Mathematical Modelling and Numerical Simulation with Applications, 2023, 3(4), 297–317

https://dergipark.org.tr/en/pub/mmnsa

ISSN Online: 2791-8564 / Open Access

https://doi.org/10.53391/mmnsa.1387125

RESEARCH PAPER

An enhanced SUPG-stabilized finite element formulation for simulating natural phenomena governed by coupled system of reaction-convection-diffusion equations

Süleyman Cengizci [1,2,*,‡]

[1]Computer Programming, Antalya Bilim University, Antalya 07190, Türkiye, [2]Department of Business Administration, Antalya Bilim University, Antalya 07190, Türkiye

[*]Corresponding Author

[‡]suleyman.cengizci@antalya.edu.tr (Süleyman Cengizci)

Abstract

Many phenomena arising in nature, science, and industry can be modeled by a coupled system of reaction-convection-diffusion (RCD) equations. Unfortunately, obtaining analytical solutions to RCD systems is typically not possible and, therefore, usually requires the use of numerical methods. On the other hand, since solutions to RCD-type equations can exhibit rapid changes and may have boundary/inner layers, classical computational tools yield approximations polluted with physically meaningless oscillations when convection dominates the transport process. Towards that end, in order to eliminate such numerical instabilities without sacrificing accuracy, this work employs a stabilized finite element formulation, the so-called streamline-upwind/Petrov–Galerkin (SUPG) method. The SUPG-stabilized formulation is then also supplemented with the YZβ shock-capturing mechanism to achieve higher-quality approximations around sharp gradients. A comprehensive set of numerical test experiments, including cross-diffusion systems, the Schnakenberg reaction model, and mussel-algae interactions, is considered to reveal the robustness of the proposed formulation, which we call the SUPG-YZβ formulation. Comparisons with reported studies reveal that the proposed formulation performs quite well without introducing excessive numerical dissipation.

Keywords: Reaction-convection-diffusion; finite elements; stabilization; shock-capturing; SUPG-YZβ formulation

AMS 2020 Classification: 35G61; 65M60; 76M10

1 Introduction

Reaction-convection-diffusion (RCD) equations are used to model a wide range of natural phenomena in addition to many industrial and engineering applications. Some of these applications

➤ **Received:** 06.11.2023 ➤ **Revised:** 08.12.2023 ➤ **Accepted:** 09.12.2023 ➤ **Published:** 30.12.2023

include financial engineering (e.g., Black–Scholes and Heston option pricing models), chemistry (e.g., chemically reactive transport phenomena), semiconductor theory (e.g., drift-diffusion equations), fluid dynamics (e.g., Burgers'-type and Navier–Stokes equations), heat transfer (e.g., natural heat convection phenomena), and mathematical physics and astrophysics (e.g., Fokker–Planck-type equations). The coupled systems consisting of RCD-type equations are also essential for modeling many phenomena that involve interactions between more than one species and frequently arise in biological and chemical sciences, such as tumor growth models, chemotaxis processes, bacteria pattern formation, predator-prey dynamics, etc. We refer the interested reader to the extensive work of Painter [1] and Bellomo et al. [2] for chemotaxis and cross-diffusion models and their applications in biology, physiology and pathology, ecology, and even in the social sciences (e.g., crime hotspot models). The review papers [3] and [4] can also be referred to for more on pattern formation phenomena arising in plasma physics and the influence of temperature on such systems, respectively.

Analytical solutions to RCD-type systems are generally impossible to obtain since they are typically of a nonlinear nature and/or defined on sophisticated domains. Therefore, numerical approximations to the solutions of such systems are searched for. Unfortunately, despite the availability of several classical and mature numerical methods with solid theoretical foundations and sharp error estimates, such as the finite difference method (FDM), finite volume method (FVM), and finite element method (FEM), these methods are insufficient to provide accurate approximations to the solutions of RCD-type equations and coupled systems composed of such equations in convection dominance, leading to spurious oscillations. In order to overcome such numerical instability issues, the above-mentioned classical methods have been enhanced with several techniques over the years. The following paragraph presents a very concise overview of reported studies dedicated to solving coupled systems of RCD-type equations numerically. For a more comprehensive overview, the material in these references can also be referred to.

The authors of [5] investigated the effect of advection on coupled systems of reaction-diffusion (RD) equations, more specifically, the Schnackenberg and glycolysis reaction kinetics models having toroidal velocity fields, by employing the classical finite element method. Sarra considered unsteady RCD-type partial differential equations (PDEs) by employing a local radial basis function (RBF) method in [6]. The author particularly focused on chemotaxis models and Turing systems defined on complex-shaped domains. In [7], the authors proposed positivity-preserving nonstandard finite difference schemes for cross-diffusion models arising in biosciences, including malignant invasion, convective predator-prey pursuit and evasion model, and reaction-diffusion-chemotaxis model. Yücel et al. [8] studied optimal control problems governed by a system of convection-dominated RCD-type PDEs by employing a discontinuous GFEM (dGFEM) formulation. They used a symmetric interior penalty Galerkin (SIPG) discretization for the diffusion term and an upwinding discretization for the convection term, along with an adaptive mesh refinement algorithm. The author of [9] used a meshless finite difference method equipped with B-splines for solving time-dependent RD- and RCD-type coupled systems, including tumor invasion models and cross-diffusion problems. Wang et al. [10] studied the dynamics and pattern formation of a coupled time-dependent RCD system defined on a one-dimensional (1D) domain for modeling the interaction of mussels and algae. Most recently, two-dimensional (2D) elliptic-type singularly perturbed weakly-coupled systems of RCD equations, in which the diffusion and convection terms are controlled by two different parameters, were studied by Clavero et al. [11]. They proposed a first-order uniformly convergent finite difference scheme defined on layer-adapted Bakhvalov–Shishkin meshes. One can also refer to [12–14] and references therein for several applications of scalar and coupled RCD-type PDEs arising in chemical processes. Finally, in the context of

fractional differential equations, the studies [15–17] and the material therein can be referred to.

In the finite element framework, among the others, one of the most established, robust, and popular stabilized methods is the streamline-upwind/Petrov–Galerkin (SUPG) formulation. The method was first introduced for advection-diffusion equations and incompressible flow simulations by Hughes and Brooks [18, 19]. Following that, the compressible-flow SUPG method was introduced by Tezduyar and Hughes [20–22] in the context of conservation variables. The compressible-flow SUPG method introduced in 1982 is today denoted by "$(\text{SUPG})_{82}$." The $(\text{SUPG})_{82}$ formulation, in its initial form, was used without making use of any discontinuity-capturing (also commonly referred to as shock-capturing) mechanism. The test simulations demonstrated that regions with steep gradients require extra treatment. Then, the $(\text{SUPG})_{82}$ formulation was subsequently reformulated in terms of the entropy variables and equipped with a shock-capturing mechanism in [23], and more satisfactory results were obtained. In [24], the $(\text{SUPG})_{82}$ formulation was supplemented with a shock-capturing operator quite similar to the one introduced in [23] by Hughes et al., and the added term included a shock-capturing parameter, which is today called "δ_{91}." The set of stabilization parameters, which is almost universally denoted by "τ," used with the $(\text{SUPG})_{82}$ formulation introduced in [20–22] are called "τ_{82}" today. The SUPG-stabilized formulation for the reaction-advection-diffusion equation introduced in [25] included a shock-capturing term and a stabilization parameter that took into account the interaction between the shock-capturing and SUPG stabilization terms. Thus, the effect of the shock-capturing mechanism does not increase that of the SUPG stabilization when the advection and shock directions coincide. In [24], the definition of stabilization parameter τ_{82} was slightly modified by Le Beau et al. On the other hand, although the definition of $(\text{SUPG})_{82}$ parameters underwent some minor modifications in subsequent years, they were still used with the same shock-capturing parameter, δ_{91}, until 2004. Eventually, in 2004, several new ways of determining the stabilization and shock-capturing parameters in the $(\text{SUPG})_{82}$ framework were introduced in [26, 27] by Tezduyar. These new stabilization parameters are today referred to as "τ_{04}." As to the shock-capturing parameters, the new strategies introduced can be divided into two categories: the discontinuity-capturing directional dissipation (DCDD) [26, 28, 29] and the residual-based YZβ shock-capturing [26, 27]. Throughout this paper, we restrict our attention to the YZβ mechanism. Some of the reasons for adopting it include that it is easier to calculate the YZβ shock-capturing parameter than δ_{91}, the parameter β offers options for mild and sharp shocks, and as it was also reported in [30–32], the YZβ parameter yields more accurate results than δ_{91}. One can find various applications of the SUPG-YZβ combination, including arterial drug delivery, shallow-water equations, chemically reactive models, and natural convection heat transfer, in [12, 33–37]. For other stabilized formulation and shock-capturing mechanisms, we refer the interested reader to [38–40] and the material in these studies. Besides that, in [41–43], the interested reader can find various applications of Petrov–Galerkin-like methods.

In this paper, we deal with stabilized finite element computations of coupled systems of RCD-type equations. In doing this, we first consider the test problems as they were reported in the literature in order to make comparisons. Following that, whenever possible, each problem is considered for convection dominance, i.e., for much more computationally challenging cases, for which the classical methods fail to yield oscillation-free approximations and/or are insufficient to capture steep gradients. Thus, new challenging benchmark problems are introduced to the literature. The main computational method we use is the SUPG finite element formulation. We also augment the SUPG-stabilized formulation with the YZβ shock-capturing technique. To the author's best knowledge, this is the first report employing the SUPG-YZβ combination for handling such kinds of problems. The semi-discrete formulations are discretized in time with the backward Euler

scheme. Then, nonlinear equation systems arising from the space-time-discretized formulations are solved with the Newton–Raphson (N–R) algorithm, and the resulting linear systems are handled with a direct method, i.e., the lower-upper (LU) factorization technique.

The rest of the manuscript is organized as follows. In Section 2, a system of coupled 2D RCD-type PDEs is introduced as a model problem, and a semi-discrete GFEM formulation is described. In Section 3, a SUPG-stabilized finite element formulation combined with YZβ shock-capturing is introduced for the model problem. Section 4 first focuses on further computational details, such as the temporal discretization of the semi-discrete formulation, the quadrature degree associated with the numerical integration, and the computing environment FEniCS, in which the solvers are developed and computations are carried out. Later on in this section, four main numerical experiments with various scenarios are presented. Finally, in Section 5, some concluding remarks are made, along with a brief discussion on possible extensions of this current work.

2 Model problem and classical GFEM formulation

Let us consider the following coupled system of time-dependent RCD equations:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{a} \cdot \nabla \mathbf{u} - \nabla \cdot (\mathbf{D} \nabla \mathbf{u}) - \mathbf{f}(\mathbf{u}) = \mathbf{s}, \tag{1}$$

where the vector of unknowns, $\mathbf{u}$, is defined by $\mathbf{u} = [u_1, u_2]^T$, $\mathbf{a} = [a_1, a_2]$ is the velocity field associated with advection, and $\mathbf{D}$ represents the diffusivity matrix, which is given as:

$$\mathbf{D} = \begin{bmatrix} \epsilon_1 & 0 \\ 0 & \epsilon_2 \end{bmatrix}. \tag{2}$$

Here, the diffusion parameters ϵ_1 and ϵ_2 are non-negative and typically small. The vector $\mathbf{f}$ represents the reaction term, which is typically of nonlinear nature, and $\mathbf{s}$ is the source vector. For the moment, the system is assumed to be equipped with an appropriate set of initial and boundary conditions.

By multiplying both sides of system (1) by a test vector $\mathbf{w} \in \boldsymbol{\mathcal{V}}_{\mathbf{u}} \subset \boldsymbol{\mathcal{H}}_0^1$, the classical GFEM formulation can be obtained as follows:

$$\begin{cases} \text{find } \mathbf{u} \in \boldsymbol{\mathcal{S}}_{\mathbf{u}} \text{ such that } \forall \mathbf{w} \in \boldsymbol{\mathcal{V}}_{\mathbf{u}} : \\ \displaystyle\int_\Omega \mathbf{w} \cdot \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{a} \cdot \nabla \mathbf{u} - \nabla \cdot (\mathbf{D} \nabla \mathbf{u}) - \mathbf{f}(\mathbf{u}) - \mathbf{s} \right) d\Omega = 0, \end{cases} \tag{3}$$

where $\mathbf{u} \in \boldsymbol{\mathcal{S}}_{\mathbf{u}} \subset \boldsymbol{\mathcal{H}}_g^1$ is the solution vector, and the spaces $\boldsymbol{\mathcal{S}}_{\mathbf{u}}$ and $\boldsymbol{\mathcal{V}}_{\mathbf{u}}$ are the trial and test function spaces, respectively. The Sobolev spaces $\boldsymbol{\mathcal{H}}_0^1$ and $\boldsymbol{\mathcal{H}}_g^1$ are defined as follows:

$$\boldsymbol{\mathcal{H}}_0^1 = \left\{ \boldsymbol{\Phi} : \boldsymbol{\Phi} \in \left[\mathcal{H}^1(\Omega) \right]^2 \text{ and } \boldsymbol{\Phi}|_{\partial\Omega} = \mathbf{0} \right\}, \tag{4}$$

$$\boldsymbol{\mathcal{H}}_g^1 = \left\{ \boldsymbol{\Phi} : \boldsymbol{\Phi} \in \left[\mathcal{H}^1(\Omega) \right]^2 \text{ and } \boldsymbol{\Phi}|_{\partial\Omega} = \mathbf{g} \right\}, \tag{5}$$

where

$$\mathcal{H}^1 = \left\{ \Phi : \|\Phi\|_{L^2_\Omega} + \|\nabla \Phi\|_{L^2_\Omega} < \infty \right\}. \tag{6}$$

The vector $\mathbf{g} = [g_1, g_2]$ denotes the vector of prescribed Dirichlet-type boundary conditions. Here, the space $L^2_\Omega = L^2(\Omega)$ is the space of square-integrable functions defined on Ω, and is equipped with the standard L^2-norm:

$$\|\Phi\|_{L^2_\Omega} = \sqrt{\int_\Omega \Phi^2 dx}. \tag{7}$$

Employing integration-by-parts, the variational formulation given by Eq. (3) can be recast as follows:

$$\begin{cases} \text{find } \mathbf{u} \in \boldsymbol{\mathcal{S}}_{\mathbf{u}} \text{ such that } \forall \mathbf{w} \in \boldsymbol{\mathcal{V}}_{\mathbf{u}} : \\ \int_\Omega \mathbf{w} \cdot \left(\dfrac{\partial \mathbf{u}}{\partial t} + \mathbf{a} \cdot \nabla \mathbf{u} - \mathbf{s} \right) d\Omega + \int_\Omega \left(\nabla \mathbf{w} : (\mathbf{D} \nabla \mathbf{u}) \right) d\Omega \\ - \int_\Omega \mathbf{w} \cdot \mathbf{f}(\mathbf{u}) \, d\Omega - \int_{\Gamma_N} \mathbf{w} \cdot \mathbf{h} \, d\Gamma = 0, \end{cases} \tag{8}$$

where $\mathbf{h} = \mathbf{D} \nabla \mathbf{u} \cdot \mathbf{n}$ is the Neumann-type boundary data, $\mathbf{n}$ is outward-oriented unit normal vector, and Γ denotes the boundary of the computational domain Ω, i.e., $\Gamma = \partial\Omega$. Note that $\Gamma = \Gamma_N \cup \Gamma_D$ and $\Gamma_N \cap \Gamma_D = \emptyset$, where the subscripts "N" and "D" indicate that whether the boundary is subject to Neumann- or Dirichlet-type boundary conditions.

If the computational domain Ω is divided into finite number of elements Ω^e, $e = 1, 2, \ldots, n_{\text{el}}$, where n_{el} denotes the number of these elements, then the GFEM formulation associated with system (1) reads:

$$\begin{cases} \text{find } \mathbf{u}^h \in \boldsymbol{\mathcal{S}}^h_{\mathbf{u}} \text{ such that } \forall \mathbf{w}^h \in \boldsymbol{\mathcal{V}}^h_{\mathbf{u}} : \\ \int_\Omega \mathbf{w}^h \cdot \left(\dfrac{\partial \mathbf{u}^h}{\partial t} + \mathbf{a} \cdot \nabla \mathbf{u}^h - \mathbf{s} \right) d\Omega + \int_\Omega \left(\nabla \mathbf{w}^h : \left(\mathbf{D} \nabla \mathbf{u}^h \right) \right) d\Omega \\ - \int_\Omega \mathbf{w}^h \cdot \mathbf{f}\left(\mathbf{u}^h \right) d\Omega - \int_{\Gamma_N} \mathbf{w}^h \cdot \mathbf{h}^h \, d\Gamma = 0, \end{cases} \tag{9}$$

where the superscript "h" indicates that the functions that are the components of the associated vectors/matrices come from a finite-dimensional space. The finite-dimensional function spaces are defined as follows:

$$\boldsymbol{\mathcal{S}}^h_{\mathbf{u}} = \boldsymbol{\mathcal{V}}^h_{\mathbf{u}} = \left\{ \boldsymbol{\Phi}^h \in \left[\mathcal{C}\left(\overline{\Omega} \right) \right]^2 : \boldsymbol{\Phi}^h|_{\partial\Omega} = \mathbf{0}, \, \boldsymbol{\Phi}^h|_{\Omega^e} \in \left[\mathcal{P}_1\left(\Omega^e \right) \right]^2, \, \forall \Omega^e \in \mathcal{T}^h \right\}, \tag{10}$$

where $\mathcal{P}_1(\Omega^e)$ is the space of linear polynomials over the triangular element $\Omega^e \in \mathcal{T}^h$, $\mathcal{C}\left(\overline{\Omega} \right)$ is the space of continuous functions defined on the closure of the computational domain, and $\mathcal{T}^h$ is the triangulation of the domain Ω into triangular elements.

3 Stabilized finite element formulations

This section describes the SUPG and SUPG-YZβ formulations of the model problem given by Eq. (1), respectively.

The SUPG formulation associated with system (1) can be given as follows:

$$\begin{cases} \text{find } \mathbf{u}^h \in \boldsymbol{\mathcal{S}}_{\mathbf{u}} \text{ such that } \forall \mathbf{w}^h \in \boldsymbol{\mathcal{V}}_{\mathbf{u}}^h : \\ \displaystyle\int_\Omega \mathbf{w}^h \cdot \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{a} \cdot \nabla \mathbf{u}^h - \mathbf{s} \right) d\Omega + \int_\Omega \left(\nabla \mathbf{w}^h : \left(\mathbf{D} \nabla \mathbf{u}^h \right) \right) d\Omega \\ \displaystyle - \int_\Omega \mathbf{w}^h \cdot \mathbf{f}\left(\mathbf{u}^h \right) d\Omega - \int_{\Gamma_N} \mathbf{w}^h \cdot \mathbf{h}^h \, d\Gamma \\ \displaystyle + \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} \boldsymbol{\tau}_{\text{SUPG}} \left(\frac{\partial \mathbf{w}^h}{\partial x_k} \right) \mathbf{a}^T \cdot \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{a} \cdot \nabla \mathbf{u}^h - \nabla \cdot \left(\mathbf{D} \nabla \mathbf{u}^h \right) - \mathbf{f}\left(\mathbf{u}^h \right) - \mathbf{s} \right) d\Omega = 0, \end{cases} \tag{11}$$

where the finite-dimensional space $\boldsymbol{\mathcal{V}}_{\mathbf{u}}^h$ is defined by Eq. (10). In this formulation, e is the element counter and $\boldsymbol{\tau}_{\text{SUPG}}$ is the diagonal SUPG stabilization matrix. How these stabilization parameters are determined directly affects the accuracy and quality of the numerical approximations.

Remark 1 *In the last line of Eq. (11), by the term* $\left(\frac{\partial \mathbf{w}^h}{\partial x_k} \right)$*, we refer to Einstein summation convention; i.e.,*

$$\left(\frac{\partial \mathbf{w}^h}{\partial x_k} \right) = \sum_{k=1}^{n_{\text{sd}}} \frac{\partial w_k^h}{\partial x_k}, \tag{12}$$

where w_k^h *is the k—th component of the test vector* $\mathbf{w}^h$*.*

For solving stationary problems, the stabilization matrix, $\boldsymbol{\tau}_{\text{SUPG}}$, is composed of stabilization parameters, τ_{SUPG}^i's, which are defined as follows [44]:

$$\tau_{\text{SUPG}}^i = \left[\left(\frac{2\|\mathbf{a}\|}{h^e} \right)^2 + \left(\frac{4\epsilon_i}{(h^e)^2} \right)^2 \right]^{-\frac{1}{2}}, \tag{13}$$

where $i = 1, 2$, the norm $\|\cdot\|$ represents the standard Euclidean norm, and h^e is the cell diameter associated with element Ω^e. For unsteady problems, these parameters can be defined as

$$\tau_{\text{SUPG}}^i = \left[\left(\frac{2}{\Delta t} \right)^2 + \left(\frac{2\|\mathbf{a}\|}{h^e} \right)^2 + \left(\frac{4\epsilon_i}{(h^e)^2} \right)^2 \right]^{-\frac{1}{2}}. \tag{14}$$

For systems involving different convection vectors, e.g., $\mathbf{a}_i$'s, these parameters can be defined in the following fashion:

$$\tau_{\text{SUPG}}^i = \left[\left(\frac{2}{\Delta t} \right)^2 + \left(\frac{2\|\mathbf{a}_i\|}{h^e} \right)^2 + \left(\frac{4\epsilon_i}{(h^e)^2} \right)^2 \right]^{-\frac{1}{2}}. \tag{15}$$

In these definitions given by Eqs. (13)–(14), the superscript "i" in τ_{SUPG}^i indicates that the parameter is associated with the ith equation in the system. Similarly, the subscripts in convection vectors (i.e., $\mathbf{a}_i$'s) and diffusion parameters (i.e., ϵ_i's) indicate that they belong to the ith equation.
For further details and a review of various definitions of the stabilization parameters and element

length scales, we refer the interested reader to [38, 39, 45, 46] and references therein.

Remark 2 *One should note that, compared to the classical GFEM formulation given by Eq. (3), the SUPG formulation introduced by Eq. (11) involves additional element-based stabilization terms controlled by the stabilization parameters. By adding these terms, the original system gains artificial dissipation in the streamline direction.*

Remark 3 *One can also find a variation in the definition of stabilization parameters given by Eqs. (13)–(14) based on the approach followed in [44, 47]:*

$$\tau^i_{\text{SUPG}} = \left[\left(\frac{2}{\Delta t} \right)^2 + \left(\frac{2\|\mathbf{a}_i\|}{h^e} \right)^2 + 9 \left(\frac{4\epsilon_i}{(h^e)^2} \right)^2 \right]^{-\frac{1}{2}}. \tag{16}$$

For stationary problems, the term associated with time is simply omitted, as done in Eq. (13).

We adopt the stabilization parameter described by Eq. (16), in our computations. Then, the stabilization matrix, $\boldsymbol{\tau}_{\text{SUPG}}$, associated with the model problem described by Eq. (1) can be given as follows:

$$\boldsymbol{\tau}_{\text{SUPG}} = \begin{bmatrix} \tau^1_{\text{SUPG}} & 0 \\ 0 & \tau^2_{\text{SUPG}} \end{bmatrix}. \tag{17}$$

Eventually, the SUPG-YZβ formulation associated with system (1) can be described as follows:

$$\begin{cases} \text{find } \mathbf{u}^h \in \mathcal{S}^h_{\mathbf{u}} \text{ such that } \forall \mathbf{w}^h \in \mathcal{V}^h_{\mathbf{u}} : \\ \displaystyle\int_\Omega \mathbf{w}^h \cdot \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{a} \cdot \nabla \mathbf{u}^h - \mathbf{s} \right) d\Omega + \int_\Omega \left(\nabla \mathbf{w}^h : \left(\mathbf{D} \nabla \mathbf{u}^h \right) \right) d\Omega \\ \displaystyle - \int_\Omega \mathbf{w}^h \cdot \mathbf{f}\left(\mathbf{u}^h \right) d\Omega - \int_{\Gamma_{\text{N}}} \mathbf{w}^h \cdot \mathbf{h}^h \, d\Gamma \\ \displaystyle + \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} \boldsymbol{\tau}_{\text{SUPG}} \left(\frac{\partial \mathbf{w}^h}{\partial x_k} \right) \mathbf{a}^T \cdot \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{a} \cdot \nabla \mathbf{u}^h - \nabla \cdot \left(\mathbf{D} \nabla \mathbf{u}^h \right) - \mathbf{f}\left(\mathbf{u}^h \right) - \mathbf{s} \right) d\Omega \\ \displaystyle + \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} \nu_{\text{SHOC}} \left(\frac{\partial \mathbf{w}^h}{\partial x_k} \right) \cdot \left(\frac{\partial \mathbf{u}^h}{\partial x_k} \right) d\Omega = 0, \end{cases} \tag{18}$$

where the term ν_{SHOC} is the stabilization parameter associated with the YZβ shock-capturing technology. The shock-capturing parameter is defined in light of studies by Tezduyar [30–32]. In this work, we slightly modify the original definition of the shock-capturing parameter to solve the model problem as follows [33]:

$$\nu^i_{\text{SHOC}} = |\text{Y}^{-1}\text{Z}_i| \left(\sum_{i=1}^{n_{\text{sd}}} \left| \text{Y}^{-1} \frac{\partial u^h_i}{\partial x_i} \right|^2 \right)^{\frac{\beta}{2}-1} \left(\frac{h^i_{\text{SHOC}}}{2} \right)^{\beta}, \tag{19}$$

where

$$Z_i = \frac{\partial u_i}{\partial t} - \nabla \cdot \left(\epsilon_i \nabla u_i^h\right) + \mathbf{a} \cdot \nabla u_i^h - f_i - s_i. \tag{20}$$

Remark 4 *Compared to the SUPG-based stabilized formulation given by Eq. (11), the SUPG-YZβ formulation described by Eq. (18) involves additional element-based stabilization terms associated with the shock-capturing mechanism. These new terms introduce additional artificial diffusion in the direction of solution gradients, which helps to mitigate undershoots and overshoots around sharp layers.*

Remark 5 *As also mentioned by Remark 1, by the terms* $\left(\frac{\partial \mathbf{w}^h}{\partial x_k}\right)$ *and* $\left(\frac{\partial \mathbf{u}^h}{\partial x_k}\right)$ *in Eq. (18), we refer to Einstein summation convention.*

Remark 6 *By using Eq. (20) in computations, we adopt the residual form of* Z_i*, which is similar to that used by Bazilevs et al. in [33] as a variation of the advective form introduced in [26, 27]:*

$$Z_i = \mathbf{a} \cdot \nabla u_i^h. \tag{21}$$

In addition to that used in [33], following this way, we also include the reaction and source terms in the definition of Z_i*.*

Remark 7 *The definition of quantity* Z_i *can be extended to handle the case of different advection vectors* a_i*'s in the same way followed for describing the SUPG stabilization parameter defined by Eq. (15).*

In Eq. (19), the quantity Y can be determined as follows:

$$\mathrm{Y} = \sqrt{u_{1,\,\mathrm{ref}}^2 + u_{2,\,\mathrm{ref}}^2}. \tag{22}$$

The reference values $u_{\mathrm{i,\,ref}}$'s are typically determined according to the initial data given for time-dependent problems. For steady-state problems, they can also be determined as reference values or through numerical experiments. The local element length scales, h_{SHOC}^i's, are defined as

$$h_{\mathrm{SHOC}}^i = 2\left(\sum_{a=1}^{n_{\mathrm{en}}} |\mathbf{j}_i \cdot \nabla N_a|\right)^{-1}, \tag{23}$$

with the unit vector in the direction of the gradient of u_i^h:

$$\mathbf{j}_i = \frac{\nabla u_i^h}{\|\nabla u_i^h\|}. \tag{24}$$

Here, the term N_a represents the interpolating function associated with element node a. The indices n_{sd} and n_{en} stand for the number of space-dimensions and number of element nodes. The sharpness parameter β is typically set as $\beta = 1$ for mild shocks and $\beta = 2$ for sharper shocks [30–32]. Since the main focus of this study is problems highly dominated by convection, we set the parameter β as $\beta = 2$ in our computations.

Remark 8 *The finite element formulations introduced in the previous lines for a two-species model given by Eq.* (1) *can be easily extended to models with more species and/or higher dimensions. Similarly, as discussed in the previous lines, these formulations can also be modified for handling systems of RCD-type equations with different convection fields instead of ones having the same convection vector .*

4 Numerical experiments

After providing some computational details, such as the time-integration, absolute and relative error tolerances associated with linear and nonlinear solvers, and computing platform, as the first numerical experiment, we consider a cross-diffusion reaction-diffusion system with component-wise analytical solutions in order to validate our GFEM solvers. Then, again for verification purposes, we deal with a steady convection-dominated RCD system with component-wise exact solutions. Following that, we focus on coupled systems of time-dependent RCD equations.

Further computational details

Throughout this work, for unsteady problems, the time discretization is performed with the backward Euler scheme, i.e., the semi-discrete (spatially discretized) formulations (see Eqs. (9), (11), and (18)) introduced in the previous sections are discretized such that as advancing from time-step n to $n+1$:

$$\frac{\partial U^h}{\partial t} = \frac{U^h_{n+1} - U^h_n}{\Delta t} = \mathcal{J}^h_{n+1}, \tag{25}$$

where $\mathcal{J}^h_{n+1}$ represents the rest of the terms in the variational formulations computed at time steps $n+1$. The relative and absolute convergence criteria associated with the N–R algorithm are both set to 1.0×10^{-12}. All computations are carried out in serial in the FEniCS [48–50] scientific computing environment, which is particularly dedicated to the finite element solution of differential equations and allows the user high-level C++ and Python interfaces, on a computer equipped with Intel i7-12650H CPU and 40GB RAM running Ubuntu 20.04.5 LTS. For further details on the FEniCS Project, we refer the interested reader to the references provided above and the official webpage of the project: `https://fenicsproject.org/`. Since the test computations, apart from the first numerical experiment (*Application 1*), are of highly nonlinear nature, we set the quadrature degree associated with the numerical integration to eleven. Besides, all the finite element meshes are triangular and generated by using the built-in `mshr` component of FEniCS.

Test computations

Application 1 – Reaction-diffusion with cross-diffusion. We take this test example, which was originally studied in [51], from [9], for comparison purposes. It is described as follows:

$$\begin{cases} \dfrac{\partial u}{\partial t} = D_{11}\,\Delta u + D_{12}\,\Delta v + D_{13}\,v, \\ \dfrac{\partial v}{\partial t} = D_{21}\,\Delta u + D_{22}\,\Delta v + D_{23}\,u, \end{cases} \tag{26}$$

where the spatial domain is defined by $\Omega = (0, 2\pi)^2$, and $t > 0$. The component-wise initial conditions are defined as

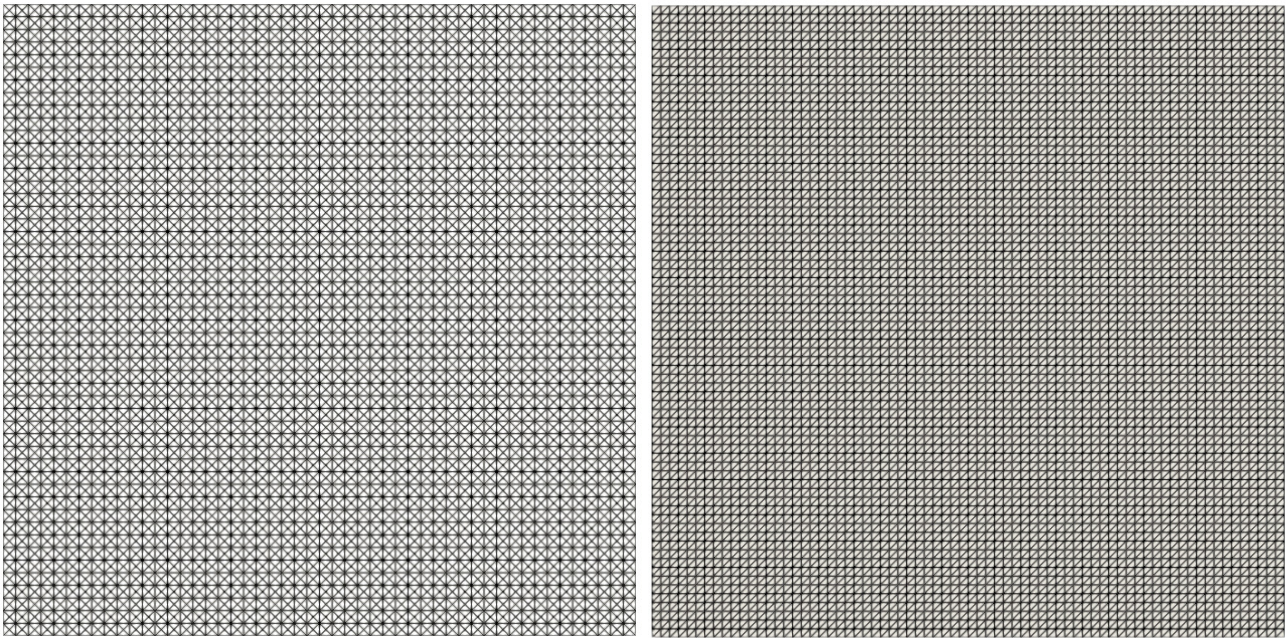
$$u\,(x_1, x_2, 0) = \cos\,(2x_1) + \cos\,(2x_2)\,, \quad v\,(x_1, x_2, 0) = \cos\,(x_1) + \cos\,(x_2)\,. \tag{27}$$

At walls, zero-flux, i.e., homogeneous Neumann-type boundary conditions apply. Then, the component-wise analytical solutions to the system described by Eq. (26) are given as

$$u\left(x_1, x_2, t\right) = \exp\left(-4t\beta\right)\left[\cos\left(2x_1\right) + \cos\left(2x_2\right)\right], \tag{28}$$

$$v\left(x_1, x_2, t\right) = \exp\left(-t\beta\right)\left[\cos\left(x_1\right) + \cos\left(x_2\right)\right], \tag{29}$$

where $D_{11} = D_{22} = \beta = 0.01$, $D_{12} = 1.5\beta$, $D_{21} = 0.5\beta$, $D_{13} = D_{12}$, and $D_{23} = 4D_{21}$. We follow the same fashion used in [9] by setting the final time as $t_f = 50$ and time-step size as $\Delta t = 0.005$, which results in $10,000$ iterations. In computations, the mesh constructed with crossed elements, which is shown in Figure 1a, is used. We directly employ the GFEM to solve this problem since it



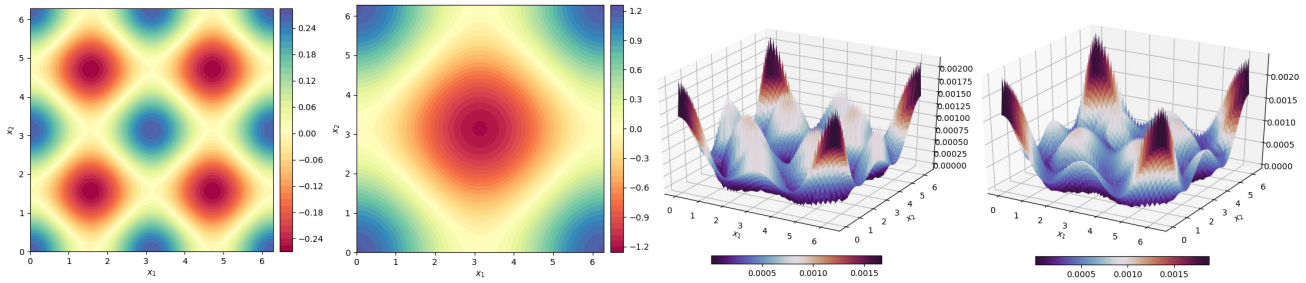
(a) **(b)**

Figure 1. Meshes: (a) constructed with $n_{\text{el}} = 10,000$ crossed elements and $n_{\text{en}} = 5,101$ nodes for solving *Application 1*, (b) constructed with $n_{\text{el}} = 10,368$ elements and $n_{\text{en}} = 5,329$ nodes for solving *Application 2*.

does not have convective terms. Figure 2 shows the component-wise GFEM solutions of *Application 1*, which is described by system (26), along with the corresponding absolute errors. It is observed that the absolute errors take their maximum values around corners of the computational domain. In Figure 3, the absolute errors in the GFEM approximation along the line $x_2 = \pi$ are displayed. It is revealed that the maximum absolute error is around 0.001. In comparison to the results presented in [9], where the author employed B-spline basis functions of order $k = 8$, our results show good agreement with them (see Figure 3).

Application 2 – Convection-diffusion with nonlinear reaction. This second example is a stationary problem and is from [8] by Yücel et al.:

$$\begin{cases} -\epsilon_u \Delta u + \beta_u \cdot \nabla u + \alpha_u u + \gamma_u uv - f_u = 0, \\ -\epsilon_v \Delta v + \beta_v \cdot \nabla v + \alpha_v v + \gamma_v uv - f_v = 0, \end{cases} \tag{30}$$



(a) (b) (c) (d)

Figure 2. GFEM approximations for solving *Application 1*; $t_f = 50$ and $\Delta t = 0.005$: (a) surface plot for $u(x_1, x_2)$, (b) surface plot for $v(x_1, x_2)$, (c) elevation plot for absolute error in $u(x_1, x_2)$, and (d) elevation plot for absolute error in $v(x_1, x_2)$.

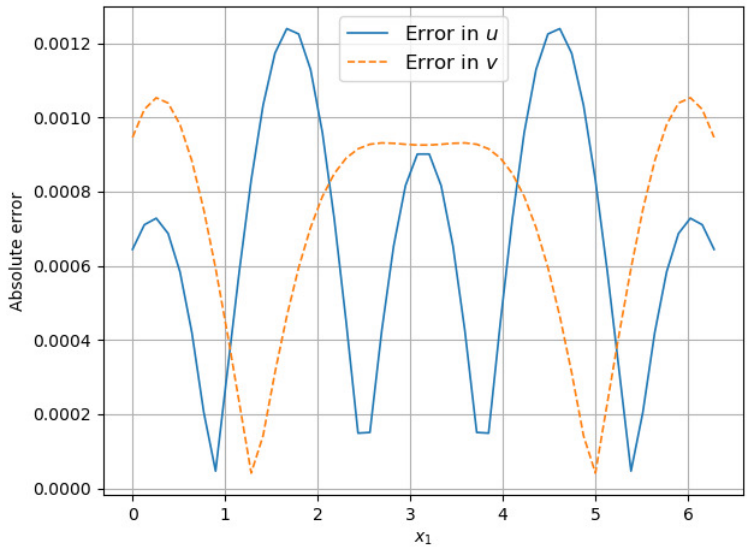


Figure 3. Comparison of absolute errors for solving *Application 1* along line $x_2 = \pi$.

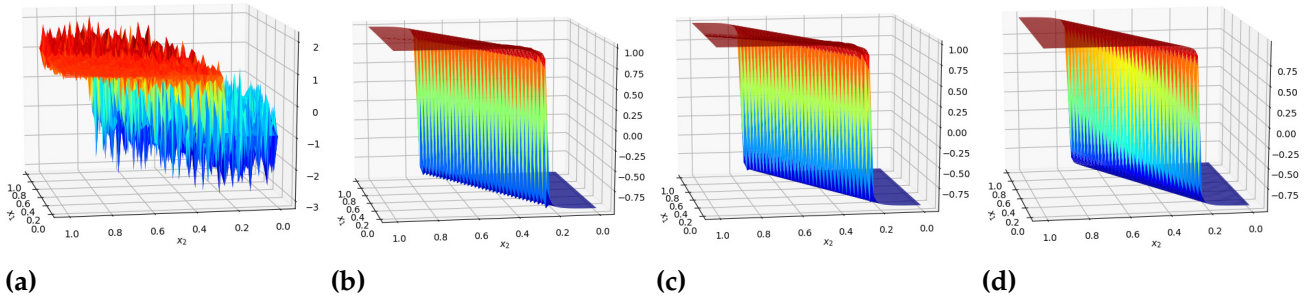
where the unknown functions, i.e., $u(\mathbf{x})$ and $v(\mathbf{x})$, represents the reactant concentrations, the computational domain is taken as $\Omega = (0, 1)$, and the given functions $f_u(\mathbf{x})$ and $f_v(\mathbf{x})$ are the source functions. The parameters are set as follows: the diffusion coefficients are $\epsilon_u = \epsilon_v = 10^{-5}$, the convection vectors are $\beta_u = [2, 3]$ and $\beta_v = [1, 0]$, the reaction coefficients $\alpha_u = \alpha_v = 1.0$, and $\gamma_u = \gamma_v = 0.1$. The source functions f_u and f_v are determined such that the following analytical solutions hold [8]:

$$u(x_1, x_2) = \frac{2}{\pi} \arctan\left(\frac{1}{\sqrt{\epsilon_u}}\left[-\frac{1}{2}x_1 + x_2 - 0.25\right]\right), \tag{31}$$

$$v(x_1, x_2) = 4\exp\left(\frac{-1}{\sqrt{\epsilon_v}}\left((x_1 - 0.5)^2 + 3(x_2 - 0.5)^2\right)\right)\sin(\pi x_1)\cos(\pi x_2). \tag{32}$$

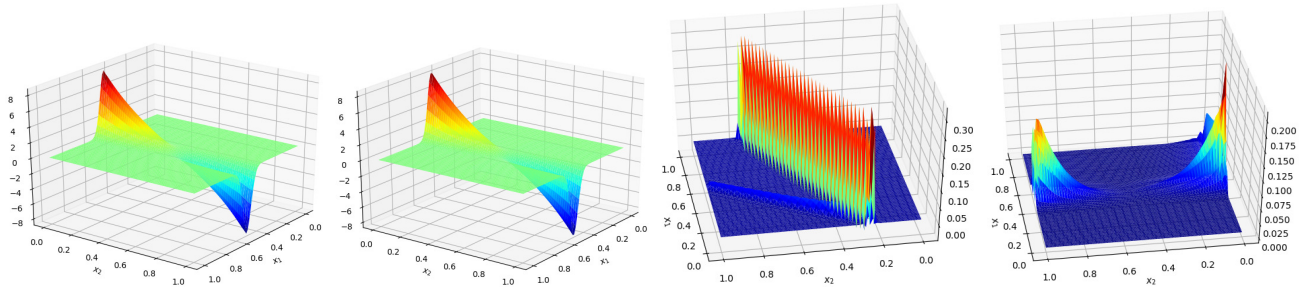
Figure 1b shows the mesh constructed with 10,368 triangular elements used in computations for solving *Application 2*. Figure 4 presents a comparison of the performances of the proposed formulations for $u(x_1, x_2)$. It is clearly seen that the GFEM approximation is completely polluted with nonphysical oscillations. Although the SUPG formulation manages to eliminate spurious oscillations significantly, it requires additional treatment to resolve steep gradients. As to the SUPG-YZβ formulation, there is not any significant oscillatory behavior, and the resulting approximation

is in good agreement with the exact solution. Since the $v\,(x_1, x_2)$ component of the solution vector **u** does not exhibit oscillatory behavior, only the exact solution and SUPG-YZβ approximations are given in Figure 5, along with absolute errors in SUPG-YZβ approximations for solving $u\,(x_1, x_2)$ and $v\,(x_1, x_2)$ components. We observe that the absolute errors almost completely vanish far from the regions where solutions exhibit rapid changes. In other words, the SUPG-YZβ formulation looks for a compromise between stability and accuracy. While high-quality solution profiles are obtained similar to those reported in [8], the SUPG-YZβ formulation achieves this on a coarser mesh without the need for any mesh refinement techniques.



(a) **(b)** **(c)** **(d)**

Figure 4. Comparison of approximations to $u\,(x_1, x_2)$ obtained with various formulations for solving *Application 2*: (a) GFEM, (b) SUPG, (c) SUPG-YZβ, and (d) exact solution.



(a) **(b)** **(c)** **(d)**

Figure 5. Comparison of SUPG-YZβ approximations with exact solutions for solving *Application 2*: (a) SUPG-YZβ solution for $v\,(x_1, x_2)$, (b) exact solution to $v\,(x_1, x_2)$, (c) absolute error in SUPG-YZβ approximation for $u\,(x_1, x_2)$, and (d) absolute error in SUPG-YZβ approximation for $v\,(x_1, x_2)$.

Application 3 – Schnakenberg reaction model. Here, we deal with the Schnakenberg reaction model, which was originally introduced by Schnakenberg in [52]. The model can be described as follows [5]:

$$\begin{cases} \dfrac{\partial u}{\partial t} + \mathbf{a} \cdot \nabla u - \Delta u - \gamma \left(\alpha - u + u^2 v\right) = 0, \\ \dfrac{\partial v}{\partial t} + \mathbf{a} \cdot \nabla v - d\,\Delta v - \gamma \left(\beta - u^2 v\right) = 0, \end{cases} \tag{33}$$

where the diffusion constant d refers to the relationship between the species diffusivities, the constants α and β denote the production and consumption for species u and v, respectively, and the nonlinear term $u^2 v$ is the catalysis term, which represents activation for u and consumption for

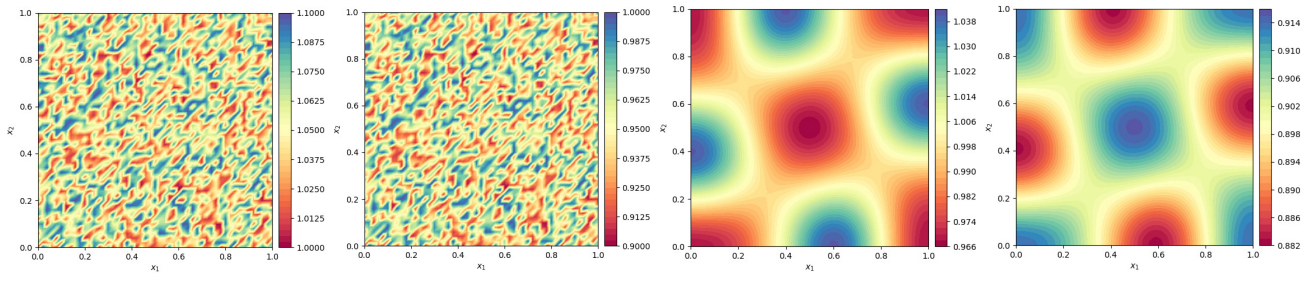
v. We consider this problem for two different sets of parameters and initial/boundary conditions. *Case I* aims to compare our results with those reported previously. Then, in *Case II*, the parameters and initial/boundary conditions are determined in such a way that the solution to the Schnakenberg model involves sharp gradients. For both computations, we suppose that the problems are defined on the domain $\Omega = (0,1)$, the time-step size is taken as $\Delta t = 0.005$, and the final time is $t_f = 2.5$. Note that the parameter β given in the model is not related to YZβ shock-capturing.

Case I: The set of parameters are [5]: $\alpha = 0.1$, $\beta = 0.9$, $\gamma = 230.82$, and the velocity field is $\mathbf{a} = [-\omega\,(x_2 - 0.5)\,, \omega\,(x_1 - 0.5)]$, where $\omega = 0.6$. In computations, a mesh having the same structure as that given in Figure 1b but constructed with 5,408 triangular elements and 2,809 nodes is used. Figure 6a–Figure 6b show the initial conditions for u and v, respectively. For determining these conditions, we perturb each reactive component around the steady-states by around 10% [53]. That is, the initial conditions are defined as follows:

$$u\,(x_1, x_2, t = 0) = u_s + \varepsilon u_s, \tag{34}$$
$$v\,(x_1, x_2, t = 0) = v_s + \varepsilon v_s, \tag{35}$$

where, $(u_s, v_s) = \left(\alpha + \beta, \frac{\beta}{(\alpha+\beta)^2}\right)$. Note that the reaction terms vanish for $(u, v) = (u_s, v_s)$. Homogeneous Neumann-type boundary conditions apply on walls.

In Figure 6c–6d, we present SUPG-YZβ solutions to *Application 3* for *Case I*. Compared to the results reported in [5], the present solution profiles are in good agreement when a mesh constructed with a similar number of elements is used. This fact indicates that the proposed formulation does not distort the solutions by introducing unnecessary artificial diffusivity. The rotation of the Turing patterns is due to the velocity field $\mathbf{a}$, and these rotations are in the same direction as $\mathbf{a}$. On the other hand, numerical experiments reveal that when finer meshes are employed, all the proposed formulations yield slightly different approximations than those reported in [5]. The author believes that the slight difference in [5] is due to the coarser mesh used because of the limited computational resources available on those days when the numerical experiments were carried out.



(a) **(b)** **(c)** **(d)**

Figure 6. *Application 3 – Case I*: (a) initial condition for $u\,(x_1, x_2, t)$, (b) initial condition for $v\,(x_1, x_2, t)$, (c) SUPG-YZβ approximation to $u\,(x_1, x_2)$, and (d) SUPG-YZβ approximation to $v\,(x_1, x_2)$.

Case II: In this case, we modify the original system given by Eq. (33) as follows:

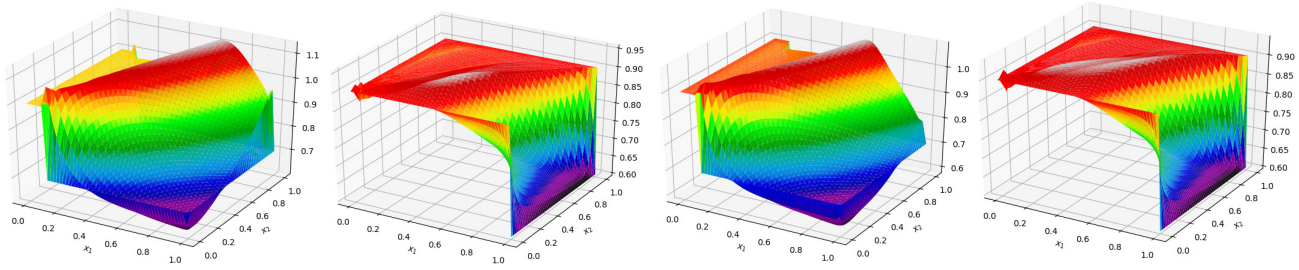
$$\begin{cases} \dfrac{\partial u}{\partial t} + \mathbf{a}_u \cdot \nabla u - d_u\,\Delta u - \gamma\left(\alpha - u + u^2 v\right) = 0, \\ \dfrac{\partial v}{\partial t} + \mathbf{a}_v \cdot \nabla v - d_v\,\Delta v - \gamma\left(\beta - u^2 v\right) = 0. \end{cases} \tag{36}$$

We set the parameters as: $\alpha = 0.1$, $\beta = 0.9$, $\gamma = 0.52$, the convection vectors are $\mathbf{a}_u = [x_1, 2x_2]$ and $\mathbf{a}_v = [x_1, 2x_1]$, and the diffusion parameters are $d_u = d_v = 10^{-8}$. We use the same mesh used for *Case I*. The initial conditions are set as $u(x_1, x_2, t = 0) = 1.0$ and $v(x_1, x_2, t = 0) = 0.9$. The Dirichlet-type boundary conditions are prescribed as follows:

$$g_u(x_1, x_2, t) = \begin{cases} 0.7, & \text{if } (x_1 > 0.1 \text{ and } x_2 > 0.4) \text{ or } x_2 < 0.6, \\ 1.0, & \text{otherwise,} \end{cases} \tag{37}$$

$$g_v(x_1, x_2, t) = \begin{cases} 0.6, & \text{if } \quad x_1 = 1, \\ 0.9, & \text{otherwise.} \end{cases} \tag{38}$$

In Figure 7, we compare the SUPG and SUPG-YZβ approximations for solving *Case II* of *Application 3*. The N–R algorithm fails to converge for the GFEM formulation. It can be observed that the sharp gradients in solutions, particularly those obtained for $u(x_1, x_2, t)$, are resolved accurately without any significant localized oscillations by employing the SUPG-YZβ formulation.



(a) **(b)** **(c)** **(d)**

Figure 7. *Application 3 – Case II*: (a) SUPG approximation to $u(x_1, x_2, t)$, (b) SUPG approximation to $v(x_1, x_2, t)$, (c) SUPG-YZβ approximation to $u(x_1, x_2)$, and (d) SUPG-YZβ approximation to $v(x_1, x_2)$.

Application 4 – A mussel-algae interaction model. This 1D model, which was originally introduced in [54] by Koppel et al., is taken from [10] in its nondimensionalized form:

$$\begin{cases} \dfrac{\partial u}{\partial t} = D_{11} \dfrac{\partial^2 u}{\partial x_1^2} - q \dfrac{\partial u}{\partial x_1} + \alpha(1 - u) - uv, \\ \dfrac{\partial v}{\partial t} = D_{21} \dfrac{\partial^2 v}{\partial x_1^2} + \sigma uv - \gamma v^2 - \dfrac{v}{v + 1}, \end{cases} \tag{39}$$

where $0 < x_1 < L = 10$ and $t > 0$. The unknown functions $u(x_1, t)$ and $v(x_1, t)$ represent the algae and mussel density, respectively, α is the exchange rate of mussels, and γ denotes the competition between the mussels (intraspecific competition). One of the primary food sources that mussels consume is algae. Following [10], it is assumed that algae constantly convects at the rate of q, at which algae is supplied to the mussels bed by unidirectional water flow, from the open sea toward the shore. We consider system (39) for three different sets of parameter and boundary conditions. In the first two cases, we verify the proposed formulation and solvers by comparing the results obtained with those reported by Wang et al. [10]. As to the third scenario, we modify the originally introduced parameter set and conditions such that system (39) becomes convection-dominated. For all cases, the number of elements is $n_{\text{el}} = 256$ and time step-size is set

to $\Delta t = 0.1$.

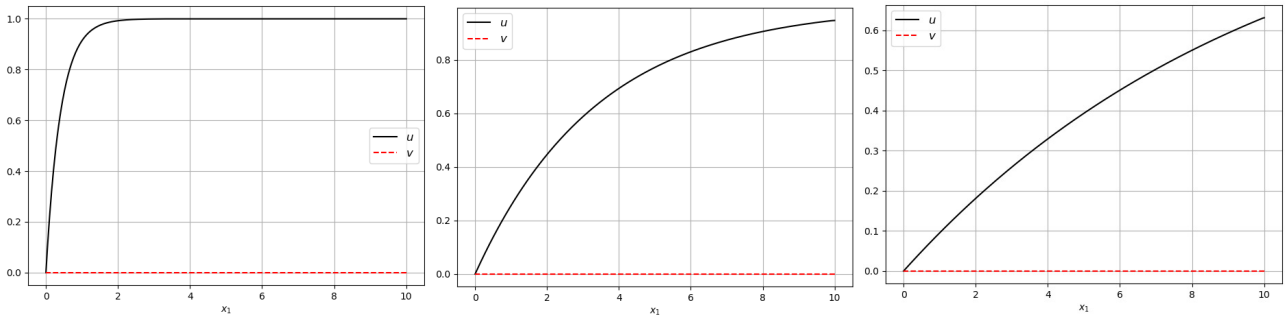
Case I: In the first case, system (39) is subject to homogeneous Neumann boundary condition at $x_1 = L$ and Danckwerts-type inflow boundary condition applies at $x_1 = 0$ (for further details, see [10, 55]):

$$D_{11} \frac{\partial u\,(0,t)}{\partial x_1} = q u\,(0,t)\,, \tag{40}$$

$$\frac{\partial u\,(L,t)}{\partial x_1} = 0. \tag{41}$$

The parameter set is taken from [10]: $D_{11} = 0.1$, $D_{21} = 0.3$, $\alpha = 0.6$, $\sigma = 0.5$, and $\gamma = 0.2$. In numerical experiments, we study various values of the advection rate q, i.e., $q = 10^{-4}$, $q = 2$, and $q = 6$. The initial conditions are $u\,(x_1, 0) = 0.1$ and $v\,(x_1, 0) = 1.0$. The simulations are run for the terminal time $t_{\mathrm{f}} = 100$.

Figure 8 shows that, for a range of convection rate constant q, the mussels die out and only the algae remain. It implies that mussels cannot exist when the rate at which ingested algae are converted to mussels and their production is less than the rate at which they are consumed. Nonetheless, the algae's biomass is affected by the water flow, i.e., the biomass of the algae decreases as the advection rate q increases.



(a) **(b)** **(c)**

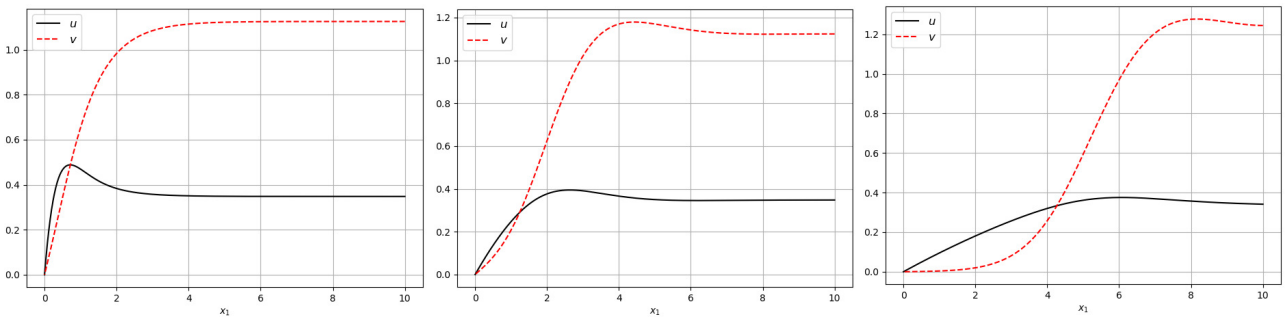
Figure 8. *Comparison of SUPG-YZβ approximations for solving Application 4 – Case I: (a) $q = 10^{-4}$, (b) $q = 2$, and (c) $q = 6$.*

Case II: In this case, we consider the same boundary conditions used in the first case. The initial conditions are $u\,(x_1, 0) = 0.8$ and $v\,(x_1, 0) = 0.6$. As to the parameters, we only change the conversion constant σ to $\sigma = 2.0$. The terminal time is set to $t_{\mathrm{f}} = 100$.

Figure 9 presents the SUPG-YZβ approximations for various values of the convection constant q. These figures indicate that algae are carried downstream by the water as the advection rate increases. Because of this, mussels have more food available downstream, which causes them to accumulate downstream as well. In both cases (*Case I* and *Case II*), we observe that the results are in pretty good agreement with those reported in [10]. We, in the last case, examine the system given by Eq. (39) for convection dominance.

Case III: For this case, the initial conditions are $u\,(x_1, 0) = 0.1$ and $v\,(x_1, 0) = 1.0$. The parameter set is: $D_{11} = 10^{-7}$, $D_{21} = 3 \times 10^{-7}$, $\alpha = 1.6$, $\gamma = 1.2$, $\sigma = 2.0$, and $q = 10$. The Dirichlet-type boundary conditions are prescribed as follows:

$$g_u\,(x_1 = 0, t) = 0, \quad g_u\,(x_1 = L, t) = 0, \tag{42}$$



(a) **(b)** **(c)**

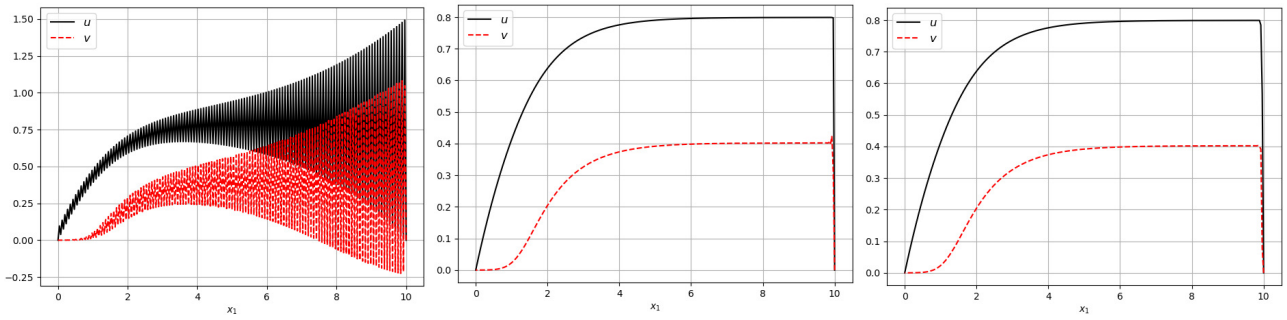
Figure 9. *Comparison of SUPG-YZβ approximations for solving Application 4 – Case II: (a) $q = 10^{-4}$, (b) $q = 2$, and (c) $q = 6$.*

and

$$g_v\left(x_1 = 0, t\right) = 0, \quad g_v\left(x_1 = L, t\right) = 0. \tag{43}$$

The terminal time is set as $t_{\mathrm{f}} = 10$.

In Figure 10a, it is observed that the GFEM yields approximations completely polluted with node-to-node spurious oscillations. During the numerical simulations, it was revealed that when the number of elements was increased, the situation got even worse. On the other hand, it is seen in Figure 10b that the SUPG formulation performs quite well, eliminating almost all nonphysical oscillations but a very tight region near $x_1 = 10$. Finally, we observe the effect of the shock-capturing mechanism in Figure 10c; it helps capture the steep gradient that occurs near $x_1 = 10$ successfully without introducing excessive dissipation.



(a) **(b)** **(c)**

Figure 10. *Comparison of approximations for solving Application 4 – Case III obtained with: (a) GFEM, (b) SUPG, and (c) SUPG-YZβ.*

5 Concluding remarks

We have proposed a streamline-upwind/Petrov–Galerkin finite element formulation supplemented with YZβ shock-capturing, the so-called SUPG-YZβ formulation, for solving coupled systems of reaction-convection-diffusion equations. For comparison purposes, we first tested the accuracy of the proposed formulation and verified the solver codes for numerical experiments available in the literature. In order to assess the genuine performance of the proposed formulation and demonstrate that the standard Galerkin finite element formulation fails in convection dominance, we have modified the original problems by making them convection-dominated.

We have observed that the SUPG-YZβ formulation successfully eliminates spurious oscillations. The method accomplishes this by making use of only linear interpolation functions and meshes that are relatively coarser than those used in the majority of reported studies, without the need for any fitted or adaptive mesh strategies. In addition to these, it is also noted that the proposed shock-capturing mechanism does not cause the solutions to become distorted by introducing excessive numerical dissipation. Besides that, although any adaptive mesh strategies are not adopted, coarser meshes are used compared to the reported studies, and only linear interpolation functions are employed, the approximations obtained do not exhibit any significant numerical instabilities for more challenging cases.
Our future research is planned to focus on tumor growth phenomena, which can be represented by a coupled system of partial differential equations of the reaction-convection-diffusion type.

Declarations

List of abbreviations

Not applicable.

Ethical approval

The author states that this research complies with ethical standards. This research does not involve either human participants or animals.

Consent for publication

Not applicable.

Conflicts of interest

The author confirms that there is no competing interest in this study.

Data availability statement

Data availability is not applicable to this article as no new data were created or analyzed in this study.

Funding

Not applicable.

Author's contributions

The author has made substantial contributions to the conception, design of the work, the acquisition, analysis, interpretation of data, and the creation of new software used in the work.

References

[1] Painter, K.J. Mathematical models for chemotaxis and their applications in self-organisation phenomena. *Journal of Theoretical Biology*, 481, 162-182, (2019). [CrossRef]

[2] Bellomo, N., Outada, N., Soler, J., Tao, Y. and Winkler, M. Chemotaxis and cross-diffusion models in complex environments: Models and analytic problems toward a multiscale vision. *Mathematical Models and Methods in Applied Sciences*, 32(04), 713-792, (2022). [CrossRef]

[3] Trelles, J.P. Pattern formation and self-organization in plasmas interacting with surfaces. *Journal of Physics D: Applied Physics*, 49(39), 393002, (2016). [CrossRef]

[4] Van Gorder, R.A. Influence of temperature on Turing pattern formation. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 476(2240), 20200356, (2020). [CrossRef]

[5] Garzón-Alvarado, D.A., Galeano, C.H. and Mantilla, J.M. Turing pattern formation for reaction-convection-diffusion systems in fixed domains submitted to toroidal velocity fields. *Applied Mathematical Modelling*, 35(10), 4913-4925, (2011). [CrossRef]

[6] Sarra, S.A. A local radial basis function method for advection-diffusion-reaction equations on complexly shaped domains. *Applied Mathematics and Computation*, 218(19), 9853-9865, (2012). [CrossRef]

[7] Chapwanya, M., Lubuma, J.M.S. and Mickens, R.E. Positivity-preserving nonstandard finite difference schemes for cross-diffusion equations in biosciences. *Computers & Mathematics with Applications*, 68(9), 1071-1082, (2014). [CrossRef]

[8] Yücel, H., Stoll, M. and Benner, P. A discontinuous Galerkin method for optimal control problems governed by a system of convection–diffusion PDEs with nonlinear reaction terms. *Computers & Mathematics with Applications*, 70(10), 2414-2431, (2015). [CrossRef]

[9] Hidayat, M.I.P. Meshless finite difference method with B-splines for numerical solution of coupled advection-diffusion-reaction problems. *International Journal of Thermal Sciences*, 165, 106933, (2021). [CrossRef]

[10] Wang, J., Tong, X. and Song, Y. Dynamics and pattern formation in a reaction-diffusion-advection mussel-algae model. *Zeitschrift fur angewandte Mathematik und Physik*, 73, 117, (2022). [CrossRef]

[11] Clavero, C., Shiromani, R. and Shanthi, V. A numerical approach for a two-parameter singularly perturbed weakly-coupled system of 2-D elliptic convection-reaction-diffusion PDEs. *Journal of Computational and Applied Mathematics*, 436, 115422, (2024). [CrossRef]

[12] Cengizci, S., Uğur, Ö. and Natesan, S. SUPG-YZβ computation of chemically reactive convection-dominated nonlinear models. *International Journal of Computer Mathematics*, 100(2), 283-303, (2023). [CrossRef]

[13] Uzunca, M., Karasözen, B. and Manguoğlu, M. Adaptive discontinuous Galerkin methods for non-linear diffusion-convection-reaction equations. *Computers & Chemical Engineering*, 68, 24-37, (2014). [CrossRef]

[14] Yücel, H., Stoll, M. and Benner, P. Discontinuous Galerkin finite element methods with shock-capturing for nonlinear convection dominated models. *Computers & Chemical Engineering*, 58, 278-287, (2013). [CrossRef]

[15] Joshi, H. and Jha, B.K. Chaos of calcium diffusion in Parkinson's infectious disease model and treatment mechanism via Hilfer fractional derivative. *Mathematical Modelling and Numerical Simulation With Applications*, 1(2), 84-94, (2021). [CrossRef]

[16] Joshi, H. and Jha, B.K. Modeling the spatiotemporal intracellular calcium dynamics in nerve cell with strong memory effects. *International Journal of Nonlinear Sciences and Numerical Simulation*, 24(6), 2383-2403, (2021). [CrossRef]

[17] Joshi, H., Yavuz, M. and Stamova, I. Analysis of the disturbance effect in intracellular calcium dynamic on fibroblast cells with an exponential kernel law. *Bulletin of Biomathematics*, 1(1), 24-39, (2023). [CrossRef]

[18] Brooks, A.N. and Hughes, T.J. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes

equations. *Computer Methods in Applied Mechanics and Engineering*, 32(1-3), 199-259, (1982). [CrossRef]

[19] Hughes, T.J.R. and Brooks A.N. A multi-dimensional upwind scheme with no crosswind diffusion. In: *Finite Element Methods for Convection Dominated Flows*, (pp. 19-35). AMD-Vol. 34, New York, USA: ASME, (1979).

[20] Tezduyar, T.E. and Hughes, T.J.R. Development of time-accurate finite element techniques for first-order hyperbolic systems with particular emphasis on the compressible Euler equations. *NASA Technical Report NASA-CR-204772*, NASA, 510, (1982).

[21] Tezduyar, T. and Hughes, T. Finite element formulations for convection dominated flows with particular emphasis on the compressible Euler equations. In: Proceedings of AIAA 21st Aerospace Sciences Meeting, AIAA Paper 83-0125, Reno, Nevada, 1983. [CrossRef]

[22] Hughes, T.J.R. and Tezduyar, T.E. Finite element methods for first-order hyperbolic systems with particular emphasis on the compressible Euler equations. *Computer Methods in Applied Mechanics and Engineering*, 45(1-3), 217-284, (1984). [CrossRef]

[23] Hughes, T.J.R., Franca, L.P. and Mallet, M. A new finite element formulation for computational fluid dynamics: VI. Convergence analysis of the generalized SUPG formulation for linear time-dependent multidimensional advective-diffusive systems. *Computer Methods in Applied Mechanics and Engineering*, 63(1), 97-112, (1987). [CrossRef]

[24] Le Beau, G.J. and Tezduyar, T.E. Finite element computation of compressible flows with the SUPG formulation. In: *Advances in Finite Element Analysis in Fluid Dynamics*, (pp. 21-27). FED-Vol. 123, New York, USA: ASME, (1991).

[25] Tezduyar, T.E. and Park, Y. Discontinuity-capturing finite element formulations for non-linear convection-diffusion-reaction equations. *Computer Methods in Applied Mechanics and Engineering*, 59(3), 307-325, (1986). [CrossRef]

[26] Tezduyar, T.E. Finite elements in fluids: Stabilized formulations and moving boundaries and interfaces. *Computers & Fluids*, 36(2), 191-206, (2003). [CrossRef]

[27] Tezduyar, T.E. Finite element methods for fluid dynamics with moving boundaries and interfaces. In: E Stein, RD Borst, TJR Hughes (Eds.), *Encyclopedia of Computational Mechanics*, Volume 3: Fluids, Wiley, (2004). [CrossRef]

[28] Rispoli, F., Corsini, A. and Tezduyar, T.E. Finite element computation of turbulent flows with the discontinuity-capturing directional dissipation (DCDD). *Computers & Fluids*, 36(1), 121-126, (2007). [CrossRef]

[29] Tezduyar, T.E. Computation of moving boundaries and interfaces and stabilization parameters. *International Journal for Numerical Methods in Fluids*, 43(5), 555-575, (2003). [CrossRef]

[30] Tezduyar, T.E. and Senga, M. Stabilization and shock-capturing parameters in SUPG formulation of compressible flows. *Computer Methods in Applied Mechanics and Engineering*, 195(13-16), 1621-1632, (2006). [CrossRef]

[31] Tezduyar, T.E. and Senga, M. SUPG finite element computation of inviscid supersonic flows with YZβ shock-capturing. *Computers & Fluids*, 36(1), 147-159, (2007). [CrossRef]

[32] Tezduyar, T.E., Senga, M. and Vicker, D. Computation of inviscid supersonic flows around cylinders and spheres with the SUPG formulation and YZβ shock-capturing. *Computational Mechanics*, 38, 469-481, (2006). [CrossRef]

[33] Bazilevs, Y., Calo, V.M., Tezduyar, T.E. and Hughes, T.J.R. YZβ discontinuity capturing for advection-dominated processes with application to arterial drug delivery. *International Journal*

for Numerical Methods in Fluids, 54(6-8), 593-608, (2007). [CrossRef]

[34] Cengizci, S. and Uğur, Ö. A stabilized FEM formulation with discontinuity-capturing for solving Burgers'-type equations at high Reynolds numbers. *Applied Mathematics and Computation*, 442, 127705, (2023). [CrossRef]

[35] Cengizci, S. and Uğur, Ö. SUPG formulation augmented with YZβ shock-capturing for computing shallow-water equations. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, 103(6), e202200232, (2023). [CrossRef]

[36] Cengizci, S., Uğur, Ö. and Natesan, S. A SUPG formulation augmented with shock-capturing for solving convection-dominated reaction-convection-diffusion equations. *Computational and Applied Mathematics*, 42, 235, (2023). [CrossRef]

[37] Tezduyar, T.E., Ramakrishnan, S. and Sathe, S. Stabilized formulations for incompressible flows with thermal coupling. *International Journal for Numerical Methods in Fluids*, 57(9), 1189-1209, (2008). [CrossRef]

[38] John, V. and Knobloch, P. On spurious oscillations at layers diminishing (SOLD) methods for convection-diffusion equations: Part I-A review. *Computer Methods in Applied Mechanics and Engineering*, 196(17-20), 2197-2215, (2007). [CrossRef]

[39] John, V. and Knobloch, P. On spurious oscillations at layers diminishing (SOLD) methods for convection-diffusion equations: Part II-Analysis for $P1$ and $Q1$ finite elements. *Computer Methods in Applied Mechanics and Engineering*, 197(21-24), 1997-2014, (2008). [CrossRef]

[40] John, V., Knobloch, P. and Novo, J. Finite elements for scalar convection-dominated equations and incompressible flow problems: a never ending story?. *Computing and Visualization in Science*, 19, 47-63, (2018). [CrossRef]

[41] Ak, T., Karakoç, S.B.G. and Biswas, A. Application of Petrov-Galerkin finite element method to shallow water waves model: Modified Korteweg-de Vries equation. *Scientia Iranica B*, 24(3), 1148-1159, (2017). [CrossRef]

[42] Bhowmik, S.K. and Karakoc, S.B.G. Numerical approximation of the generalized regularized long wave equation using Petrov-Galerkin finite element method. *Numerical Methods for Partial Differential Equations*, 35(6), 2236-2257, (2019). [CrossRef]

[43] Karakoc, S.B.G., Saha, A. and Sucu, D. A novel implementation of Petrov-Galerkin method to shallow water solitary wave pattern and superperiodic traveling wave and its multistability: Generalized Korteweg-de Vries equation. *Chinese Journal of Physics*, 68, 605-617, (2020). [CrossRef]

[44] Shakib, F. *Finite element analysis of the compressible Euler and Navier-Stokes equations*. Ph.D. Thesis, Department of Mechanical Engineering, Stanford University: California, (1988).

[45] Franca, L.P., Frey, S.L. and Hughes, T.J. Stabilized finite element methods: I. Application to the advective-diffusive model. *Computer Methods in Applied Mechanics and Engineering*, 95(2), 253-276, (1992). [CrossRef]

[46] Tezduyar, T.E. Stabilized finite element formulations for incompressible flow computations. *Advances in Applied Mechanics*, 28, 1-44, (1991). [CrossRef]

[47] Donea, J. and Huerta, A. *Finite Element Methods for Flow Problems*. John Wiley & Sons: England, (2003). [CrossRef]

[48] Abali, B.E. *Computational Reality: Solving Nonlinear and Coupled Problems in Continuum Mechanics.* Advanced Structured Materials, Springer: Singapore, (2016). [CrossRef]

[49] Alnæs, M., Blechta, J., Hake, J., Johansson, A., Kehlet, B., Logg, A. et al. The FEniCS project version 1.5. *Archive of Numerical Software*, 3(100), 9-23, (2015). [CrossRef]

[50] Logg, A., Mardal, K.A. and Wells, G. *Automated solution of differential equations by the finite element method: The FEniCS book* (Vol. 84). Lecture Notes in Computational Science and Engineering, Springer-Verlag: Berlin, Heidelberg, (2012). [CrossRef]

[51] Zhang, J. and Yan, G. Lattice Boltzmann simulation of pattern formation under cross-diffusion. *Computers & Mathematics with Applications*, 69(3), 157-169, (2015). [CrossRef]

[52] Schnakenberg, J. Simple chemical reaction systems with limit cycle behaviour. *Journal of Theoretical Biology*, 81(3), 389-400, (1979). [CrossRef]

[53] Garzón-Alvarado, D.A., Galeano, C.H. and Mantilla, J.M. Computational examples of reaction-convection-diffusion equations solution under the influence of fluid flow: First example. *Applied Mathematical Modelling*, 36(10), 5029-5045, (2013). [CrossRef]

[54] Koppel, J.V.D., Rietkerk, M., Dankers, N. and Herman, P.M.J. Scale-dependent feedback and regular spatial patterns in young mussel beds. *The American Naturalist*, 165(3), E66-E77, (2005). [CrossRef]

[55] Jones, D.A., Smith, H.L., Dung, L. and Ballyk, M. Effects of random motility on microbial growth and competition in a flow reactor. *SIAM Journal on Applied Mathematics*, 59(2), 573-596, (1998). [CrossRef]

[56] Murray, J.D. *Mathematical Biology II: Spatial Models and Biomedical Applications. Interdisciplinary Applied Mathematics*, New York, NY, USA: Springer, (2013). [CrossRef]

[57] Vanegas, J.C., Landinez, N.S. and Garzón-Alvarado, D.A. Modelo matemático de la coagulación en la interfase hueso implante dental. *Revista Cubana de Investigaciones Biomédicas*, 28(3), 167-191, (2009).

Mathematical Modelling and Numerical Simulation with Applications (MMNSA)
(https://dergipark.org.tr/en/pub/mmnsa)

How to cite this article: Cengizci, S. (2023). An enhanced SUPG-stabilized finite element formulation for simulating natural phenomena governed by coupled system of reaction-convection-diffusion equations. *Mathematical Modelling and Numerical Simulation with Applications*, 3(4), 297-317. https://doi.org/10.53391/mmnsa.1387125