential time differencing methods for nonlinear parabolic problems. 2022, arXiv preprint arXiv:2201.05240.
[24] Hundsdorfer W, Verwer JG. Numerical solution of time-dependent advection-diffusion-reaction equations. Springer-Verlag Berlin; 2003.
[25] Qian L, Wei GW. A note on the truncation error of least squares kernels. Appl Math Lett 2000;13:97–102.
[26] Alejandro Valdés Lópeza, Hernándezb D, Carlos Aguilar-Maderac G, Roxana Cortés Martíneza, Herrera-Hernández EC. Boundary conditions influence on turing patterns under anomalous diffusion: A numerical exploration. Phys D 2024;470:134353.
[27] Barkley D. A model for fast computer simulation of waves in excitable media. Phys D: Nonlinear Phenom 1991;49:61–70.