



Universiteit Utrecht

DEPARTMENT OF MATHEMATICS

BACHELOR THESIS

**Standing and rotating waves in
reaction-diffusion systems with circular
symmetry**

Shashi Chotkan

supervised by
Prof. dr. Yuri A. KUZNETSOV
Dr. P.A. ZEGELING

June 12, 2020

Abstract

Many academic papers are written about reaction-diffusion systems with circular symmetry but without explicit formulas for normal form coefficients and without numerical simulations to verify normal form predictions. In this paper, we will provide explicit formulas to compute normal form coefficients using general techniques. Additionally, we provide an example and predict the behavior using the computed normal form coefficients. And then we will simulate the system numerically in MATLAB and compare results with the prediction.

Contents

1	Introduction	3
2	Differential equation on center manifold	5
3	Coefficients of the normal form	9
4	Numerical simulation	12
5	Example	13
6	Discussion	16
7	Appendix	17
	References	23

1 Introduction

1.1 Motivation

Reaction-diffusion systems appear in many fields of Science, see e.g. the classical book [1]. They are often used in Life Sciences, all the way from population dynamics to modeling chemical reactions. Specific examples of these applications can be found in [2].

The reaction-diffusion systems allow describing pattern formation, e.g. the appearance of stationary and traveling waves. Changing parameters in the system can produce different solutions with different spatial characteristics. Finding these critical points and studying solutions appearing nearby can give a scientist a lot of information about the studied system. A particular example of these critical points is Hopf bifurcation points, where small-amplitude periodic solutions appear when a stationary solution exhibits an oscillatory instability.

Hopf bifurcation is a well-understood phenomenon in models without diffusion described by finite-dimensional ordinary differential equations, see [3]. Reaction-diffusion systems – even on bounded domains – are the infinite-dimensional dynamical systems. Moreover, their local bifurcation analysis is complicated by the multiplicity of the eigenvalues of the linear part, which typically occurs if the domain is symmetric, e.g. invariant w.r.t. rotations.

1.2 Aims

In what follows, we consider only the simplest rotationally symmetric reaction-diffusion systems, i.e. those on the unit circle S^1 . Small-amplitude time-periodic solutions appearing in such systems via Hopf bifurcation were studied by many authors, in particular in [4, 5]. Periodic solutions near degenerate Hopf bifurcations were also analyzed. To obtain a characterization of all periodic and non-periodic small-amplitude solutions, an approach based on the Center Manifold reduction was first applied in [6, 7] and later by many authors, including [8]. A reaction-diffusion system with the rotational symmetry has been recently used as an example in [9], where a symbolic/numerical algorithm to compute the normal forms of PDEs on their center manifolds is proposed and implemented.

However, in all papers – except the hardly available [6] – no directly applicable formulas for the normal form coefficients at the Hopf bifurcation in reaction-diffusion systems on the circle were given. Moreover, no detailed normal form computations predicting the appearance of stable waves via the primary Hopf bifurcation in a concrete reaction-diffusion system were demonstrated and verified by simulations.

Thus, the aim of this thesis is threefold:

- (1) Revisit the normal form computations of [6–8] and derive as explicit as possible formulas for the critical normal form coefficients of the 4-dimensional symmetric normal form at the primary Hopf bifurcation in general reaction-diffusion systems on the circle.
- (2) Implement the derived formulas in MATLAB and apply them to a three-component reaction-diffusion system originally proposed by Polyakova in [10].
- (3) Develop and implement a finite-difference method to numerically simulate general multi-component reaction-diffusion systems on the circle, and verify the normal form predictions on Polyakova’s example.

1.3 Mathematical problem

Consider the following reaction-diffusion system for $\mathbf{u} : S^1 \times [0, \infty) \rightarrow \mathbb{R}^m$ satisfying

$$\frac{\partial \mathbf{u}}{\partial t} = D(\mu) \frac{\partial^2 \mathbf{u}}{\partial \theta^2} + F(\mathbf{u}, \mu) \quad (\theta \bmod 2\pi), \quad (1)$$

where $D(\mu)$ is a diagonal $m \times m$ matrix with positive elements smoothly depending on parameter $\mu \in \mathbb{R}$, and $F : \mathbb{R}^m \times \mathbb{R} \rightarrow \mathbb{R}^m$ is a smooth vector function, such that $\mathbf{u} = 0$ is a stationary homogeneous solution of (1) for all parameter values, i.e. $F(0, \mu) \equiv 0$.

As shown in [11], the system (1) generates a smooth (and smoothly depending on the parameter) local semiflow Φ_μ^t on (the real subspace of) the complex Hilbert space H , the completion of the space $C^2(S^1, \mathbb{C}^m)$ of twice continuously-differentiable vector-functions on the circle S^1 under the norm corresponding to the inner product

$$\langle u, v \rangle = \sum_{k=1}^m \int_0^{2\pi} \left(\bar{u}_k v_k + \frac{d\bar{u}_k}{d\theta} \frac{dv_k}{d\theta} + \frac{d^2 \bar{u}_k}{d\theta^2} \frac{d^2 v_k}{d\theta^2} \right) dx. \quad (2)$$

The operator of the linearization of (1) at the stationary solution $\mathbf{u} = 0$ is

$$L(\mu) = D(\mu) \frac{\partial^2}{\partial \theta^2} + A(\mu),$$

where $A(\mu) = F_{\mathbf{u}}(0, \mu)$. The spectrum of $L(\mu)$ in H consists of eigenvalues. Due to the S^1 -symmetry of the problem, the eigenvalues of $L(\mu)$ are generically double, with two linearly-independent eigenfunctions each.

Let $L_0 = L(0)$, $D_0 = D(0)$, $A_0 = A(0)$. Suppose that there is a unique $k > 0$ such that the $m \times m$ matrix

$$-k^2 D_0 + A_0$$

has a simple pair of purely imaginary eigenvalues $\pm i\omega_0$, $\omega_0 > 0$. Let $V \in \mathbb{C}^m$ be the corresponding eigenvector, i.e.

$$(-k^2 D_0 + A_0)V = i\omega_0 V.$$

Note that vector V is not uniquely defined: One can multiply it by a nonzero complex number. Given V , it is easy to verify that the vector-functions

$$\varphi_{1,2}(\theta) = V e^{\pm ik\theta} \tag{3}$$

and their complex-conjugate will be the eigenfunctions of L_0 corresponding to the pair of double eigenvalues $\lambda_{1,2} = \pm i\omega_0$:

$$L_0 \varphi_{1,2} = i\omega_0 \varphi_{1,2} \quad \text{and} \quad L_0 \bar{\varphi}_{1,2} = -i\omega_0 \bar{\varphi}_{1,2}. \tag{4}$$

Suppose that for small $|\mu|$ the operator $L(\mu)$ has a double pair of complex eigenvalues $\lambda_1(\mu) = \lambda(\mu)$ and $\lambda_2(\mu) = \overline{\lambda(\mu)}$, where

$$\lambda(\mu) = \mu + i\omega(\mu)$$

with $\omega(0) = \omega_0$. This means that we use $\text{Re}(\lambda_{1,2})$ as the system parameter. We aim is to study solutions of (1) in a small neighborhood of the origin in H , when μ changes sign. The case when all eigenvalues of $L(\mu)$ except $\lambda_{1,2}(\mu)$ have negative real parts is of particular interest in applications since it corresponds to ‘‘diffusion oscillatory instability’’ of the trivial steady state when μ changes from negative to positive. Note that this is only possible if $m \geq 3$.

2 Differential equation on center manifold

2.1 Center manifold reduction

To the local semiflow $\Phi_\mu^t : H \rightarrow H$ generated by the reaction-diffusion system (1) the Center Manifold Theorem is applicable [11]. In this case, it guarantees the existence for all sufficiently small $|\mu|$ of a real 4-dimensional locally invariant for Φ_μ^t smooth manifold $W_\mu^c \subset H$, which depends smoothly on μ . This manifold is called the *center manifold* and is tangent at $\mu = 0$ to the 4-dimensional linear invariant subspace $H_0 \subset H$ of L_0 , corresponding to the double pair of its eigenvalues $\pm i\omega_0$. The manifold W_μ^c is locally normally hyperbolic. If all eigenvalues of $L(\mu)$ have negative real parts for $\mu < 0$, the center manifold is locally attracting for Φ_μ^t .

Any $\varphi \in H_0$ can be written as

$$\varphi = w_1\varphi_1 + w_2\varphi_2 + \bar{w}_1\bar{\varphi}_1 + \bar{w}_2\bar{\varphi}_2, \quad (5)$$

where $w_{1,2} \in \mathbb{C}$. A smooth projection of W_μ^c onto H_0 allows to use (w_1, w_2) as local coordinates in W_μ^c . In these coordinates, the restriction of Φ_μ^t to W_μ^c is a local flow generated by a smooth system of two complex ODEs

$$\begin{cases} \dot{w}_1 = \lambda(\mu)w_1 + f_1(w_1, w_2, \mu), \\ \dot{w}_2 = \lambda(\mu)w_2 + f_2(w_1, w_2, \mu), \end{cases} \quad (6)$$

where $f_j = O(\|w\|^2)$. One can show that due to the S^1 -symmetry of the original reaction-diffusion problem, this system does not change under the transformations

$$\begin{aligned} (1) \quad P_\alpha(w_1, w_2) &= (e^{ik\alpha}w_1, e^{-ik\alpha}w_2) \quad (\alpha \bmod 2\pi), \\ (2) \quad P_r(w_1, w_2) &= (w_2, w_1), \end{aligned} \quad (7)$$

corresponding to the rotation of the unit circle $\theta \mapsto \theta + \alpha$ and to its reflection $\theta \mapsto -\theta$, respectively.

Moreover, there is an invertible smooth change of the complex variables that preserves the symmetry (7) of (6) and transforms it into the normal form

$$\begin{cases} \dot{z}_1 = z_1 (\lambda(\mu) + G(\mu)|z_1|^2 + H(\mu)|z_2|^2) + g(z_1, z_2, \mu), \\ \dot{z}_2 = z_2 (\lambda(\mu) + H(\mu)|z_1|^2 + G(\mu)|z_2|^2) + g(z_2, z_1, \mu), \end{cases} \quad (8)$$

where G, H are smooth complex-valued functions of μ , while $g = O(\|z\|^5)$ is a smooth (but not necessarily analytic) function satisfying

$$e^{-i\beta}g(z_1e^{i\beta}, z_2e^{-i\beta}, \mu) = g(z_1, z_2, \mu) \quad (\beta \bmod 2\pi).$$

This normal form appeared in several studies, starting with [6, 7].

2.2 Analysis of the truncated normal form

Consider the following system of complex differential equations

$$\begin{cases} \dot{z}_1 = z_1 (\mu + i\omega + G|z_1|^2 + H|z_2|^2), \\ \dot{z}_2 = z_2 (\mu + i\omega + H|z_1|^2 + G|z_2|^2), \end{cases} \quad (9)$$

where $\mu \in \mathbb{R}$ and $G, H \in \mathbb{C}$. This systems is obtained from the full normal form (8) by truncating the $O(\|z\|^5)$ -terms, and not indicating the dependence of G, H and ω on μ for simplicity. One can show that adding any symmetric $O(\|z\|^5)$ -terms does not change the qualitative properties of (9) near the bifurcation [12, 13].

We would like to analyse solutions of (9) near the equilibrium point $z_1 = z_2 = 0$. We can do this by first substituting $z_j = \rho_j e^{i\phi_j}$ to convert to polar coordinates. Here ρ_j represents the amplitude of z_j and ϕ_j represents the angle with the positive real axis. This gives the following equations

$$\begin{cases} \dot{\rho}_1 e^{i\phi_1} + i\dot{\phi}_1 \rho_1 e^{i\phi_1} = \rho_1 e^{i\phi_1} (\mu + i\omega + G\rho_1^2 + H\rho_2^2), \\ \dot{\rho}_2 e^{i\phi_2} + i\dot{\phi}_2 \rho_2 e^{i\phi_2} = \rho_2 e^{i\phi_2} (\mu + i\omega + H\rho_1^2 + G\rho_2^2). \end{cases}$$

By decomposing G, H into real and pure imaginary parts, that is $G = G_R + iG_I$, $H = H_R + iH_I$ where $G_R, G_I, H_R, H_I \in \mathbb{R}$ we can separate the real and the imaginary components of the equations. This gives

$$\begin{cases} \dot{\rho}_1 = \mu\rho_1 + G_R\rho_1^3 + H_R\rho_1\rho_2^2, \\ \dot{\rho}_2 = \mu\rho_2 + H_R\rho_1^2\rho_2 + G_R\rho_2^3, \\ \dot{\phi}_1 = \omega + G_I\rho_1^2 + H_I\rho_2^2, \\ \dot{\phi}_2 = \omega + H_I\rho_1^2 + G_I\rho_2^2. \end{cases} \quad (10)$$

The trivial equilibrium $\rho_1 = \rho_2 = 0$ of (10) corresponds to the spatially uniform trivial solution of the reaction-diffusion system (1). The periodic solutions with either $\rho_1 = 0$ or $\rho_2 = 0$ of (10) correspond to the *rotating waves* in (1), propagating in opposite directions. Finally, the periodic solutions with $\rho_{1,2} > 0$, which compose a two-dimensional invariant torus in (10), correspond to a family of *standing waves* with different positions of nodes in (1). These facts become evident upon substituting the periodic solutions of the system into the parametrization (5) of H_0 . Moreover, the stability of each solution in the reaction-diffusion system can be deduced from that in (10).

Since the amplitude equations split from the equations for $\phi_{1,2}$, we will omit the last two equations. Setting $a = G_R$ and $b = H_R$ gives the following cubic real system

$$\begin{cases} \dot{\rho}_1 = \mu\rho_1 + a\rho_1^3 + b\rho_1\rho_2^2, \\ \dot{\rho}_2 = \mu\rho_2 + b\rho_1^2\rho_2 + a\rho_2^3. \end{cases} \quad (11)$$

Notice that there exist a scalar potential function $\phi : \mathbb{R}^2 \rightarrow \mathbb{R}$ namely

$$\phi(\rho) = \frac{1}{4} [a(\rho_1^4 + \rho_2^4) + 2\mu(\rho_1^2 + \rho_2^2) + 2b\rho_1^2\rho_2^2],$$

such that $\dot{\rho} = \nabla\phi$. Thus the planar amplitude system (11) is gradient and cannot have periodic solutions.

We can simplify system (11) by the substitution of variables

$$\begin{aligned} r_1 &= \rho_1^2, \\ r_2 &= \rho_2^2. \end{aligned} \quad (12)$$

so that $\dot{r}_j = 2\rho_j\dot{\rho}_j$, $j = 1, 2$. This gives the quadratic system

$$\begin{cases} \dot{r}_1 = 2r_1(\mu + ar_1 + br_2), \\ \dot{r}_2 = 2r_2(\mu + br_1 + ar_2), \end{cases} \quad (13)$$

with the equilibria

$$\begin{aligned} r_1 = 0 & & r_2 = 0 \\ r_1 = 0 & & r_2 = -\frac{\mu}{a} \\ r_1 = -\frac{\mu}{a} & & r_2 = 0 \\ r_1 = -\frac{a\mu}{a+b} & & r_2 = -\frac{\mu}{a+b} \end{aligned} .$$

For these equilibria we do have to make sure that their coordinates are not negative. The Jacobian matrix of the system (12) is

$$J = 2 \begin{pmatrix} \mu + 2ar_1 + br_2 & br_1 \\ br_2 & \mu + 2ar_2 + br_1 \end{pmatrix}.$$

We would like to analyse equilibria of (12). Their coordinates, eigenvalues, and eigenvectors of the corresponding Jacobian matrix J are given in Table 1.

Indeed, only equilibria with non-negative coordinates have to be considered. Generically, stability of an equilibrium depend on the eigenvalues. This means we have to distinguish different combinations of signs of

p	E_0	E_1	E_2	E_3
r	$(0, 0)$	$(0, -\frac{\mu}{a})$	$(-\frac{\mu}{a}, 0)$	$(-\frac{\mu}{a+b}, -\frac{\mu}{a+b})$
J	$2 \begin{pmatrix} \mu & 0 \\ 0 & \mu \end{pmatrix}$	$-\frac{2\mu}{a} \begin{pmatrix} b-a & 0 \\ b & a \end{pmatrix}$	$-\frac{2\mu}{a} \begin{pmatrix} a & b \\ 0 & b-a \end{pmatrix}$	$-\frac{2\mu}{a+b} \begin{pmatrix} a & b \\ b & a \end{pmatrix}$
λ_1	2μ	-2μ	-2μ	-2μ
λ_2	2μ	$\frac{2\mu(a-b)}{a}$	$\frac{2\mu(a-b)}{a}$	$-\frac{2\mu(a-b)}{a+b}$
v_1	$(1, 0)$	$(0, 1)$	$(1, 0)$	$(1, 1)$
v_2	$(0, 1)$	$(-\frac{2a-b}{b}, 1)$	$(-\frac{b}{2a+b}, 1)$	$(-1, 1)$

Table 1: Equilibria of (12) with the corresponding Jacobian matrix, eigenvalue and eigenvector.

Case	Region
A	$a < 0, a - b > 0, a + b < 0$;
B	$a < 0, a - b < 0, a + b < 0$;
C	$a < 0, a - b < 0, a + b > 0$.

Table 2: Different cases of stability in the phase diagrams.

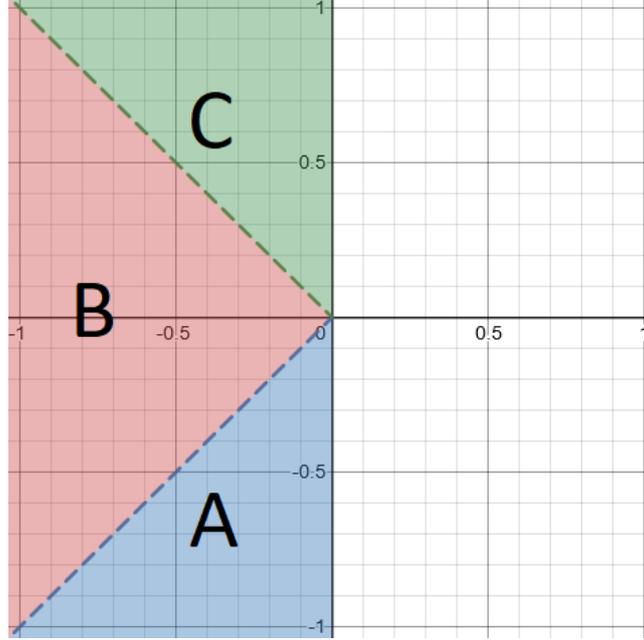


Figure 1: These regions distinguish cases of the stability in the phase diagrams. Here the horizontal axis is a and the vertical axis is b .

the eigenvalues for different parameter values a, b and μ . Thus, we have to distinguish the following regions in the (a, b) -plane (see Table 2 and Figure 1):

In Case A, see Figure 2, the equilibrium E_0 in system (12) is locally asymptotically stable for $\mu \leq 0$ but becomes unstable for $\mu > 0$, when two equilibria $E_{1,2}$ on the axes appear together with the positive equilibrium E_3 . The equilibria $E_{1,2}$ are locally asymptotically stable, while the positive one is a saddle. For the reaction-diffusion system (1), this means that for $\mu \leq 0$ we have only the stable uniform solution, while for $\mu > 0$ this solution becomes unstable, but two asymptotically stable and rotating in opposite directions waves, and a family of unstable standing waves, appear. The waves have small amplitude that tends to zero as $\mu \downarrow 0$.

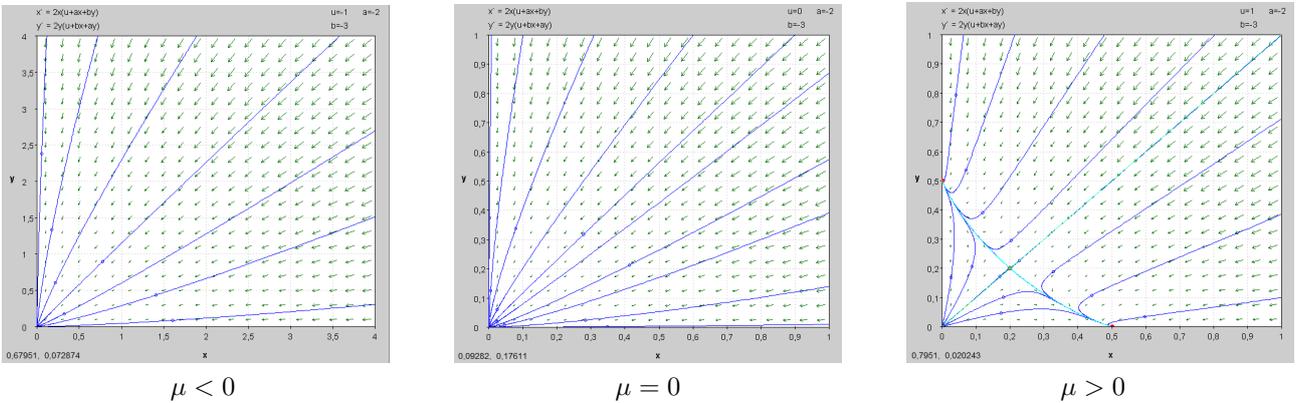


Figure 2: Bifurcation diagram in case A. ¹

In Case B, see Figure 3, the local stability of the trivial equilibrium in (12) is for small $|\mu|$ as in Case A. However, for $\mu > 0$, the equilibrium points $E_{1,2}$ are saddles, while the equilibrium E_3 is asymptotically stable.

¹Note that the phase portraits have been generated using PPLANE9

For the reaction-diffusion system (1), this means that for $\mu \leq 0$ we still have only the stable uniform solution, which becomes unstable for $\mu > 0$. However, for small $\mu > 0$ the rotating waves are unstable, while the standing waves are stable.

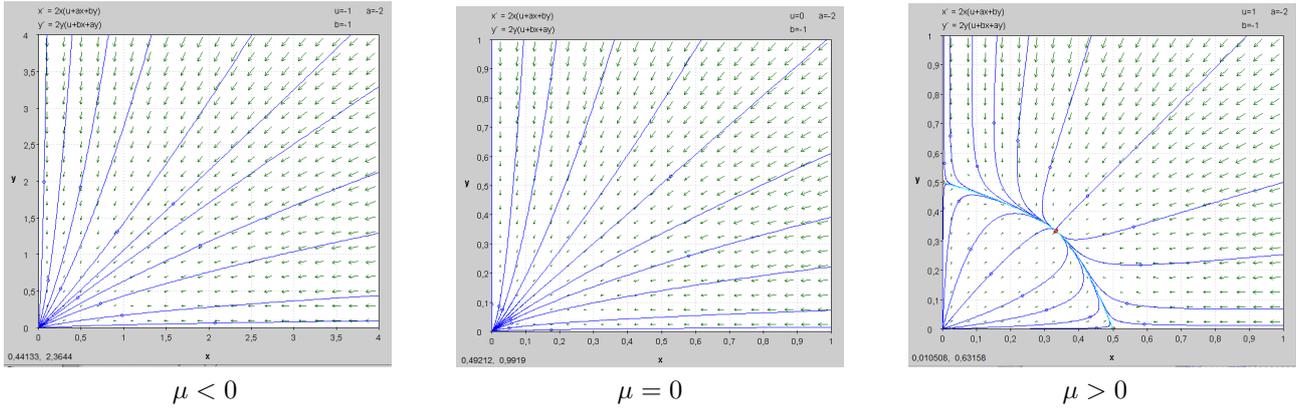


Figure 3: Bifurcation diagram in case B.

Case C presented in Figure 4 is rather different. For $\mu < 0$ the unstable trivial equilibrium E_0 in (12) coexists with a positive saddle E_3 . This saddle merges at $\mu = 0$ with E_0 forming an unstable equilibrium, while for $\mu > 0$ the equilibrium E_3 remains unstable but coexists with two saddles $E_{1,2}$. For the reaction-diffusion system (1), this means that for $\mu < 0$, the unstable uniform solution coexists with unstable standing waves, while for $\mu > 0$, two rotating waves appear but they are both unstable. Thus, no stable small amplitude solutions exist for all small $|\mu|$.

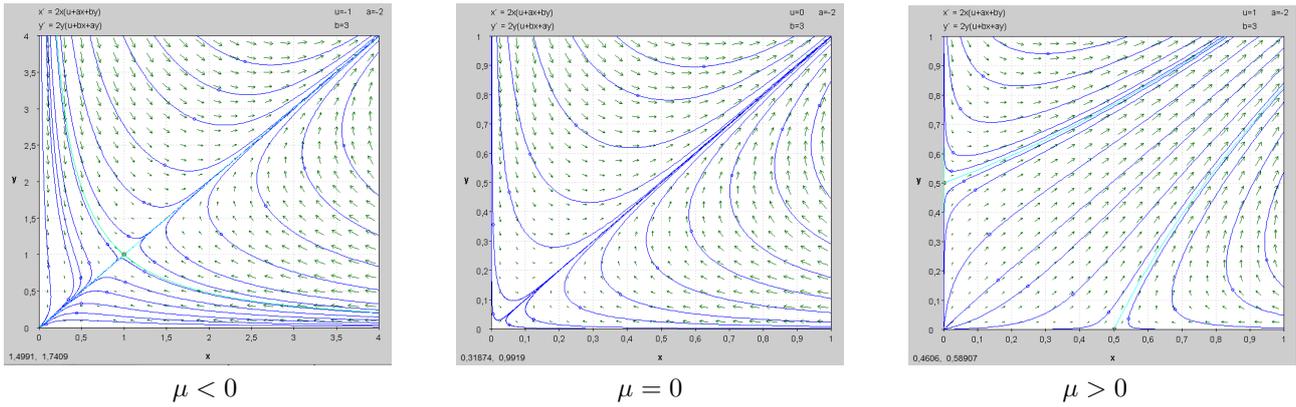


Figure 4: Bifurcation diagram in case C.

3 Coefficients of the normal form

Since the critical values (a_0, b_0) determine the rearrangements of the phase portraits, it is sufficient to compute the normal form coefficients (a, b) at $\mu = 0$. We follow the procedure applied in [3] for the double Hopf bifurcation in ODEs, which is considerably simpler than that of [8], where however higher-order normal forms were studied.

Consider the reaction-diffusion system (1) at the critical parameter value $\mu = 0$ and write the Taylor expansion of $F(\mathbf{u}, 0)$ as

$$F(\mathbf{u}, 0) = A_0 u + \frac{1}{2} B(\mathbf{u}, \mathbf{u}) + \frac{1}{6} C(\mathbf{u}, \mathbf{u}, \mathbf{u}) + O(\|\mathbf{u}\|^4).$$

Here $A_0 = DF(0, 0)$ and the components of the multi-linear functions can be written as

$$\begin{aligned} B_i(\mathbf{v}, \mathbf{w}) &= \sum_{j,k=1}^m \frac{\partial^2 F_i(0, 0)}{\partial u_j \partial u_k} v_j w_k, \\ C_i(\mathbf{v}, \mathbf{w}, \mathbf{z}) &= \sum_{j,k,l=1}^m \frac{\partial^3 F_i(0, 0)}{\partial u_j \partial u_k \partial u_l} v_j w_k z_l, \end{aligned}$$

for $i = 1, 2, \dots, m$.

Write the system (1) near the equilibrium $u = 0$ as an abstract ODE in the Hilbert space H :

$$\frac{du}{dt} = L_0 u + \frac{1}{2} B(u, u) + \frac{1}{6} C(u, u, u) + O(\|u\|^4), \quad (14)$$

where

$$L_0 = D_0 \frac{\partial^2}{\partial \theta^2} + A_0$$

is a linear operator in H with a suitable definition domain, and B and C are naturally defined smooth multilinear functions on $H \times H$ and $H \times H \times H$, respectively.

The adjoint operator L_0^* satisfies

$$\langle u, L_0 v \rangle = \langle L_0^* u, v \rangle,$$

where the inner product (2) is used. Explicitly,

$$L_0^* = D_0 \frac{\partial^2}{\partial \theta^2} + A_0^T,$$

with the same definition domain as L_0 . This operator also has two purely imaginary eigenvalues $\lambda_{1,2} = \pm i\omega_0$, $\omega_0 > 0$, such that for $\lambda_1 = -i\omega_0$ there are two linearly-independent eigenfunctions $\psi_{1,2}$, such that

$$L_0^* \psi_1 = -i\omega_0 \psi_1 \quad \text{and} \quad L_0^* \psi_2 = -i\omega_0 \psi_2. \quad (15)$$

Then

$$\psi_{1,2}(\theta) = W e^{\pm ik\theta} \quad k > 0, \quad (16)$$

where $W \in \mathbb{C}^m$ satisfies

$$(-k^2 D_0 + A_0^T) W = -i\omega_0 W. \quad (17)$$

We select the eigenfunctions so that

$$\langle \psi_l, \varphi_j \rangle = \delta_{lj},$$

which can be achieved by normalizing $V, W \in \mathbb{C}^m$ such that

$$\overline{W}^T V = \frac{1}{2\pi(1 + k^2 + k^4)}.$$

Define

$$U = 2\pi(1 + k^2 + k^4)W, \quad (18)$$

i.e. take the eigenvector $U \in \mathbb{C}^m$ of matrix $(-k^2 D_0 + A_0^T)$ corresponding to the eigenvalue $(-i\omega_0)$ and normalized such that

$$\overline{U}^T V = 1. \quad (19)$$

The critical center manifold W_0^c is tangent to H_0 and can be parametrized by $(z_1, z_2) \in \mathbb{C}^2$. To compute the cubic normal form coefficients, we need its approximation

$$u = z_1 \varphi_1 + z_2 \varphi_2 + \bar{z}_1 \bar{\varphi}_1 + \bar{z}_2 \bar{\varphi}_2 + \sum_{j+k+l+r \geq 2} \frac{1}{j!k!l!r!} h_{jklr} z_1^j \bar{z}_1^k z_2^l \bar{z}_2^r, \quad (20)$$

where the complex-valued periodic functions $h_{ijkl} = h_{ijkl}(\theta)$ are yet to be defined, and

$$h_{0200} = \bar{h}_{2000}, \quad h_{0002} = \bar{h}_{0020}, \quad h_{0101} = \bar{h}_{1010}, \quad h_{0110} = \bar{h}_{1001}.$$

We can assume, however, that with the selected parametrization the restriction of the system to W_0^c has the normal form

$$\begin{cases} \dot{z}_1 = i\omega_0 z_1 + G_0 z_1^2 \bar{z}_1 + H_0 z_1 z_2 \bar{z}_2 + O(\|z\|^4), \\ \dot{z}_2 = i\omega_0 z_2 + H_0 z_1 z_2 \bar{z}_1 + G_0 z_2^2 \bar{z}_2 + O(\|z\|^4). \end{cases} \quad (21)$$

Notice that no quadratic terms are present in (21).

Substituting (20) into (14) using (21), and then collecting the quadratic in (z_1, z_2) terms, we get the following equations for h_{ijkl} with $i + j + k + l = 2$:

$$(2i\omega_0 - L_0)h_{2000} = B(\varphi_1, \varphi_1), \quad (22)$$

$$-L_0 h_{1100} = B(\varphi_1, \bar{\varphi}_1), \quad (23)$$

$$(2i\omega_0 - L_0)h_{0020} = B(\varphi_2, \varphi_2), \quad (24)$$

$$-L_0 h_{0011} = B(\varphi_2, \bar{\varphi}_2), \quad (25)$$

$$(2i\omega_0 - L_0)h_{1010} = B(\varphi_1, \varphi_2), \quad (26)$$

$$-L_0 h_{1001} = B(\varphi_1, \bar{\varphi}_2). \quad (27)$$

Using $\varphi_{1,2}$ from (3), we obtain

$$\begin{aligned} B(\varphi_1, \varphi_1) &= e^{2ik\theta} S, & B(\varphi_1, \bar{\varphi}_1) &= R, & B(\varphi_2, \varphi_2) &= e^{-2ik\theta} S, \\ B(\varphi_2, \bar{\varphi}_2) &= R, & B(\varphi_1, \varphi_2) &= S, & B(\varphi_1, \bar{\varphi}_2) &= e^{2ik\theta} R, \end{aligned}$$

where

$$S = B(V, V), \quad R = B(V, \bar{V}), \quad (28)$$

so that equations (22)-(27) can be solved explicitly. For example, (22) now reads

$$(2i\omega_0 - L_0)h_{2000} = e^{2ik\theta} S.$$

Recalling the definition of L_0 , we see that this is equivalent to

$$-D_0 \frac{d^2 h_{2000}}{d\theta^2} + (2i\omega_0 I_m - A_0)h_{2000} = e^{2ik\theta} S$$

with the unique solution

$$h_{2000}(\theta) = e^{2ik\theta} (2i\omega_0 I_m - A_0 + 4k^2 D_0)^{-1} S. \quad (29)$$

Here the matrix is invertible due to our basic spectral assumptions. Similarly, we obtain

$$h_{1100}(\theta) = -A_0^{-1} R, \quad (30)$$

$$h_{0020}(\theta) = e^{-2ik\theta} (2i\omega_0 I_m - A_0 + 4k^2 D_0)^{-1} S, \quad (31)$$

$$h_{0011}(\theta) = -A_0^{-1} R, \quad (32)$$

$$h_{1010}(\theta) = (2i\omega_0 I_m - A_0)^{-1} S, \quad (33)$$

$$h_{1001}(\theta) = e^{-2ik\theta} (-A_0 + 4k^2 D_0)^{-1} R. \quad (34)$$

Collecting the cubic terms, for which $j + k + l + r = 3$, we get the the following equations for h_{2100} and h_{1011} :

$$(i\omega_0 - L_0)h_{2100} = C(\varphi_1, \varphi_1, \bar{\varphi}_1) + B(h_{2000}, \bar{\varphi}_1) + 2B(h_{1100}, \varphi_1) - 2G_0 \varphi_1, \quad (35)$$

$$\begin{aligned} (i\omega_0 - L_0)h_{1011} &= C(\varphi_1, \varphi_2, \bar{\varphi}_2) + B(h_{1010}, \varphi_2) + B(h_{1001}, \bar{\varphi}_2) \\ &\quad + B(h_{0011}, \varphi_1) - H_0 \varphi_1. \end{aligned} \quad (36)$$

Take the inner product of ψ_1 in H with both sides of (35). The left-hand side gives

$$\begin{aligned} \langle \psi_1, (i\omega_0 - L_0)h_{2100} \rangle &= \langle (i\omega_0 - L_0)^* \psi_1, h_{2100} \rangle \\ &= \langle (-i\omega_0 - L_0^*) \psi_1, h_{2100} \rangle \\ &= -\langle L_0^* \psi_1 + i\omega_0 \psi_1, h_{2100} \rangle \\ &= 0. \end{aligned}$$

Thus, the product with the right-hand side of (35) must also vanish:

$$\langle \psi_1, C(\varphi_1, \varphi_1, \bar{\varphi}_1) + B(h_{2000}, \bar{\varphi}_1) + 2B(h_{1100}, \varphi_1) - 2G_0\varphi_1 \rangle = 0.$$

Using $\langle \psi_1, \varphi_1 \rangle = 1$, we obtain

$$G_0 = \frac{1}{2} \langle \psi_1, C(\varphi_1, \varphi_2, \bar{\varphi}_1) + B(h_{2000}, \bar{\varphi}_1) + 2B(h_{1100}, \varphi_1) \rangle. \quad (37)$$

Similarly, from (36) follows

$$H_0 = \langle \psi_1, C(\varphi_1, \varphi_2, \bar{\varphi}_2) + B(h_{1010}, \varphi_2) + B(h_{1001}, \bar{\varphi}_2) + B(h_{0011}, \varphi_1) \rangle. \quad (38)$$

In formulas (37) and (38), all quantities are defined before. These expressions correspond to the given without any comment coefficients a and b in [8, Table A.5]. A straightforward computation using (28), (29)–(33), and (19) now gives

$$\begin{aligned} G_0 &= \frac{1}{2} \bar{U}^T [C(V, V, \bar{V}) \\ &+ B((2i\omega_0 I_m - A_0 + 4k^2 D_0)^{-1} B(V, V), \bar{V}) \\ &- 2B(A_0^{-1} B(V, \bar{V}), V)], \end{aligned} \quad (39)$$

$$\begin{aligned} H_0 &= \bar{U}^T [C(V, V, V) \\ &+ B((2i\omega_0 I_m - A_0)^{-1} B(V, V), \bar{V}) \\ &+ B((-A_0 + 4k^2 D_0)^{-1} B(V, \bar{V}), V) - B(A_0^{-1} B(V, \bar{V}), V)]. \end{aligned} \quad (40)$$

The critical normal form coefficients can be computed by

$$a_0 = \operatorname{Re} G_0, \quad b_0 = \operatorname{Re} H_0.$$

4 Numerical simulation

In this section, we will show a way to solve the reaction-diffusion system numerically. The main procedure we will use is as follows. We first discretize the space S^1 to replace the problem of solving a system of a partial differential equation by that for a system of ordinary differential equations. Then we will put the equations in the standard form which enables us to easily integrate the system using standard software.

The spatial domain of (1), which is S^1 , can be discretized into N points using the following map

$$Q : \mathbb{N}_{N+1} \rightarrow S^1, \\ Q_k = Q(k) = \frac{2\pi(k-1)}{N}.$$

Here Q_1 and Q_{N+1} represent topologically the same point in S^1 , because $\{0\}$ and $\{2\pi\}$ are glued together in S^1 . Hence, only N unique points in S^1 are generated by Q .

While the other terms can be trivially transformed accordingly since they do not depend on space, special care however have to be taken to the second derivative. The second derivative $\frac{\partial^2 \mathbf{u}}{\partial \theta^2}$ can be approximated using the second-order central finite difference

$$\frac{\partial^2 \mathbf{u}_j(\theta, t)}{\partial \theta^2} \approx \frac{\mathbf{u}_j(\theta + h, t) - 2\mathbf{u}_j(\theta, t) + \mathbf{u}_j(\theta - h, t)}{h^2}.$$

We define $\mathbf{v} \in \mathbb{R}^{m \cdot N}$ as $\mathbf{v}_{i+N \cdot j} = \mathbf{u}_j(Q_i, t)$. Moreover we extend this definition as follows

$$\mathbf{v}_{1+N \cdot j} = \mathbf{v}_{(N+1)+N \cdot j}. \quad (41)$$

This gives the following correspondence with \mathbf{v} where we choose $h = \frac{2\pi}{N}$.

$$\begin{aligned} \frac{\partial^2 \mathbf{v}_{i+N \cdot j}}{\partial \theta^2} &= \frac{\partial^2 \mathbf{u}_j(Q_i, t)}{\partial \theta^2} \approx \frac{\mathbf{u}_j(Q_i + \frac{2\pi}{N}, t) - 2\mathbf{u}_j(Q_i, t) + \mathbf{u}_j(Q_i - \frac{2\pi}{N}, t)}{4\pi^2 N^{-2}} \\ &= \frac{N^2}{4\pi^2} (\mathbf{v}_{i+1+N \cdot j}(t) - 2\mathbf{v}_{i+N \cdot j}(t) + \mathbf{v}_{i-1+N \cdot j}(t)) \end{aligned}$$

This is defined for every i including $i = 1$ and $i = N$, because of the extended definition of $\mathbf{v}_{i+N \cdot j}$ in (41).

As mentioned before the other term $F(\cdot)$ does not depend on θ . This means that we have the following relation with $F(\mathbf{u})$

$$F(\mathbf{u}(Q_i, t)) = F(\mathbf{v}_{i+N \cdot 0}(t), \mathbf{v}_{i+N \cdot 1}(t), \mathbf{v}_{i+N \cdot 2}(t)).$$

Using the correspondence, we arrive at the following ordinary differential equation.

$$\begin{aligned} \mathbf{u} : S^1 \times [0, \infty) &\rightarrow \mathbb{R}^m \\ \frac{\partial \mathbf{u}}{\partial t} &= D(\mu) \frac{\partial^2 \mathbf{u}}{\partial \theta^2} + F(\mathbf{u}, \mu) \quad (\theta \bmod 2\pi) \\ &\Downarrow \\ \mathbf{v} : [0, \infty)^N &\rightarrow \mathbb{R}^{N \cdot m} \\ \frac{d\mathbf{v}_{i+N \cdot j}}{dt} &= \frac{D_{jj} N^2}{4\pi^2} (\mathbf{v}_{i+1+N \cdot j} - 2\mathbf{v}_{i+N \cdot j} + \mathbf{v}_{i-1+N \cdot j}) + F(\mathbf{v}_{i+N \cdot 0}, \mathbf{v}_{i+N \cdot 1}, \dots, \mathbf{v}_{i+N \cdot m}) \end{aligned}$$

We can write this in a more organised manner as follows where we have to take special care with the boundary as $\mathbf{v}_{1+N \cdot \star}$ and $\mathbf{v}_{(n+1)+N \cdot \star}$ refer to the same function.

That is the system can be written in matrix form as

$$\frac{d\mathbf{v}}{dt} = \frac{N^2}{4\pi^2} \begin{pmatrix} \mathbf{B}_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mathbf{B}_m \end{pmatrix} \begin{pmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_m \end{pmatrix} + \begin{pmatrix} F_1(\mathbf{v}_1, \dots, \mathbf{v}_m) \\ \vdots \\ F_m(\mathbf{v}_1, \dots, \mathbf{v}_m) \end{pmatrix}. \quad (42)$$

where $\mathbf{B}_j \in M_N(\mathbb{R})$ is defined as

$$\mathbf{B}_j = D_{jj} \begin{pmatrix} -2 & 1 & & & 1 \\ 1 & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & 1 \\ 1 & & & 1 & -2 \end{pmatrix}.$$

and $\mathbf{v}_j \in \mathbb{R}^N$ is the j -th part vector of \mathbf{v} partitioned by $(\mathbf{v}_j)_i = \mathbf{v}_{i+N \cdot j}$.

5 Example

In this section we will study a specific example of (1) with only three components, that is $m = 3$. The following reaction-diffusion system was introduced in [10] as an extension of the *Brusselator* [1]:

$$\begin{cases} \frac{\partial u_1}{\partial t} = d_1 \frac{\partial^2 u_1}{\partial \theta^2} + a - (b+1)u_1 + u_1^2 u_2 + u_2 u_3, \\ \frac{\partial u_2}{\partial t} = d_2 \frac{\partial^2 u_2}{\partial \theta^2} - u_1^2 u_2 + b u_1 - u_2 u_3, \\ \frac{\partial u_3}{\partial t} = d_3 \frac{\partial^2 u_3}{\partial \theta^2} + R(u_1 - u_2 u_3), \end{cases} \quad (43)$$

where $a, b, R, d_1, d_2, d_3 \in \mathbb{R}$. Thus

$$D = \begin{pmatrix} d_1 & 0 & 0 \\ 0 & d_2 & 0 \\ 0 & 0 & d_3 \end{pmatrix}, \quad F(u_1, u_2, u_3) = \begin{pmatrix} a - (b+1)u_1 + u_1^2 u_2 + u_2 u_3 \\ -u_1^2 u_2 + b u_1 - u_2 u_3 \\ R(u_1 - u_2 u_3) \end{pmatrix}.$$

The following numerical parameter values will be fixed later:

$$a = 2.5, \quad R = 0.135, \quad d_1 = 0.1, \quad d_2 = 0.01, \quad d_3 = 2.0,$$

while b will be a bifurcation parameter.

5.1 Critical value

A uniform equilibrium u_E of (43) can be found by solving for $F(\mathbf{u}) = 0$. This gives the algebraic system of equations

$$\begin{cases} 0 = a - (b+1)u_1 + u_1^2 u_2 + u_2 u_3, \\ 0 = -u_1^2 u_2 + b u_1 - u_2 u_3, \\ 0 = R(u_1 - u_2 u_3), \end{cases}$$

with the following solution

$$u_1 = a, \quad u_2 = \frac{b-1}{a}, \quad u_3 = \frac{a^2}{b-1}.$$

Now we can write $F(\mathbf{u} + \mathbf{v}) = A(\mathbf{u})\mathbf{v} + o(\|\mathbf{v}\|)$ where $A(\mathbf{u}) = DF(\mathbf{u})$. We have

$$A(\mathbf{u}) = DF(\mathbf{u}) = \begin{pmatrix} -(b+1) + 2u_1 u_2 & u_1^2 + u_3 & u_2 \\ -2u_1 u_2 + b & -u_1^2 - u_3 & -u_2 \\ R & -R u_3 & -R u_2 \end{pmatrix},$$

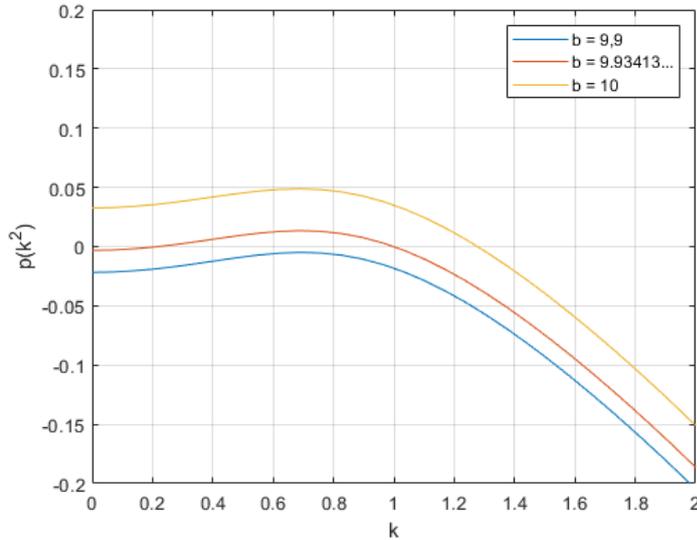


Figure 5: The real part of the complex eigenvalues of $A - k^2 D$

so that at the equilibrium u_E we have

$$A(u_E) = A\left(a, \frac{b-1}{a}, \frac{a^2}{b-1}\right) = \begin{pmatrix} b-3 & \frac{a^2b}{b-1} & \frac{b-1}{a} \\ -b+2 & -\frac{a^2b}{b-1} & -\frac{b-1}{a} \\ R & -\frac{Ra^2}{b-1} & -\frac{R(b-1)}{a} \end{pmatrix}.$$

Our goal now is to find k and b such that the matrix $(A - k^2D)$ has a pair of purely-imaginary eigenvalues $\pm i\omega_0$. For different values of b , the maximal real part of the eigenvalues of $A - k^2D$ is shown in Figure 5. This chart has been generated using MATLAB. The script can be found in the Appendix under the name CRITICAL PARAMETER CHART SCRIPT.

Using the CRITICAL PARAMETER FINDER script in the Appendix, a good approximation of the critical parameter can be found, which gives the value

$$b = 9.934138724791575.$$

This parameter has the special property that $p(1) = 0$ which gives us exactly the bifurcation point at the first harmonic. Moreover, since $p(0) < 0$, the uniform equilibrium is stable w.r.t. uniform perturbations, and since $p(k) < 0$ for $k = 2, 3, \dots$, it is also stable w.r.t. all other harmonic perturbations with $k = 2, 3, 4, \dots$

5.2 Computation of the normal form coefficients

To compute the normal form coefficients, we have provided a MATLAB script named NORMAL FORM COEFFICIENT SCRIPT in the Appendix ². It is assumed that the critical bifurcation parameter values are known. These values need to be assigned in the SYSTEM PARAMETERS section. The script produces the following eigenvectors for this example:

$$U = (-0.025 - 2.009i, 0.646 - 1.75i, -0.265 - 0.656i), \\ V = (-0.627 - 0.257i, 0.734, -0.036 + 0.024i).$$

By plugging these vectors in the formulas (39) and (40) – which the script does automatically – we get the following result

$$G(0) = -1.5893 - 0.6871i, \\ H(0) = -2.0947 - 0.1112i.$$

Which implies that

$$a_0 = -1.5893, \quad b_0 = -2.0947.$$

Thus, $a_0 < 0$ and $b_0 < 0$, implying that Case A takes place. Thus, we expect the appearance of two stable rotating waves and a family of unstable standing waves. In the next subsection, we confirm this prediction by direct simulations.

5.3 Numerical simulation

Following the numerical method described in Section 4, we get

$$\frac{d\mathbf{v}}{dt} = \frac{N^2}{4\pi^2} \begin{pmatrix} \mathbf{B}_1 & 0 & 0 \\ 0 & \mathbf{B}_2 & 0 \\ 0 & 0 & \mathbf{B}_3 \end{pmatrix} \begin{pmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \\ \mathbf{v}_3 \end{pmatrix} + \begin{pmatrix} F_1(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3) \\ F_2(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3) \\ F_3(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3) \end{pmatrix}, \quad (44)$$

where $\mathbf{B}_j \in M_N(\mathbb{R})$ are defined as

$$\mathbf{B}_j = D_{jj} \begin{pmatrix} -2 & 1 & & & 1 \\ 1 & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & 1 \\ 1 & & & 1 & -2 \end{pmatrix}$$

and $\mathbf{v}_j \in \mathbb{R}^N$ is the j -th part vector of v partitioned by $(\mathbf{v}_j)_i = \mathbf{v}_{i+N \cdot j}$.

²Note that the script is very general and can be used for systems with any number of components. The user only needs to provide the system with the correct harmonics of interest, critical parameter values and the equilibrium point. To run the script, make sure you have installed SYMBOLIC MATH TOOLBOX as well.

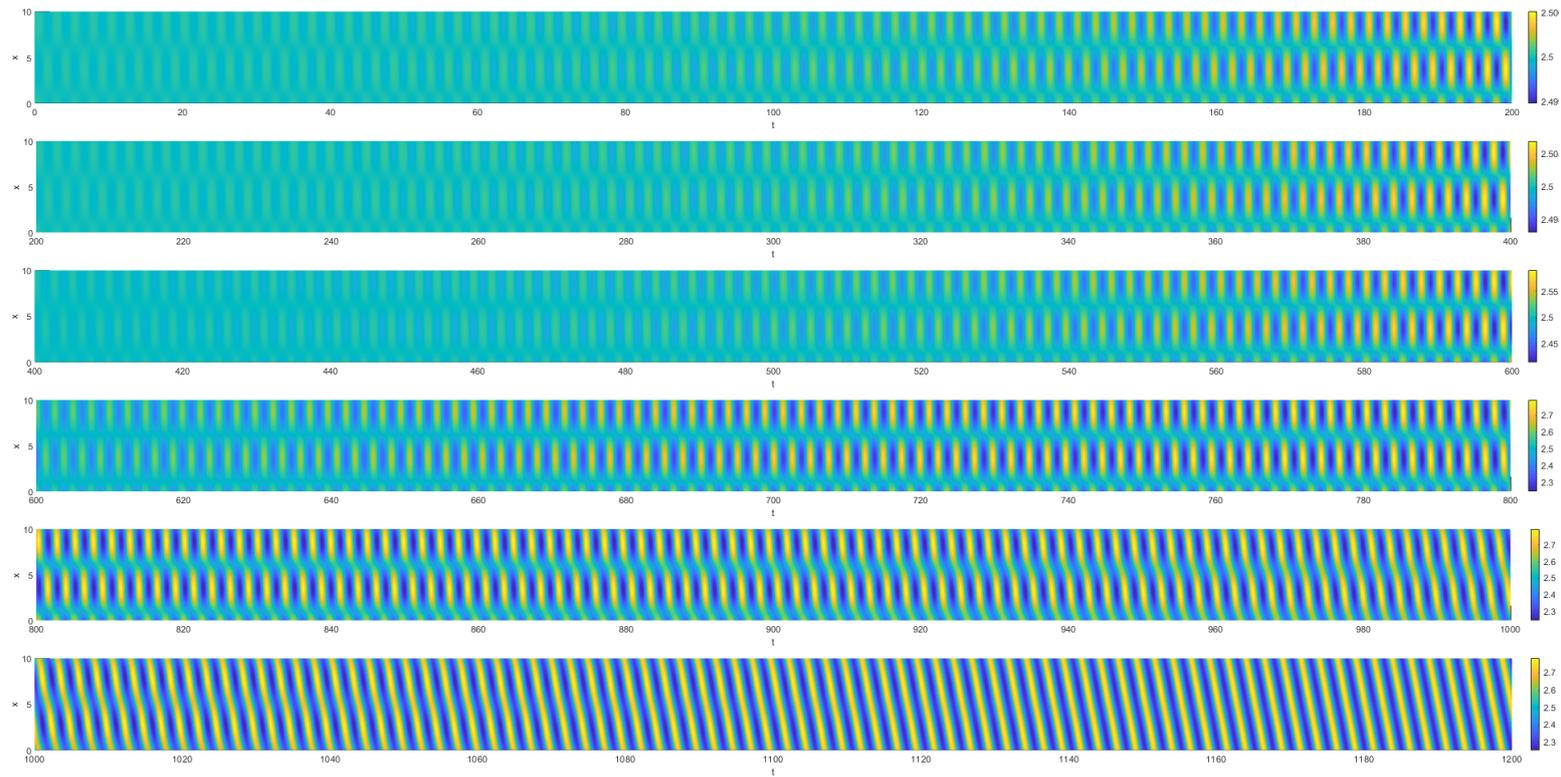


Figure 6: Numerical simulation of system (44) using ODE23S.

System (44) can be numerically integrated using ODE23S in MATLAB. Using the parameter value

$$b = 9.934138724791575 + 10^{-5},$$

that is slightly super-critical, we simulate the discretized reaction-diffusion system (43) with the initial distribution close to the uniform equilibrium

$$u(\theta) = 10^{-5} \cdot (\cos \theta, \cos(\theta + 0.3 \cdot 2\pi), \cos(\theta + 0.7 \cdot 2\pi)) + u_E,$$

and obtain a transient to a rotating wave presented in Figure 6.

It is interesting that the solution first approaches an unstable standing wave, but then evolves towards the stable rotating wave. This behavior agrees with the theoretical predictions, see Figure 2.

The results can be reproduced by running the SIMULATION SCRIPT in MATLAB. The parameters such as a , b and R can all be changed inside the script. However note that changing any of the parameters would change the system dynamics drastically, so the patterns obtained by the simulation may not coincide with the local prediction we have made in Section 3, since we consider only patterns near the bifurcation point. So we encourage to only make small parameter changes. Note that the accuracy of the simulation can be changed in the ODESET line. In this case we have chosen 10^{-9} as the error margin, which is produce accurate results but it is very slow. The simulation can be made faster by choosing a smaller error margin.

6 Discussion

We have successfully derived explicit formulas to compute the critical normal form coefficients for Hopf bifurcation in reaction-diffusion system on S^1 , and implemented them in MATLAB scripts to compute the critical bifurcation point and evaluate the coefficients numerically. We were also able to run simulations using MATLAB with a finite-difference method and confirm that the normal form predictions do correspond to the simulations. However we were only able to provide one fully fledged example, because of lack of time. Using the scripts we have provided, many more examples can be analyzed and in a fair amount of computation time. There are a few directions in which this research can be extended.

The explicit formulas (39) and (40) for the normal form coefficients can be generalized to a unit disk instead of just S^1 . Our original plan was to also include an example on a unit disk, however as it turns out, running the corresponding simulation with our method would not be feasible as the system size grows quadratically. This is because we need to consider a two dimensional grid as the discretization of space. However with faster numerical methods, this can be achieved.

One way to improve the simulation speed in S^1 drastically is by using truncated Fourier expansions. That is, we can approximate each component of the solution by

$$u(t, \theta) = a_0(t) + \sum_{k=1}^N [a_k(t) \cos(k\theta) + b_k(t) \sin(k\theta)],$$

and then transform the problem into an ordinary differential system by substituting this expression in (1) and integrating over $[0, 2\pi]$ its product with $\cos(j\theta)$ and $\sin(j\theta)$ to find $\dot{a}_j(t)$ and $\dot{b}_j(t)$. This can be done because the solutions are periodic in θ . This method can also be generalized to higher dimensional space, e.g. to a unit disk.

More complicated patterns can be analyzed and predicted using normal forms. For instance in [8] 6-dimensional normal forms were computed. These results can be used together with numerical simulations to analyse more sophisticated patterns. However the formulas we have provided are applicable under the assumption that we are dealing with the 4-dimensional center manifold. So special care needs to be taken when parametrizing higher-dimensional W_μ^c , and higher-order Taylor terms may be required to solve for the relevant normal form coefficients.

7 Appendix

Normal form coefficient script:

```
global a b R F u ue m

% Diffusion matrix
D0 = [
    0.1, 0, 0;
    0, 0.01, 0;
    0, 0, 2.0;
];

D0s = size(D0);
m = D0s(1);

% System parameters
a = 2.5;
R = 0.135;
b = 9.93413872;

syms('u', 1 : m);
syms('F', 1 : m);

% Reaction part
F(1) = a - (b + 1) * u(1) + u(1) * u(1) * u(2) + u(2) * u(3);
F(2) = -u(1) * u(1) * u(2) + b * u(1) - u(2) * u(3);
F(3) = R * (u(1) - u(2) * u(3));

% Uniform equilibrium
ue(1) = a;
ue(2) = (b - 1) / a;
ue(3) = a * a / (b - 1);

% Harmonic
k = 1;

% Get Jacobian matrix and evaluate at equilibrium
DF = jacobian(F(1 : m), u(1 : m));
A0 = vpa(subs(DF, u(1 : m), ue(1:m)));

M = -k^2 * D0 + A0;

% Gather eigenvalues and eigenvectors
[VV,Lambda] = eig(M);

[WW,Mu] = eig( transpose(M) );

% Only select the correct eigenvector for U and V
for i = 1 : m
    LambdaI = Lambda(i, i);
    if (imag(LambdaI) > 0)
        V = [VV(1 : m, i)];
    end
end

for i = 1 : m
    MuI = Mu(i, i);
    if (imag(MuI) < 0)
        W = [WW(1 : m, i)];
    end
end

% Make the eigenvector relatively orthogonal
S = dot(W, V);

U = (1 / conj(S)) * W;

V
U

% Define the complex conjugate of the eigenvectors
Vb = conj(V);
Ub = conj(U);

% Evaluate higher order Taylor terms at the eigenvectors and complex conjugate
BVV = Taylor2(V, V);
```

```

BVVb = Taylor2(V, Vb);

CVVV = Taylor3(V, V, V);
CVVVb = Taylor3(V, V, Vb);

% Solve non-singular systems
MatA = 2 * LambdaI * eye(m) - A0 + 4 * k^2 * D0;
MatB = 2 * LambdaI * eye(m) - A0;
MatC = -A0 + 4 * k^2 * D0;

% Evaluate final coefficients
GOV = [
    CVVVb + Taylor2(inv(MatA) * BVV, Vb) - 2 * Taylor2(inv(A0) * V, Vb)
];

HOV = [
    CVVV + Taylor2(inv(MatB) * BVV, Vb) + Taylor2(inv(MatC) * BVVb, V) - Taylor2(inv(A0) *
    BVVb, V)
];

G0 = 0.5 * dot(U, transpose(GOV));
H0 = dot(U, transpose(HOV));

% Print result
G0
H0

% Compute second order Taylor term
function B = Taylor2(v, w)
    global F u ue m

    for s = 1 : m
        Bs = 0;
        for i = 1 : m
            for j = 1 : m
                value = subs(diff(diff(F(s), u(i)), u(j)), u(1 : m), ue(1 : m));
                Bs = Bs + vpa(value) * v(1) * w(j);
            end
        end
        B(s, 1) = Bs;
    end
end

% Compute third order Taylor term
function C = Taylor3(v, w, z)
    global F u ue m

    for s = 1 : m
        Cs = 0;
        for i = 1 : m
            for j = 1 : m
                for k = 1 : m
                    value = subs(diff(diff(diff(F(s), u(i)), u(j)), u(k)), u(1 : m), ue(1 : m)
                    );
                    Cs = Cs + vpa(value) * v(1) * w(j) * z(k);
                end
            end
        end
        C(s, 1) = Cs;
    end
end

```

Simulation script:

```

global SpacePoints SpaceLength TimePoints a b R d1 d2 d3 odeopt;

% Setup simulation parameters
N = 1;
timeLength = 300;
TimePoints = 3000;

a = 2.5;
b = 9.934138724791575 + 1e-2;
R = 0.135;
d1 = 0.1;

```

```

d2 = 0.01;
d3 = 2.0;

% Setup initial function parameters
initialFreq1 = 1;
initialFreq2 = 1;
initialFreq3 = 1;
initialShift1 = 0 * 2 * pi;
initialShift2 = 0.3 * 2 * pi;
initialShift3 = 0.7 * 2 * pi;
initialEq1 = a;
initialEq2 = (b - 1) / a;
initialEq3 = a^2 / (b - 1);
initialAmplitude1 = 1e-2;
initialAmplitude2 = 1e-2;
initialAmplitude3 = 1e-2;

SpacePoints = 100;
SpaceLength = 2 * pi;

% Accuracy vs Time
% 4 = 160k+ 9 sec
% 5 = 1800k+ 73 sec
% 6 = 4100k+ 143 sec
% 8 = 19000k+ 377 sec
% 10 = 89903k+ 3848 sec

% Setup ode options
odeopt = odeset('RelTol', 1e-9, 'AbsTol', 1e-9);

% Setup initial condition
x0 = 0 : SpaceLength / (SpacePoints - 1) : SpaceLength;
y0s = zeros(3, SpacePoints);
y0s(1, :) = initialEq1 + initialAmplitude1 * cos((x0(:) + initialShift1) * initialFreq1);
y0s(2, :) = initialEq2 + initialAmplitude2 * cos((x0(:) + initialShift2) * initialFreq2);
y0s(3, :) = initialEq3 + initialAmplitude3 * cos((x0(:) + initialShift3) * initialFreq3);
y0 = [y0s(1, :), y0s(2, :), y0s(3, :)];

% Plot solution
for i = 0 : (N - 1)
    i

    % Plot every step of 2400 in 8 tiles
    cla reset
    tiledlayout(8, 1);
    for j = 0 : 7
        k = i * 8 + j;
        nexttile
        sim(x0, y0, k * timeLength + 0.00000001, (k + 1) * timeLength);
    end

    % Save the image as .fig file
    savefig('picture' + string(i) + '.fig')
end

t.Padding = 'none';
t.TileSpacing = 'compact';

function sim(x0, y0, BeginTime, EndTime)
    global TimePoints odeopt SpacePoints;

    % Solve system
    t = BeginTime : (EndTime - BeginTime) / (TimePoints - 1) : EndTime;
    [tt, yy] = ode23(@odefun, [0 t], y0, odeopt);

    % Post process solution
    yy0 = yy(2 : TimePoints + 1, 0 * SpacePoints + 1 : 1 * SpacePoints);
    yy1 = yy(2 : TimePoints + 1, 1 * SpacePoints + 1 : 2 * SpacePoints);
    yy2 = yy(2 : TimePoints + 1, 2 * SpacePoints + 1 : 3 * SpacePoints);

    xs = x0;
    ys = zeros(3, TimePoints, SpacePoints);
    ys(1, :, :) = yy0(:, :);
    ys(2, :, :) = yy1(:, :);
    ys(3, :, :) = yy2(:, :);

```

```

    % Plot simulation
    surfacePlot(ys, BeginTime, EndTime);
end

function surfacePlot(ys, BeginTime, EndTime)
    global SpaceLength SpacePoints TimePoints

    x = 0 : SpaceLength / (SpacePoints - 1) : SpaceLength;
    t = BeginTime : (EndTime - BeginTime) / (TimePoints - 1) : EndTime;

    % Manipulate solution
    yy(:, :, 1) = ys(1, :, :);
    y = squeeze(yy);

    surf(x, t, y);
    xlabel("x");
    ylabel("t");
    zlabel("u");
    shading interp
    grid off
    colorbar
    pbaspect([1 16 1]);
    view(90, -90);
    %axis off
end

% Derivative function
function dydt = odefun(t, y)
    a = 2.5;
    b = 9.934138724791575 + 1e-2;
    R = 0.135;
    d1 = 0.1;
    d2 = 0.01;
    d3 = 2.0;

    SpacePoints = 100;
    SpaceLength = 2 * pi;

    h = SpaceLength / SpacePoints;
    L = SpacePoints;

    % Setup result array
    dydt = zeros(3 * L, 1);
    diff = zeros(3 * L, 1);
    reac = zeros(3 * L, 1);

    % Compute finite difference at glued boundary
    diff(1 + 0 * L) = y(L + 0 * L) + y(2 + 0 * L) - 2 * y(1 + 0 * L);
    diff(1 + 1 * L) = y(L + 1 * L) + y(2 + 1 * L) - 2 * y(1 + 1 * L);
    diff(1 + 2 * L) = y(L + 2 * L) + y(2 + 2 * L) - 2 * y(1 + 2 * L);

    diff(L + 0 * L) = y((L - 1) + 0 * L) + y(1 + 0 * L) - 2 * y(L + 0 * L);
    diff(L + 1 * L) = y((L - 1) + 1 * L) + y(1 + 1 * L) - 2 * y(L + 1 * L);
    diff(L + 2 * L) = y((L - 1) + 2 * L) + y(1 + 2 * L) - 2 * y(L + 2 * L);

    % Compute finite difference inside the interval
    for i = 2 : L - 1
        diff(i + 0 * L) = y((i - 1) + 0 * L) + y((i + 1) + 0 * L) - 2 * y(i + 0 * L);
        diff(i + 1 * L) = y((i - 1) + 1 * L) + y((i + 1) + 1 * L) - 2 * y(i + 1 * L);
        diff(i + 2 * L) = y((i - 1) + 2 * L) + y((i + 1) + 2 * L) - 2 * y(i + 2 * L);
    end

    % Multiply finite difference by corresponding factors
    dh2 = 1 / h^2;
    diff(1 + 0 * L : 1 * L) = d1 * dh2 * diff(1 + 0 * L : 1 * L);
    diff(1 + 1 * L : 2 * L) = d2 * dh2 * diff(1 + 1 * L : 2 * L);
    diff(1 + 2 * L : 3 * L) = d3 * dh2 * diff(1 + 2 * L : 3 * L);

    % Compute reaction terms
    for i = 1 : L
        u1 = y(i + 0 * L);
        u2 = y(i + 1 * L);
        u3 = y(i + 2 * L);

        reac(i + 0 * L) = a - (b + 1) * u1 + u1^2 * u2 + u2 * u3;
        reac(i + 1 * L) = -u1^2 * u2 + b * u1 - u2 * u3;
    end
end

```

```

        reac(i + 2 * L) = R * (u1 - u2 * u3);
    end

    % Combine reaction and diffusion terms
    dydt(:) = diff(:) + reac(:);
end

```

Critical parameter finder:

```

matf = @mat

global a R d1 d2 d3 k

a = 2.5;
R = 0.135;
d1 = 0.1;
d2 = 0.01;
d3 = 2;
k = 1;
rangeBegin = 9.9;
rangeEnd = 10;

format long
fzero(matf, [rangeBegin rangeEnd])

function v = mat(b)
    global a R d1 d2 d3 k

    Q = zeros(3, 3);
    Q(1, 1) = b - 3;
    Q(2, 1) = -b + 2;
    Q(3, 1) = R;
    Q(1, 2) = (a * a * b) / (b - 1);
    Q(2, 2) = -(a * a * b) / (b - 1);
    Q(3, 2) = -(R * a * a) / (b - 1);
    Q(1, 3) = (b - 1) / a;
    Q(2, 3) = -(b - 1) / a;
    Q(3, 3) = -(R * (b - 1)) / a;

    Q(1, 1) = Q(1, 1) - k * k * d1;
    Q(2, 2) = Q(2, 2) - k * k * d2;
    Q(3, 3) = Q(3, 3) - k * k * d3;

    eigs = eig(Q);
    v = real(eigs(1));
end

```

Critical parameter chart script:

```

kn = 30;
b1 = 9.9;
b2 = 9.93413;
b3 = 10;

k = 0 : 2 / (kn - 1) : 2;
p = zeros(3, kn);

for i = 1 : kn
    p(1, i) = filterEig(eig(mat(k(i), b1)));
    p(2, i) = filterEig(eig(mat(k(i), b2)));
    p(3, i) = filterEig(eig(mat(k(i), b3)));
end

plot(k, p);
xlabel("k");
ylabel("p(k^2)");
xlim([0 2]);
ylim([-0.2 0.2]);
grid on

legend('beta = 9,9', 'beta = 9.93413', 'beta = 10')

function Q = mat(k, b)
    a = 2.5;
    R = 0.135;
    d1 = 0.1;
    d2 = 0.01;

```

```

d3 = 2;

Q = zeros(3, 3);
Q(1, 1) = b - 3;
Q(2, 1) = -b + 2;
Q(3, 1) = R;
Q(1, 2) = (a * a * b) / (b - 1);
Q(2, 2) = -(a * a * b) / (b - 1);
Q(3, 2) = -(R * a * a) / (b - 1);
Q(1, 3) = (b - 1) / a;
Q(2, 3) = -(b - 1) / a;
Q(3, 3) = -(R * (b - 1)) / a;

Q(1, 1) = Q(1, 1) - k * k * d1;
Q(2, 2) = Q(2, 2) - k * k * d2;
Q(3, 3) = Q(3, 3) - k * k * d3;
end

function eig = filterEig(eigs)
    eig = NaN;

    if imag(eigs(1)) ~= 0
        eig = real(eigs(1));
    elseif imag(eigs(2)) ~= 0
        eig = real(eigs(2));
    elseif imag(eigs(3)) ~= 0
        eig = real(eigs(3));
    end
end

```

References

- [1] G. Nicolis and I. Prigogine, *Self-organization in Nonequilibrium Systems: From Dissipative Structures to Order through Fluctuations*. Wiley-Interscience [John Wiley & Sons], New York-London-Sydney, 1977.
- [2] J. D. Murray, *Mathematical Biology II: Spatial Models and Biomedical Applications*. Springer, third ed., 2003.
- [3] Yu. A. Kuznetsov, *Elements of Applied Bifurcation Theory*, vol. 112 of *Applied Mathematical Sciences*. Springer-Verlag, New York, third ed., 2004.
- [4] G. B. Ermentrout, “Stable small-amplitude solutions in reaction-diffusion systems,” *Quart. Appl. Math.*, vol. 39, no. 1, pp. 61–86, 1981/82.
- [5] T. Erneux, “Stability of rotating chemical waves,” *J. Math. Biol.*, vol. 12, no. 2, pp. 199–214, 1981.
- [6] Yu. A. Kuznetsov, “Self-similar waves in “reaction-diffusion” systems with circular symmetry (method of center manifold).” Preprint. USSR Academy of Sciences, Research Computing Centre, Pushchino, 1985 [in Russian].
- [7] S. A. van Gils and J. Mallet-Paret, “Hopf bifurcation and symmetry: travelling and standing waves on the circle,” *Proc. Roy. Soc. Edinburgh Sect. A*, vol. 104, no. 3-4, pp. 279–307, 1986.
- [8] W. W. Farr and M. Golubitsky, “Rotating chemical waves in the Gray–Scott model,” *SIAM Journal on Applied Mathematics*, vol. 52, no. 1, pp. 181–221, 1992.
- [9] M. Ahamadi and J.-J. Gervais, “Symbolic-numerical methods for the computation of normal forms of PDEs,” *Journal of Computational and Applied Mathematics*, vol. 158, no. 2, pp. 443 – 472, 2003.
- [10] M. S. Polyakova, “On wave solutions in quasi-conservative three-component systems with diffusion,” in *Dynamics of Biological Populations*, pp. 15–21, Gorkii State University, 1983. [in Russian].
- [11] B. D. Hassard, N. D. Kazarinoff, and Y. H. Wan, *Theory and applications of Hopf bifurcation*, vol. 41 of *London Mathematical Society Lecture Note Series*. Cambridge University Press, Cambridge-New York, 1981.
- [12] Yu. A. Kuznetsov, “Andronov-Hopf bifurcation in four-dimensional systems with circular symmetry.” Preprint. USSR Academy of Sciences, Research Computing Centre, Pushchino, 1984 [in Russian].
- [13] E. E. Shnol’ and E. V. Nikolaev, “On bifurcations of symmetric equilibrium positions that correspond to double eigenvalues,” *Sb. Math.*, vol. 190, no. 9-10, pp. 1353–1376, 1999.