Cell polarisation in a bulk-surface model can be driven by both classic and non-classic Turing instability

Johannes Borgqvist[1†], Adam Malik[1], Carl Lundholm[1], Anders Logg[1], Philip Gerlee[1], Marija Cvijovic[1‡]

1 Department of Mathematical Sciences, Chalmers University of Technology and the University of Gothenburg, Gothenburg, Sweden.

Abstract

The GTPase Cdc42 is the master regulator of cell polarisation. During this process the active form of Cdc42 is accumulated at a particular site on the cell membrane called the *pole*. It is believed that the accumulation of the active Cdc42 resulting in a pole is driven by a combination of activation-inactivation reactions and diffusion. It has been proposed using mathematical modelling that this is the result of diffusion-driven instability, originally proposed by Alan Turing. In this study we develop a 3D bulk-surface model of the dynamics of Cdc42. We show that the model can undergo both classic and non-classic Turing instability. We thoroughly investigate the parameter space for which this occurs. Using simulations we show that the model can be used to simulate polarisation and to predict a number of relevant quantitative measures, including pole size and time to polarisation.

[†] johborgq@chalmers.se

[‡] Corresponding Author: Marija Cvijovic, Phone: +46 31 772 53 21, e-mail: marija.cvijovic@chalmers.se

Introduction

Cell division control protein 42 homolog, *Cdc42* is an enzyme of the class GTPases that regulates various signalling pathways involved in cell division and cell morphology [40, 10]. The enzyme is essential for dividing cells in organisms ranging from yeast to higher mammals which means that the protein is *evolutionary conserved*. In fact, Cdc42 is one of the most conserved GTPases where the Cdc42 in yeast is 80% identical to that in human cells [13, 10, 29, 51, 64, 12, 47, 49, 19]. In the late G_1-phase during the cell cycle, a sequence of events causes the accumulation of Cdc42 [11], which is the master regulator of cell division, at a specific location on the cell membrane. This location is called the *pole* which is the site where the new cell grows out and the latter process is called budding in the case of the yeast *Saccharomyces cerevisiae*. Moreover, in the cytosol Cdc42 is bound to GDP which corresponds to its inactive state while it is bound to GTP in the membrane corresponding to its active form. The conversion between these two states is catalysed by the two classes of enzymes called GEFs and GAPs. It is believed that it is the combination of these reactions of activation and inactivation along with diffusion that results in accumulation of active Cdc42.

Experimentally, the challenge with studying the activation of Cdc42 is that the concentration profile is not uniform in the cell. Thus, accounting for spatial inhomogeneities is crucial when data of the pathway is collected, however it is very hard indeed to measure two different diffusion rates simultaneously. Firstly, the slow diffusion rate of active Cdc42 on the cell membrane must be measured which is complicated in itself as the cell membrane can be viewed as the two-dimensional surface of the three-dimensional cell approximated by a ball and hence microscopy with a high resolution is required. Secondly, the fast diffusion rate inside the three dimensional cell and the slow diffusion on the two dimensional cell membrane must be measured simultaneously. On top of accounting for the spatial distribution of Cdc42, the activation and inactivation reaction rates should be measured as well. Usually, such rates are estimated from data of spatial averages of the concentration profiles over time which is perhaps not feasible to do in the case of the mentioned polarisation system as inhomogeneous distributions of proteins are crucial for the function of the system. On account of these complications, computational models have been developed to aid in understanding the activity of Cdc42.

Numerous mathematical models of the dynamics of Cdc42 have been developed [22, 61, 38, 68, 18, 21, 55, 14, 50, 28, 66, 16, 27]. Many of these models can be reduced to a classic *activator-inhibitor* system focusing on the spatial and temporal dynamics of active and inactive Cdc42. An important consideration in such models is whether the accumulation of active Cdc42 in a single location is the result of a Turing-type mechanism or not. Some mathematical models of polarisation have a single spatial dimension, describing either the chemical concentration along a diameter of the cell or the cell perimeter while considering the cytosol spatially uniform. Later however, a model on the single-cell scale was developed where Turing patterns formed on the cell membrane, in the presence of non-linear reactions involving another species diffusing in the cytosol [39]. It was demonstrated that with this new type of bulk-surface model, a distinct type of pattern formation mechanism was possible. In classic Turing-type systems, equal diffusion rates of the reacting species can never produce Turing patterns, however this is no longer a necessary restriction in the bulk-surface model. The authors argue that the necessary difference in transport can be achieved by choosing the various reaction-rates to be unequal and this is referred to as non-classic Turing patterns [56, 58]. Sufficient conditions for the emergence of both classic and non-classic Turing patterns in the context of bulk-surface models have been derived and demonstrated [58, 56]. Additionally, a previous 1-dimensional model of cell polarisation [50] has been extended to the bulk-surface context [15]. This model is a two-species system of active and inactive form of GTPases, and does not distinguish between the inactive form in the cytosol and the inactive form in the membrane. Following the assumption that the

cytosolic diffusion rate is much larger than the diffusion rate in the membrane, this results in different diffusion rates of the two species on the membrane. Although bulk-surface models of polarisation is not a novel concept, most previous work has been focused on the occurrence of pattern formation and the qualitative behaviour of the models. However, little has been done in order to investigate the parameter space and the regions that give rise to classic or non-classic Turing patterns. Consequently, the aim of this article is to develop a biologically motivated theoretical description of cell polarisation and use this to propose an underlying mechanism. This is achieved in four steps.

1. Derive a properly motivated reaction-diffusion model of Cdc42 activation based on biological knowledge of the system
2. Use mathematical analysis to investigate the two cases of classic and non-classic diffusion driven instability
3. Conduct an extensive analysis of the parameter space and how it influences polarisation
4. Validate the proposed mechanisms using three-dimensional spatiotemporal simulations of the mathematical model, and elucidate the precise conditions allowing for the formation of a single pole

Derivation of a reaction diffusion model of Cdc42 activation

To derive the reaction mechanism for the polarisation process mediated by Cdc42, we begin by defining the geometric domain of interest. The cell is modelled as the interior of a three dimensional sphere Ω with radius R where its surface Γ corresponds to the cell membrane (Fig 1B).

$$\begin{aligned} \Omega &:= \left\{ \mathbf{x} \in \mathbb{R}^3 \mid \|\mathbf{x}\|_2^2 < R^2 \right\} \\ \Gamma &:= \left\{ \mathbf{x} \in \mathbb{R}^3 \mid \|\mathbf{x}\|_2^2 = R^2 \right\} \end{aligned} \tag{1}$$

Here $\|\mathbf{x}\|_2$ is the Euclidian distance measure. As the cytosol Ω is comparatively large compared to the membrane it could be considered as a bulk which is subsequently linked to the two-dimensional inclusion Γ corresponding to the membrane. Consequently, we would refer to models which account for this geometric description (1) as *bulk-surface* models. In this way it is possible to expand the classic Turing framework [67] to account for reaction-diffusion models with two geometric domains which is both the most natural and more realistic than 1D simplifications in the context of Cdc42.

Using this notation, it is possible to derive a *bulk-surface Activator-Inhibitor* (AI)-system of cell polarisation mediated by Cdc42 (Fig 1A). The dynamics of the proposed AI-system is governed by five reactions: *Influx of inactive Cdc42 from the cytosol*, *Dissociation of inactive Cdc42 from the membrane*, *activation of inactive Cdc42*, *inactivation of active Cdc42* and *activation of Cdc42 through a positive feedback loop*. To this end, we introduce three functions

$$\begin{aligned} G &: \Omega \times [0, T] \to \mathbb{R} \\ A &: \Gamma \times [0, T] \to \mathbb{R} \\ I &: \Gamma \times [0, T] \to \mathbb{R} \end{aligned}$$

describing the concentrations of GTP-, GDP- and GDI-bound form respectively. Both G and I are referred to inactive form of Cdc42, whereas A is in active form. The functions A and I are restricted to the membrane Γ while the G is restricted to the cytosol Ω. The concentrations inside the cytosol are measured in mol/m^3, and concentration on the membrane in mol/m^2.

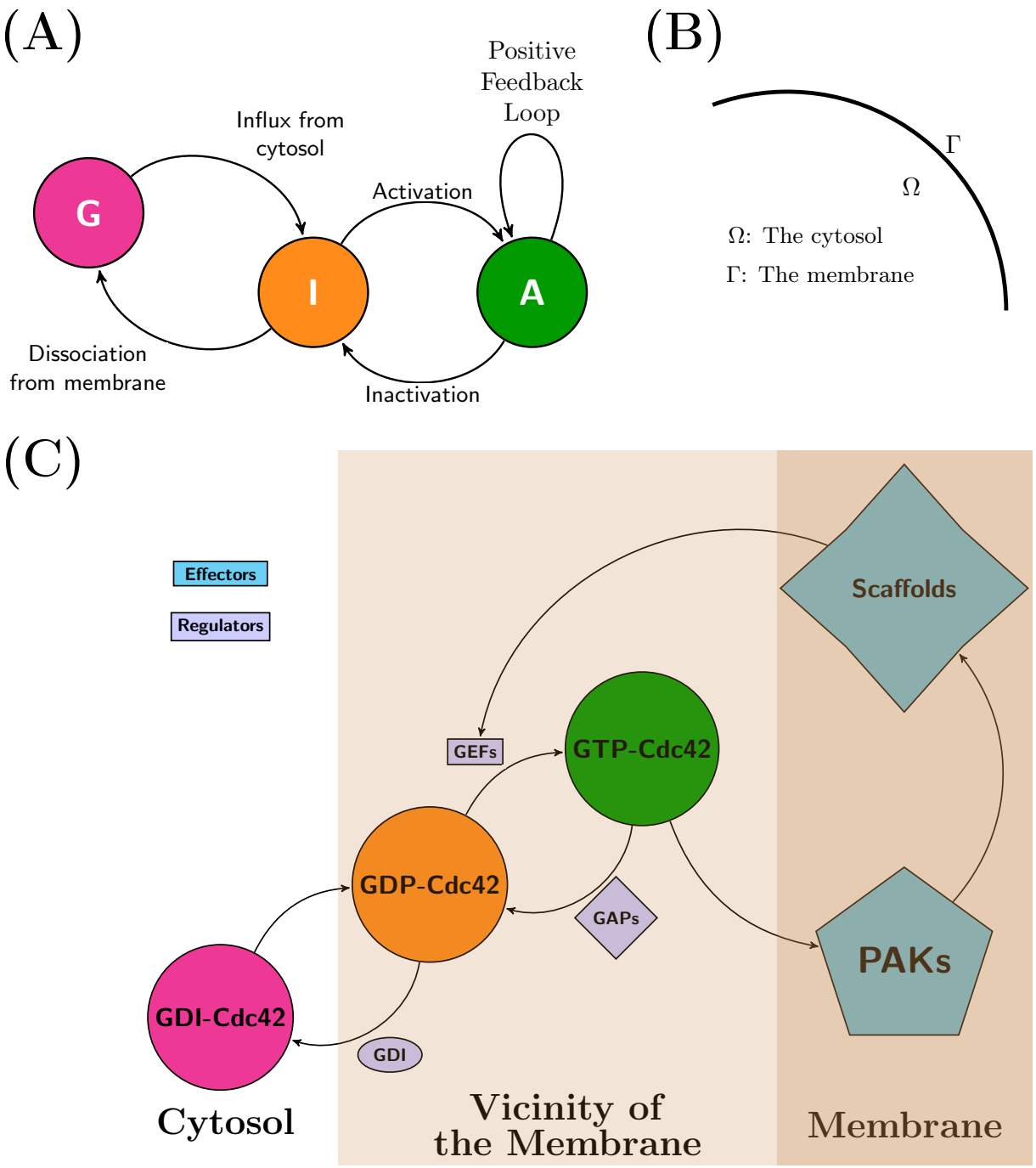

Figure 1: The model of Cdc42-activation. **(A)**: *The mechanism described as an expanded activator inhibitor system.* The GTPase Cdc42 is shuffled between its active state A (Green) corresponding to the GTP-bound form and its inactive state I (Orange) corresponding to the GDP-bound form. Also, the GDI-bound form G (Magenta) is restricted to the cytosol and is transported to the membrane which correspond to the influx of inactive form I. The reaction mechanism is determined by three classes of reactions: (1) Flux of inactive Cdc42 over the membrane, (2) Activation and Inactivation reactions restricted to the membrane determined by GEFs and GAPs respectively and (3) The positive feedback loop mediated by PAKs and Scaffolds also restricted to the membrane. **(B)**: *Geometric domain.* By letting the membrane thickness shrink to zero, the geometric description is simplified to one domain Ω corresponding to the cytosol and one boundary Γ corresponding to the membrane. **(C)** *Biological details of Cdc42 activation.* Cdc42 has an inhibited GDI-bound form (Magenta), an inactive GDP-bound form (orange) and an active GTP-bound form (green). The shuffling between these forms are determined by the three regulators namely GDI, GEFs and GAPs. The active form of Cdc42, unlike the inactive, can bind to various effector molecules such as PAKs and Scaffolds which can further bind to GEFs which enhances the activation reaction through a positive feedback loop. This sub-figure is adapted from Chiou et al. [13].

Influx of Inactive Cdc42 from the cytosol

The influx of inactive Cdc42 to the vicinity of the membrane, is determined by the concentration of GDI-bound Cdc42 in the cytosol (Fig 1C). The influx of the inactive GDP-bound form of Cdc42 would have a rate proportional to $k_1 \cdot G$ where the rate of the reaction is determined by the rate constant k_1 with units min^{-1}. We also assume that there exists a saturation level of active and inactive form of Cdc42 on the membrane, denoted by k_{max}. Accordingly, the reaction rate of the influx of cytosolic inactive GDI-bound Cdc42 to the membrane is given by:

$$\text{Influx rate} = k_1 \cdot G \cdot (k_{\text{max}} - (A + I)) \tag{2}$$

Dissociation of Inactive Cdc42 from the membrane

For the dissociation of GDP-bound Cdc42 from the membrane, we assume a first order reaction with rate constant k_{-1} which results in:

$$\text{Dissociation rate} = k_{-1} \cdot I$$

Activation of Inactive Cdc42

The activation of inactive Cdc42 which occurs on the membrane corresponds to the conversion of GDP-bound Cdc42 to the GTP-bound form. This reaction is proportional to the concentration of GEFs which we assume to be constant during the time scale on which polarisation occurs. Accordingly, the rate constant for this reaction k_2 is proportional to the concentration of GEFs, i.e. $k_2 \propto [\text{GEF}]$. Under this assumption, the reaction rate is modelled by a first order reaction as follows:

$$\text{Activation rate} = k_2 \cdot I$$

Inactivation of Active Cdc42

The inactivation of active Cdc42 corresponds to the conversion of GTP-bound Cdc42 to the GDP-bound form. This reaction is proportional to the concentration of GAPs which we assume to be constant during the time scale on which polarisation occurs. Accordingly, the rate constant for this reaction k_{-2} is proportional to the concentration of GAPs, i.e. $k_{-2} \propto [\text{GAP}]$. Under this assumption, the reaction rate is modelled by a first order reaction as follows:

$$\text{Inactivation rate} = k_{-2} \cdot A$$

Activation of Cdc42 through a positive feedback loop

The feedback loop consists of the binding of active Cdc42 to PAKs which forms a complex which can further bind to various scaffolds. Together, this sequence of events forms a structure which can bind more GEFs and thereby enhance the activation process. During the time scale that polarisation occurs, we assume that both the concentration of PAKs and GEFs are constant, and that the feedback loop that recruits GEF is nonlinear. Such a feedback mechanism has been proposed previously [21] and takes the form:

$$\text{Feedback activation rate} = k_3 \cdot A^2 \cdot I$$

where the reaction rate coefficient is $k_3 A$, and requires both A and I to occur. Combining all these terms allows us to formulate a reaction diffusion model for cell polarisation mediated by

Cdc42:

$$\frac{\partial G}{\partial t} = D_G \Delta G, \quad \mathbf{x} \in \Omega,\ t \in \mathbb{R}_+$$

$$\begin{aligned} -D_G \left[(\nabla G)^T \mathbf{n} \right] &= k_1 G \left(k_{\max} - (A + I) \right) - k_{-1} I, \quad \mathbf{x} \in \Gamma,\ t \in \mathbb{R}_+ \\ &= Q(A, I, G) \end{aligned}$$

$$\left.\begin{aligned} \frac{\partial A}{\partial t} &= k_2 I - k_{-2} A + k_3 A^2 I + D_A \Delta A \\ &= F(A, I) + D_A \Delta A \\ \\ \frac{\partial I}{\partial t} &= -F(A, I) + Q(A, I, G) + D_I \Delta I \\ &= G(A, I) + D_I \Delta I \end{aligned}\right\}, \quad \mathbf{x} \in \Gamma,\ t \in \mathbb{R}_+ \tag{3}$$

The dynamics in the cytosol Ω is entirely described by diffusion of the GDI-bound form G. The flux of inactive GDI-bound Cdc42 G from the cytosol Ω to the membrane Γ resulting in the influx of the membrane-bound inactive GDP-bound form of Cdc42 I is determined by the function Q. The function $Q(A, I, G)$ is implemented as a Robin boundary condition for the GDI-bound state G. The total mass of the system is conserved, independent of the choice of function F and Q, this can be seen by considering the temporal change of total mass, and using the partial differential equations

$$\begin{aligned} &\frac{\mathrm{d}}{\mathrm{d}t} \left(\int_\Omega G(\mathbf{x}, t)\ \mathrm{d}\mathbf{x} + \int_\Gamma A(\mathbf{s}, t) + I(\mathbf{s}, t)\ \mathrm{d}\mathbf{s} \right) \\ &= \int_\Omega D_G \Delta G\ \mathrm{d}\mathbf{x} + \int_\Gamma D_A \Delta A + Q(A, I, G) + D_I \Delta I\ \mathrm{d}\mathbf{s} \\ &= \int_\Gamma D_G (\nabla G)^T \mathbf{n} + D_A \Delta A + Q(A, I, G) + D_I \Delta I\ \partial \mathbf{s} \\ &= \int_\Gamma D_A \Delta A + D_I \Delta I\ \mathrm{d}\mathbf{s} \\ &= 0 \end{aligned}$$

which implies that the total amount of protein is constant.

We follow the standard procedure in the setting of dynamic models in biology, and non-dimensionalise the model in order to reduce the number of parameters. Our choice of non-dimensional parameters are similar to the ones in the classical Schnackenberg model. [52]. Firstly, we introduce the following dimensionless states

$$\begin{aligned} u &= A \cdot \sqrt{\frac{k_3}{k_{-2}}} \\ v &= I \cdot \sqrt{\frac{k_3}{k_{-2}}} \\ V &= G \cdot \frac{1}{R} \sqrt{\frac{k_3}{k_{-2}}} \end{aligned} \tag{4}$$

and the following dimensionless variables

$$\tau = \frac{D_A t}{R^2}$$
$$\mathbf{x} \leftarrow \frac{1}{R}\mathbf{x} \tag{5}$$

Note, that after the introduction of the spatial variable $\mathbf{x}$, the domain Ω is transformed to the *unit sphere* $\Omega := \left\{\mathbf{x} \in \mathbb{R}^3 : \|\mathbf{x}\|_2^2 < 1\right\}$ where the membrane is described by $\Gamma := \left\{\mathbf{x} \in \mathbb{R}^3 : \|\mathbf{x}\|_2^2 = 1\right\}$. The resulting dimensionless system by the introduction of these states and variables is the following (for details see S1.1 in the Supplementary material)

$$\begin{aligned}
\frac{\partial V}{\partial t} &= D\Delta V, \quad \mathbf{x} \in \Omega,\ t \in \mathbb{R}_+ \\
-D\left[(\nabla V)^T \mathbf{n}\right] &= \gamma\left\{c_1 V\left(c_{\max} - (u+v)\right) - c_{-1} v\right\}, \quad \mathbf{x} \in \Gamma,\ t \in \mathbb{R}_+ \\
&= \gamma q(u, v, V) \\
\left.\begin{aligned}
\frac{\partial u}{\partial \tau} &= \gamma\left(c_2 v - u + u^2 v\right) + \Delta u \\
&= \gamma f(u, v) + \Delta u \\
\frac{\partial v}{\partial \tau} &= \gamma\left(-f(u, v) + q(u, v, V)\right) + d\Delta v
\end{aligned}\right\}&, \quad \mathbf{x} \in \Gamma,\ t \in \mathbb{R}_+
\end{aligned} \tag{6}$$

In fact, as the cytosolic GDI-bound state of Cdc42 diffuses much faster than the membrane bound states [13], it is possible to reduce the number of equations in the system (6). More precisely, the assumption that $D_G \to \infty$ implies that the concentration of the cytosolic GDI-bound state is homogeneous and given by the functional

$$V(u+v) = V_0 - \frac{1}{|\Omega|}\int_\Gamma (u(s,t) + v(s,t))\,\mathrm{d}s$$

resulting in the two equations for u and v in (6), where the equation for V is replaced by the functional. The constant V_0 is the total average concentration of all three forms of Cdc42, and $V_0|\Omega|$ is the total amount of Cdc42 in the cell.

It is worth emphasising that the non-dimensionalisation procedure, (4) and (5), reduces the number of parameters from ten ($k_{\max}$, k_1, k_{-1}, k_2, k_{-2}, k_3, G_0, D_A, D_I and D_G) in the original system (3) to eight (γ, $c_{\max}$, c_1, c_{-1}, c_2, V_0, d and D) in the dimensionless system (6). This procedure reduces the number of parameters, and the resulting dimensionless parameters are also more meaningful compared to the original ones. For example, the parameter γ determines the relative strength of the *reaction* part of the model compared to the *diffusion* part which implies that this parameter determines which of these forces that dominate the dynamics of the system. Moreover, all of the dynamics corresponding to the activation, inactivation and the positive feed-back loop is captured in the dimensionless parameter c_2 which in the case with dimension is described by the three parameters k_2, k_{-2} and k_3. The dimensionless states, variables and parameters are summarised below (Tab 1). It is crucial to emphasise that our model builds on the framework presented by Rätz and Röger [57, 58] where the key difference is the reaction term f. Therefore, a motivation for this particular novelty is required.

Table 1: Dimensionless components of the model. The columns from left to right: the components (i.e. the states, variables and parameters), their definitions and a description of their meaning.

Component	**Definition**	**Description**
	The states	
u	$u = A \cdot \sqrt{\frac{k_3}{k_{-2}}}$	The dimensionless active form of Cdc42 (*membrane-bound*)
v	$v = I \cdot \sqrt{\frac{k_3}{k_{-2}}}$	The dimensionless inactive form of Cdc42 (*membrane-bound*)
V	$V = \frac{G}{R} \cdot \sqrt{\frac{k_3}{k_{-2}}}$	The dimensionless GDI-bound Cdc42 (*cytosolic*)
	The variables	
τ	$\tau = \frac{D_A t}{R^2}$	The dimensionless time (Eq (5))
$\mathbf{x} \in \Omega_R$	$\mathbf{x} \leftarrow \frac{1}{R}\mathbf{x}$	The dimensionless spatial variable (Eq (5))
	The parameters	
γ	$\gamma = \dfrac{R^2 k_{-2}}{D_A}$	The parameter has three interpretations [52, page 78]:(**1.**) γ is proportional to the area of the domain Ω (Eq (1)). (**2.**) γ is the relative strength of the two reaction terms determined by $f(u,v)$ and $g(u,v)$ (Eq (6)). (**3.**) An increase in γ corresponds to an equivalent decrease in the diffusion coefficient d.
c_1	$c_1 = \dfrac{k_1}{k_{-2}}\sqrt{\dfrac{k_{-2}}{k_3}}$	The relative influx of inactive Cdc42 from the cytosol
$c_{\max}$	$c_{\max} = k_{\max}\sqrt{\dfrac{k_3}{k_{-2}}}$	The maximum amount of membrane bound Cdc42, i.e. "$(u+v) \leq c_{\max}$"
c_{-1}	$c_{-1} = \frac{k_{-1}}{k_{-2}}$	The relative dissociation of inactive Cdc42 from the membrane
c_2	$c_2 = \frac{k_2}{k_{-2}}$	The relative activation rate of Cdc42
d	$d = \frac{D_I}{D_A}$	The relative diffusion rate of inactive Cdc42 (*diffusion in the membrane* Γ)
D	$D = \frac{D_G}{D_A}$	The relative diffusion rate of GDI-bound Cdc42 (*diffusion in the cytosol* Ω)
V_0	$V_0 = G_0 \cdot \sqrt{\frac{k_3}{k_{-2}}}$	The initial amount of dimensionless GDI-bound Cdc42

The difference regarding the function f describing the dynamics of the activation, inactivation and feedback loop of Cdc42 used in the previous model is the following. This function is described as follows (7) by Röger and Rätz [57, 58]

$$f(u,v) = a_1 + (a_3 - a_1)\frac{u}{a_2 + u} - a_4\frac{u}{a_5 + u} \tag{7}$$

which should be compared to

$$f(u,v) = c_2 v - u + u^2 v$$

the reaction-function implemented in this study. Based on this, we would argue that our model is advantageous both from a biological and mathematical perspective.

Biologically, the previous model (7) assumes *Michaelis–Menten* kinetics. Classically, this type of rate functions are implemented when a substrate S is converted to a product and where this conversion reaction is aided by an enzyme E which works as a catalyst. The particular rate function is proportional to $S/(K + S)$ for some constant $K > 0$, and this is expression is derived based on the assumption that the initial amount of substrate S_0 is substantially larger than the initial amount of enzymes E_0, i.e. $S_0 \gg E_0$. In the context of the Cdc42 model, this implies that the substrates would be the various states of Cdc42 while the enzymes would be the GEF's and GAP's. However, as Cdc42 itself is an enzyme it is more reasonable to assume that its intracellular concentration is in the same order of magnitude as that of the GAP's and GEF's.

Mathematically, our reaction term is advantageous for two reasons. Firstly, it is simpler in the sense that it has fewer parameters (one compared to six) and secondly its parameter is well-motivated. The simplicity of our reaction term is desirable provided that it can qualitatively model cell polarisation. Secondly, the results of the mathematical analysis of our model are more easily translated into biological knowledge compared to more complicated descriptions (7). This is due to the fact that various analytical results such as bounds on the involved kinetic parameters have biological implications as each parameter in our model has a concrete meaning which is not the case when the function involves a larger number of arbitrary parameters. For example, a large value of the parameter c_2 relative to the value of c_{-1} implies a high activity of the activation-inactivation module monitored by GEF's, GAP's and the positive feedback-loop relative to the activity of attachment of GDIs to the inactive form of Cdc42.

Consequently, it is of interest to see which values of the involved parameters that allow for symmetry breaking. As each parameter in the proposed model has a biological meaning, the classic and non-classic diffusion-driven instability constitute two candidate mechanisms behind cell polarisation mediated by Cdc42. It is this fact that constitutes the motivation behind the mathematical analysis of the model with the purpose of investigating firstly whether the model can undergo symmetry mechanism by means of any of these two mechanisms. If so, these two cases can be analysed, compared and investigated in terms of their biological implications and plausibility.

The bulk-surface description can model cell polarisation through both classic and non-classic Turing-patterns

Given the spatial model (6), we now investigate if spatial patterns can emerge through diffusion-driven instability. The underlying idea behind this type of pattern formation was originally formulated by Turing [67], and it entails a switch in stability in the sense of linear stability analysis. More precisely, the phenomena depends on the reaction terms, e.g. f and q, having the capacity to allow for the existence of a *stable* steady state (u^*, v^*) of the homogeneous, i.e. the system neglecting diffusion, ODE-system. Moreover, when diffusion is introduced the same

steady-state becomes an *unstable node* and it is this symmetry breaking that causes patterns to emerge. In other words, diffusion-driven instability with respect to a given steady state (u^*, v^*) implies a switch from a stable node in the homogeneous system to an unstable node in the inhomogeneous system, and we will refer to this as *classic* diffusion-driven instability. However, more recent work [57, 58] have shown that there are more ways by which symmetry breaking can be achieved. An example of such a case is a steady-state (u^*, v^*) transitioning from a *stable node* in the homogeneous system to a *saddle point* in the inhomogeneous system. The latter example will be referred to as *non-classic* diffusion-driven instability, and the exact formulation of the mathematical conditions for these two cases are presented in the Supplementary material (see section S1.2.1). It is important to emphasise that both cases rely on the existence of a steady-state with the capacity of switching stability, and it is thus crucial to find rate parameters ensuring this fundamental requirement.

To this end, we have mathematically proven that the model sustains such steady-states for certain parameters (Thm 1). More precisely, the theoretical bound (8) ensuring the existence of relevant steady states implies that the activation-inactivation parameter c_2 is constrained by the maximum concentration of membrane bound species, $c_{\max}$, and the average total concentration, V_0. Furthermore, we show that there is always one such steady-state (possibly more) which could potentially give rise to patterns and that it always lies in the interval $u^* \in (\sqrt{c_2}, \min\{c_{\max}, V_0/a\})$. These bounds indicate that using the rate parameter c_2, corresponding to the activation-inactivation reactions, it is possible to formulate a lower bound on the steady-state. Combining the maximum concentration of membrane-bound species $c_{\max}$ with the total average concentration V_0 one can formulate an upper bound. Note that these conditions (Thm 1) merely ensure the existence of steady-states, and to ensure the emergence of patterns the exact steady-state satisfying the mathematical conditions in the stability analysis (see section S1.2.1) must be found.

Theorem 1 (**Existence and characterisation of steady states**).
The system

$$\begin{aligned}\frac{\partial u}{\partial \tau} &= \gamma(c_2 v - u + u^2 v)\\ \frac{\partial v}{\partial \tau} &= \gamma\left[-(c_2 v - u + u^2 v) + c_1(V_0 - a(u+v))(c_{\max} - (u+v)) - c_{-1} v\right]\end{aligned}$$

has either 2, 4 or 6 positive steady-states within the first quadrant of the (u, v)-state space. Moreover, the system has at least one steady-state within the following region.

$$\mathcal{A} := \left\{ \begin{pmatrix} u \\ v \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \end{pmatrix} : u + v \leq \min(c_{\max}, m) \right\} \quad \textit{where } m = \frac{V_0}{a}$$

Lastly, a steady-state $(u^*, v^*) \in \mathcal{A}$ allowing for diffusion-driven instability satisfies the following bounds $u^\star \in \left(\sqrt{c_2}, \min\{c_{\max}, m\}\right)$. The following condition is necessary

$$\left(\sqrt{c_2} + \frac{1}{2\sqrt{c_2}}\right) \leq \min\{c_{\max}, m\} \tag{8}$$

in order to ensure the existence of such a steady-state.

Next we investigate the steady-states for different combinations of parameters, and determine whether they satisfy the classic and non-classic Turing conditions. For a large relative diffusion, d, the set of parameters enabling pattern formation is larger in the classic than the non-classic case. To visualise this fact, the (c_{-1}, c_2)-parameter space when the other parameters

are held constant is illustrated (Fig 2A) as well as the corresponding (c_1, c_2)-parameter space (Fig 2B). The first conclusion is that the set of parameters that allow for symmetry breaking increases proportionally with the relative diffusion, d, in both of the presented parameter planes. Conversely, when the the relative diffusion decreases the region of the parameter space allowing for diffusion-driven instability decreases as well, and in fact the non-classic case can be viewed as the classic case in the limit $d \to 1$. Note here, that the non-classic case is independent of the relative diffusion d and in the case where $d = 1$ the system can only form patterns through non-classic diffusion driven instability. However, as soon as $d > 1$ symmetry breaking can occur through both mechanisms and when the relative diffusion is large the classic region of the parameter space in larger than the non-classic counterpart.

With regards to the kinetic parameters, two major conclusions can be drawn. Firstly, the relative activation rate c_2 must be larger than the relative dissociation rate c_{-1} in order to allow classic diffusion driven instability (Fig 2A). For all tested values of the relative diffusion, the relative activation rate c_2 is approximately five to ten times larger than c_{-1}. Interestingly, in the non-classic case of diffusion driven instability the relative dissociation rate c_{-1} is closer to the relative activation rate c_2 implying that the two investigated cases have different biological implications. Secondly, within a range of relative activation rates c_2 the phenomena of diffusion driven instability is *independent* of the relative influx rate c_1 (Fig 2B). This conclusion holds true for larger values of the relative cytosolic flux than $c_1 = 10$ although the parameter space is only illustrated up to this value. Note that a general result from both parameter planes is that classic Turing instability occurs for higher relative activation rates c_2 compared to the non-classic case.

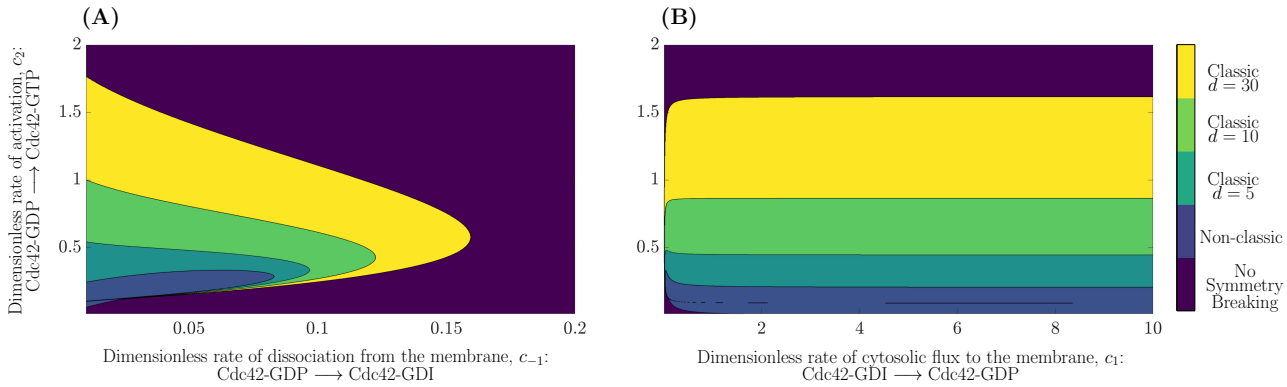

Figure 2: The parameter space for classic and non-classic diffusion-driven instability. The parameter space is divided into five regions indicated by the colour bar, namely *Classic Turing instability* with $d = 30$ (yellow), *Classic Turing instability* with $d = 10$ (light green), *Classic Turing instability* with $d = 5$ (green blue), *Non-classic Turing instability* (light blue) and *No Symmetry Breaking* (dark blue). Two intersections of the parameter space are visualised, namely **(A)** the $(c_{-1}, c_2)-$plane with a fixed value of $c_1 = 0.05$ and **(B)** the $(c_1, c_2)-$plane with a fixed value of $c_{-1} = 0.05$. The overall parameters in both cases are $V_0 = 6.0$ and $c_{\max} = 3.0$ (note that the non-classic case is independent of d).

Using these theoretical results, we are now interested in modelling cell-polarisation. Note that we have only derived conditions ensuring the formation of *a* pattern, but in fact only one specific pattern is of interest in the context of cell-polarisation. This pattern is the formation of a single *pole* corresponding to a single circular spot of active Cdc42 on the cell membrane, and to find the biologically realistic parameters sustaining this exact pattern we resort to numerical simulations.

Cell polarisation can be modelled through both mechanisms provided that the relative reaction strength is low

The experimental design for the simulations of cell polarisation is the following. Firstly, the time-evolution of one classic and one non-classic set of parameters is simulated. Secondly, the final patterns for different parameters satisfying either the classic or the non-classic case are presented. Thirdly, we choose different sets of parameters from each case and vary the relative diffusion d. Lastly, we investigate how the parameter γ corresponding to the relative strength of the reaction influences the final patterns.

Both cases of diffusion driven instability can model cell-polarisation (Fig 3). Although the time-evolution of the concentration profiles are slightly different, the final patterns are qualitatively very similar for the two cases. More precisely, the classic case (Fig 3A) forms a circular pole directly while the non-classic case (Fig 3B) initially forms more of an elongated pattern which gradually transitions into a pole. With the knowledge that the model is able to simulate cell-polarisation, it is of interest to investigate the effect of changing the kinetic parameters c_{-1} and c_2 on the pattern formation process.

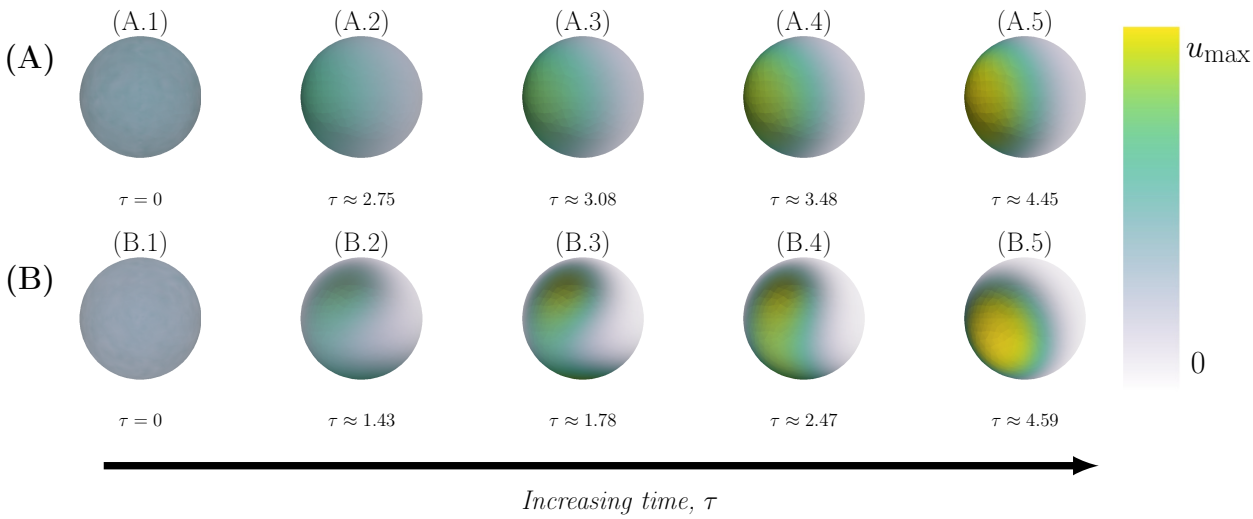

Figure 3: The time evolution of a pattern. The time-evolution of the concentration profiles for two sets of parameters corresponding to classic and non-classic respectively. In both cases the time to polarisation τ_{final} and the maximum concentration of active Cdc42 $u_{\max}$ are calculated. **(A)** *Classic*: The parameters are $(c_{-1}, c_2, \tau_{\text{final}}, u_{\max}) = (0.02, 0.45, 4.45, 3.80)$ where the following time points are displayed (A.1): $\tau = 0$ (A.2): $\tau \approx 2.75$, (A.3): $\tau \approx 3.08$, (A.4): $\tau \approx 3.48$ and (A.5) $\tau \approx 4.45$. **(B)** *Non-classic*: The parameters are $(c_{-1}, c_2, \tau_{\text{final}}, u_{\max}) = (0.01, 0.20, 4.58, 4.30)$ where the following time points are displayed (B.1):$\tau = 0$ (B.2): $\tau \approx 1.43$, (B.3): $\tau \approx 1.78$, (B.4): $\tau \approx 2.47$ and (B.5) $\tau \approx 4.59$. In both cases, the overall parameters are: $c_1 = 0.05$, $V_0 = 6.0$, $c_{\max} = 3.0$, $a = 3$, $d = 10$ and $\gamma = 25$.

The final patterns for different kinetic parameters are qualitatively similar but quantitatively different (Fig 4). When running simulations for different parameters within the $(c_{-1}, c_2)-$space (Fig 2A) while keeping the remaining parameters fixed, both the classic (Fig 4A) and the non-classic (Fig 4B) cases form biologically realistic patterns, namely a single pole. However, from a quantitative perspective the time it takes for forming this pole, τ_{final}, differs for different sets of kinetic parameters. For instance, in the classic case it varies from $\tau_{\text{final}} \approx 3.50$ (Fig 4A.3) to $\tau_{\text{final}} \approx 5.00$ (Fig 4A.4-A.5) while in the non-classic case it varies from $\tau_{\text{final}} \approx 3.74$ (Fig 4B.4) to $\tau_{\text{final}} \approx 27.62$ (Fig 4B.3). Similarly, the maximum concentration of active Cdc42 $u_{\max}$ when the pole is formed is different for different kinetic parameters. In the classic case, it varies from

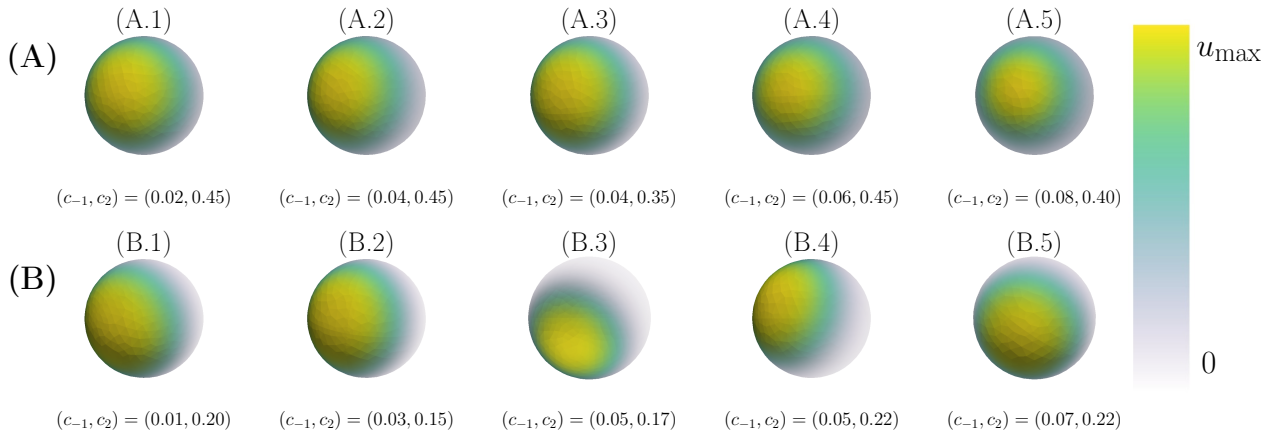

Figure 4: Final patterns for different kinetic parameters within the $(\mathbf{c_{-1}}, \mathbf{c_2})$-plane. The final patterns for various parameters in the (c_{-1}, c_2)-plane are displayed in two cases, namely classic and non-classic diffusion driven instability. In both cases, the final time when the pattern is formed τ_{final} and the maximum concentration of active Cdc42 $u_{\max}$ are calculated as functions of the kinetic rate parameters. **(A)** *Classic*: (A.1): $(c_{-1}, c_2, \tau_{\text{final}}, u_{\max}) = (0.02, 0.45, 4.45, 3.80)$, (A.2): $(c_{-1}, c_2, \tau_{\text{final}}, u_{\max}) = (0.04, 0.45, 4.00, 3.60)$, (A.3): $(c_{-1}, c_2, \tau_{\text{final}}, u_{\max}) = (0.04, 0.35, 3.50, 3.80)$, (A.4): $(c_{-1}, c_2, \tau_{\text{final}}, u_{\max}) = (0.06, 0.45, 5.00, 3.80)$ and (A.5) $(c_{-1}, c_2, \tau_{\text{final}}, u_{\max}) = (0.08, 0.40, 5.00, 3.20)$. **(B)** *Non-classic*: (B.1): $(c_{-1}, c_2, \tau_{\text{final}}, u_{\max}) = (0.01, 0.20, 4.58, 4.30)$, (B.2): $(c_{-1}, c_2, \tau_{\text{final}}, u_{\max}) = (0.03, 0.15, 4.01, 4.20)$, (B.3): $(c_{-1}, c_2, \tau_{\text{final}}, u_{\max}) = (0.05, 0.17, 27.62, 4.20)$, (B.4): $(c_{-1}, c_2, \tau_{\text{final}}, u_{\max}) = (0.05, 0.22, 3.74, 4.00)$ and (B.5) $(c_{-1}, c_2, \tau_{\text{final}}, u_{\max}) = (0.07, 0.22, 19.81, 4.00)$. In both cases, the overall parameters are: $c_1 = 0.05$, $V_0 = 6.0$, $c_{\max} = 3.0$, $a = 3$, $d = 10$ and $\gamma = 25$.

$u_{\max} = 3.20$ (Fig 4A.5) to $u_{\max} = 3.80$ (Fig 4A.1, A-3 and A.4) while in the non-classic case it varies from $u_{\max} = 4.00$ (Fig 4B.4 and B.5) to $u_{\max} = 4.30$ (Fig 4B.1). Next, we investigate the effect of increasing the relative diffusion on cell polarisation.

The major effect of increasing the relative diffusion is that the size of the pole decreases (Fig 5). In both the classic (Fig 5A) and the non-classic (Fig 5B) case, a low relative diffusion, e.g. $d = 10$ (Fig 5A.2 and B.2), results in a small pole while a large relative diffusion, e.g. $d = 50$ (Fig 5A.5 and B.5), yields a smaller pole. Also quantitatively, the maximum concentration $u_{\max}$ is higher and the final time τ_{final} is shorter in the latter case ($u_{\max} \approx 10$ and $\tau_{\text{final}} \approx 1.6 - 1.9$ for $d = 50$ (Fig 5A.5 and B.5)) compared to the former ($u_{\max} \approx 3.6 - 4.2$ and $\tau_{\text{final}} \approx 2.6 - 4.8$ for $d = 10$ (Fig 5A.2 and B.2)). Lastly, the effect of increasing the relative reaction strength γ is investigated.

A high value of the relative reaction strength increases the number of poles (Fig 6). We can see that in both the classic case (Fig 6A.1 and A.2) and the non-classic case (Fig 6B.1 and B.2) one pole is formed for values of $\gamma < 40$. Also for higher values such as $\gamma = 160$, the number of poles increases to four in the classic (Fig 6A.5) and five in the non-classic (Fig 6B.5) case. In addition, the reaction strength seems not to affect the maximum concentration $u_{\max}$ substantially. In the classic case, the maximum concentration varies between $u_{\max} \approx 2.7 - 3.8$ (Fig 6A) while the maximum concentration varies between $u_{\max} \approx 4.0 - 4.5$ (Fig 6B) in the non-classic case.

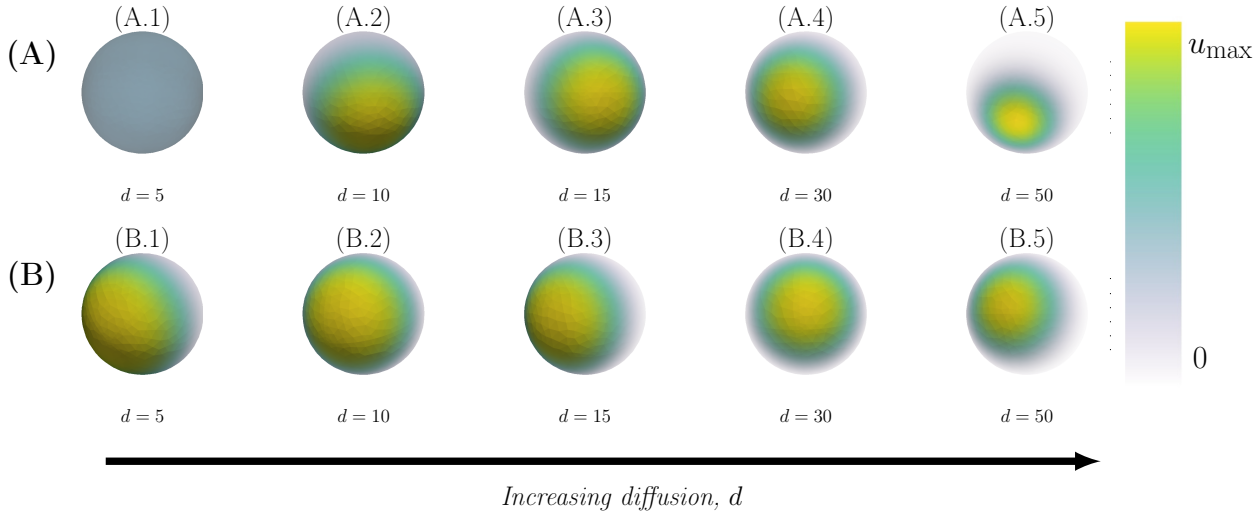

Figure 5: Final patterns for increasing relative diffusion. The final patterns for increasing relative diffusion d are displayed in two cases, namely classic and non-classic diffusion driven instability. In both cases, the final time when the pattern is formed τ_{final} and the maximum concentration of active Cdc42 $u_{\max}$ are calculated as functions of the kinetic rate parameters. **(A)** *Classic*: The overall parameters are $(c_1, c_{-1}, c_2) = (0.05, 0.04, 0.45)$ with specific parameters (A.1): no pattern is formed for $(d, \tau_{\text{final}}, u_{\max}) = (5, 15, 1.4)$, (A.2): $(d, \tau_{\text{final}}, u_{\max}) = (10, 4.84, 3.6)$, (A.3): $(d, \tau_{\text{final}}, u_{\max}) = (15, 3.75, 4.9)$, (A.4): $(d, \tau_{\text{final}}, u_{\max}) = (30, 2.76, 7.4)$ and (A.5) $(d, \tau_{\text{final}}, u_{\max}) = (50, 1.85, 9.9)$. **(B)** *Non-classic*: The overall parameters are $(c_1, c_{-1}, c_2) = (0.05, 0.03, 0.15)$ with specific parameters (B.1): $(d, \tau_{\text{final}}, u_{\max}) = (5, 4.80, 2.8)$, (B.2): $(d, \tau_{\text{final}}, u_{\max}) = (10, 2.60, 4.2)$, (B.3): $(d, \tau_{\text{final}}, u_{\max}) = (15, 2.11, 5.3)$, (B.4): $(d, \tau_{\text{final}}, u_{\max}) = (30, 1.62, 7.8)$ and (B.5) $(d, \tau_{\text{final}}, u_{\max}) = (50, 1.58, 10.0)$. In both cases, the overall parameters are: $V_0 = 6.0$, $c_{\max} = 3.0$, $a = 3$ and $\gamma = 25$.

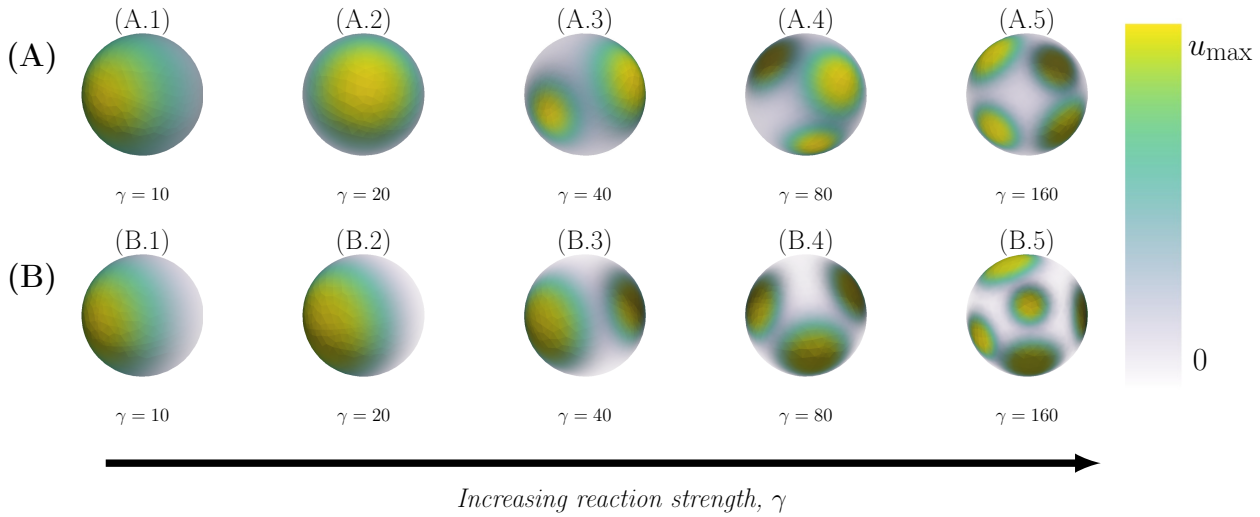

Figure 6: Final patterns for increasing relative reaction strength. The final patterns for increasing relative reaction strength γ are displayed in two cases, namely classic and non-classic diffusion driven instability. In both cases, the maximum concentration of active Cdc42 $u_{\max}$ is calculated as functions of the kinetic rate parameters. **(A)** *Classic*: The overall parameters are $(c_1, c_{-1}, c_2) = (0.05, 0.04, 0.45)$ with specific parameters (A.1): $(\gamma, u_{\max}) = (10, 2.7)$, (A.2): $(\gamma, u_{\max}) = (20, 3.6)$, (A.3): $(\gamma, u_{\max}) = (40, 3.8)$, (A.4): $(\gamma, u_{\max}) = (80, 3.8)$ and (A.5) $(\gamma, u_{\max}) = (160, 3.8)$. **(B)** *Non-classic*: The overall parameters are $(c_1, c_{-1}, c_2) = (0.05, 0.03, 0.15)$ with specific parameters (B.1): $(\gamma, u_{\max}) = (10, 4.0)$, (B.2): $(\gamma, u_{\max}) = (20, 4.2)$, (B.3): $(\gamma, u_{\max}) = (40, 4.4)$, (B.4): $(\gamma, u_{\max}) = (80, 4.4)$ and (B.5) $(\gamma, u_{\max}) = (160, 4.5)$. In both cases, the overall parameters are: $V_0 = 6.0$, $c_{\max} = 3.0$, $a = 3$ and $d = 10$.

Discussion

A biologically realistic model of Cdc42-mediated cell polarisation has been constructed, analysed and verified. Each part of the model is well-motivated by the biological literature, and includes a realistic spatial description of the cell. This is due to the fact that it includes both the cytosol and the membrane implying that it belongs to the class of so called *bulk-surface* models. The analysis of the model resulted in a mathematical theorem guaranteeing the existence of multiple steady-states and finding a necessary condition for diffusion-driven instability. Using a thorough numerical investigation of the parametric space, we have shown that the model can form patterns by means of two distinct mechanisms, namely classic and non-classic Turing instability. Also, these results showcased the connection between these two mechanisms, where the non-classic case can be viewed as the classic case in the limit when the relative diffusion of the membrane bound species is equal, i.e. $d \to 1$. Lastly, we validated these theoretical results by showing that both these mechanisms can sustain pattern formation. Using simulations, we propose that the biologically interesting pattern in the context of cell polarisation corresponding to a single pole is mainly driven by a low value of the reaction strength parameter γ, that the size of the poles are determined by the relative diffusion d and that the effect of changing the kinetic parameters is quantitative rather than qualitative.

The formulation of the totality of the bulk-surface model is novel. Although, the individual parts are not novel the combination of them and the detailed investigation of the analytical properties of the model are. The implemented non-dimensionalisation procedure was inspired by the model by Schnackenberg [62], and it results in meaningful parameters such as the relative reaction strength γ or the activation parameter c_2 corresponding to all reactions governing activation, inactivation and the feed-back loop. The novel activation-inactivation module de-

termining the membrane-bound reactions governed by the function f constitutes a minimal well-motivated reaction term where each parameter has a concrete biological meaning. The analytical conditions developed by Rätz and Röger [58] were used as a basis for a comprehensive analysis of the parametric space giving rise to diffusion-driven instability. Lastly, even though three-dimensional simulations of Cdc42-mediated cell polarisation are not novel [17, 65], the thorough investigation of the effect of moving in the parametric space on the patterns is.

Furthermore, the construction of the model has an increased biological relevance. Firstly, the choice of the geometrical description, i.e. to construct a bulk-surface model, that includes both the membrane and the cytosol in combination with adding the cytosolic GDI-bound form of Cdc42 to the model increases the level of realism. For example, many previous models of the "wave-pinning" type [27, 15, 50, 28] have only focused on the two membrane-bound species and assumed mass-conservation in the membrane ([15] is an exception here as it includes a cytosolic state but no extra reactions associated with it). We would claim that from a biological perspective this is not entirely plausible as there is a fast-moving cytosolic state of Cdc42 that contributes to the transfer and dissociation reactions at the membrane. Therefore, our choice of reaction term f in combination with the bulk-surface description renders the model more realistic. Secondly, the minimal reaction term $f(u, v)$ governing the activation-inactivation reactions is beneficial. On the one hand, it is biologically motivated (especially in relation to other implemented reaction terms (7)) where each term has a concrete meaning in terms of reaction rates. This results in the fact that the mathematical conditions (Thm 1) as well as the parametric descriptions (Fig 2) are easily interpretable in terms of biological meaning. For instance, the activation rate c_2 is higher than the dissociation rate c_{-1} from the membrane in the classic case compared to the non-classic case where these two rates are more similar (Fig 2). Biologically, the relative size between these parameters can be viewed as a kinetic "tug-of-war" between the two stable states of Cdc42, namely the cytosolic GDI-bound form and the membrane-bound GTP-bound form (see the magenta and green balls respectively, Fig 1C). The unstable GDP-bound form of Cdc42 (see the orange ball, Fig 1C) transitions to either the cytosolic GDI-bound form with the rate c_{-1} or the active GTP-bound form with the rate c_2. Also, the investigation of the parameter space reveals that for certain activation rates c_2, any value of the cytosolic flux to the membrane c_1 allows for diffusion driven instability in both cases (Fig 2B). Furthermore, a general conclusion drawn by studying the Turing parameter space is that in both the classic and non-classic case the activation rate c_2 is larger in the former compared to the latter. This reinforces the fact that these mechanisms are distinct both from a mathematical as well as a biological perspective.

The framework can model cell polarisation using both proposed mechanisms. This was demonstrated using numerical simulations, where we first showed that patterns can be formed both using the classic and the non-classic case (Fig 3). Again, it is clear that the time-scales and dynamics of the two cases differ underscoring that the mechanisms are distinct. Then, we investigated the sensitivity of the final pattern for the two cases (Fig 4) with respect to variations in the kinetic parameters in order to conclude that the effect is quantitative rather than qualitative. More precisely, a mere change of parameters in the (c_{-1}, c_2)-plane while keeping the other parameters fixed does not alter the qualitative behaviour as a single pole is formed, however quantitative measures such as the time it takes to form the pole τ_{final} or the maximum concentration of active Cdc42 $u_{\max}$ are different for different kinetic parameters. Partially, this contradicts previous studies [63] claiming that the classic Turing patterns are not robust. In addition, we showed that the size of the pole is determined by the relative diffusion d (Fig 5). This presents an opportunity to connect the simulations of the bulk-surface models to data as a measure, however crude, of the size of the pole (for example as a percentage of the entire surface of the cell) can be used to estimate the relative diffusion. This methodology for estimating the relative diffusion is consequential as it is currently not possible to differentiate

between the three states of Cdc42 by using fluorescent markers and it is thereby not possible to estimate the relative diffusion d of the two membrane bound species. Lastly, we showed that the key parameter determining the number of poles is the strength of the reaction term γ (Fig 6). More precisely, one pole is formed for values of $\gamma < 40$ while numerous poles are formed for larger values, and this shows that the two relative parameters, i.e. γ and d, are consequential in the context of cell polarisation. These parameters are well known in the analysis of mathematical models forming patterns through diffusion driven instability as they govern the number of allowed wave numbers [52]. The smaller the value of the parameters γ the fewer wave numbers contribute to the pattern formation and vice versa. This in agreement with our simulations showing that the number of poles that are formed is smaller for smaller values of γ in contrast to the number of poles for higher values and specifically this implies that one single pole can be formed for small values of these parameters. This indicates that cell polarisation, i.e. the formation of a single spot corresponding to a pole, can be achieved for both mechanisms as the formation of this particular pattern is dependent of the relative strengths of the reaction part γ and the diffusion part d. Thus, it is not qualitatively possible to rule out either the classic or the non-classic based on the patterns formed as both mechanisms can form a pole for low values of γ. However, it might be possible to quantitatively distinguish between the cases by studying the concentration profiles over time and comparing the time it takes for the patterns to be formed. Nevertheless, this poses experimental challenges as it is hard to connect high qualitative three-dimensional data based on microscopy with numerous images over time.

In summary, we have suggested two plausible mechanisms for Cdc42-mediated cell polarisation. Perhaps cells have evolved multiple ways of maintaining this evolutionary conserved function. Moving forward, it is possible to use this biologically realistic framework in order to validate it using experimental data. After this it is possible to make predictions regarding cell polarisation in different settings such as the effect of ageing on the function of Cdc42.

Methods

The representation of the parameter space (Fig 2) has been generated using Matlab [48]. For the simulations, combination of an adaptive solver based *Finite Differences* (*FD*) in time and the *Finite Element Method* (*FEM*) in space is implemented (see section S2.2 in the Supplementary Materials for details). As the numerical implementation solves a system of PDEs, a spatial discretisation is required in terms of a *mesh* over the domain Ω (Fig 1A). For this purpose, the mesh is generated using the three-dimensional finite element mesh generator Gmsh[20]. For computational speed, we have implemented a non-uniform mesh with higher node-density close to the membrane and lower node-density in the cytosol as the former region requires more computational accuracy during the cell-polarisation simulations than the latter. For the FD- and FEM-implementations, the computing platform FEniCS [30, 32, 33, 36, 37, 34, 41, 53, 35, 42, 60, 5, 7, 45, 54, 23, 26, 59, 6, 3, 43, 46, 44, 31, 8, 4, 2, 24, 25] has been used. The visualisations (Fig 3, 4 and 6) have been constructed using the software ParaView [1, 9]. For all the simulations, we have used a cytosolic diffusion coefficient of $D = 10,000$ and the spatially inhomogeneous initial conditions are perturbed around the steady states of the three states. A detailed description of the implementations can be found in the Supplementary Material (see section S2).

Acknowledgements

MC and JB are supported by the Swedish Agency for Strategic Research (grant nr. IB13-0022). PG was supported by the Swedish Foundation for Strategic Research (grant no. AM13-0046) and AM by Vetenskapsrådet (grant no. 2014-6095)

References

[1] James Ahrens, Berk Geveci, and Charles Law. Paraview: An end-user tool for large data visualization. *The visualization handbook*, 717, 2005.

[2] Martin S. Alnæs. *UFL: a Finite Element Form Language*, chapter 17. Springer, 2012.

[3] Martin S. Alnæs, Jan Blechta, Johan Hake, August Johansson, Benjamin Kehlet, Anders Logg, Chris Richardson, Johannes Ring, Marie E. Rognes, and Garth N. Wells. The fenics project version 1.5. *Archive of Numerical Software*, 3(100), 2015.

[4] Martin S. Alnæs, Anders Logg, and Kent-Andre Mardal. *UFC: a Finite Element Code Generation Interface*, chapter 16. Springer, 2012.

[5] Martin S. Alnæs, Anders Logg, Kent-Andre Mardal, Ola Skavhaug, and Hans Petter Langtangen. Unified framework for finite element assembly. *International Journal of Computational Science and Engineering*, 4(4):231–244, 2009.

[6] Martin S. Alnæs, Anders Logg, Kristian B. Ølgaard, Marie E. Rognes, and Garth N. Wells. Unified form language: A domain-specific language for weak formulations of partial differential equations. *ACM Transactions on Mathematical Software*, 40(2), 2014.

[7] Martin S. Alnæs and Kent-Andre Mardal. On the efficiency of symbolic computations combined with code generation for finite element methods. *ACM Transactions on Mathematical Software*, 37(1), 2010.

[8] Martin S. Alnæs and Kent-Andre Mardal. *SyFi and SFC: Symbolic Finite Elements and Form Compilation*, chapter 15. Springer, 2012.

[9] Utkarsh Ayachit. *The paraview guide: a parallel visualization application*. Kitware, Inc., 2015.

[10] Erfei Bi and Hay-Oak Park. Cell polarization and cytokinesis in budding yeast. *Genetics*, 191(2):347–387, 2012.

[11] Ralph A Bradshaw and Edward A Dennis. *Handbook of cell signaling*. Academic press, 2009.

[12] WEINING Chen, HH Lim, and L Lim. The cdc42 homologue from caenorhabditis elegans. complementation of yeast mutation. *Journal of Biological Chemistry*, 268(18):13280–13285, 1993.

[13] Jian-geng Chiou, Mohan K Balasubramanian, and Daniel J Lew. Cell polarity in yeast. *Annual review of cell and developmental biology*, 33:77–101, 2017.

[14] Jian-Geng Chiou, Samuel A Ramirez, Timothy C Elston, Thomas P Witelski, David G Schaeffer, and Daniel J Lew. Principles that govern competition or co-existence in rho-gtpase driven polarization. *PLoS computational biology*, 14(4):e1006095, 2018.

[15] Davide Cusseddu, Leah Edelstein-Keshet, John A Mackenzie, Stéphanie Portet, and Anotida Madzvamuse. A coupled bulk-surface model for cell polarisation. *Journal of theoretical biology*, 2018.

[16] Brian Drawert, Stefan Hellander, Michael Trogdon, Tau-Mu Yi, and Linda Petzold. A framework for discrete stochastic simulation on 3d moving boundary domains. *The Journal of Chemical Physics*, 145(18):184113, 2016.

[17] Brian Drawert, Stefan Hellander, Michael Trogdon, Tau-Mu Yi, and Linda Petzold. A framework for discrete stochastic simulation on 3d moving boundary domains. *The Journal of chemical physics*, 145(18):184113, 2016.

[18] Jayme M Dyer, Natasha S Savage, Meng Jin, Trevin R Zyla, Timothy C Elston, and Daniel J Lew. Tracking shallow chemical gradients by actin-driven wandering of the polarization site. *Current biology*, 23(1):32–41, 2013.

[19] Suzanne Eaton, Petri Auvinen, Liqun Luo, Yuh Nung Jan, and Kai Simons. Cdc42 and rac1 control different actin-dependent processes in the drosophila wing disc epithelium. *The Journal of cell biology*, 131(1):151–164, 1995.

[20] Christophe Geuzaine and Jean-François Remacle. Gmsh: A 3-d finite element mesh generator with built-in pre-and post-processing facilities. *International journal for numerical methods in engineering*, 79(11):1309–1331, 2009.

[21] Andrew B Goryachev and Marcin Leda. Many roads to symmetry breaking: molecular mechanisms and theoretical models of yeast cell polarity. *Molecular biology of the cell*, 28(3):370–380, 2017.

[22] Andrew B Goryachev and Alexandra V Pokhilko. Dynamics of cdc42 network embodies a turing-type mechanism of yeast cell polarity. *FEBS letters*, 582(10):1437–1443, 2008.

[23] Johan Hoffman, Johan Jansson, Rodrigo V. de Abreu, Cem Degirmenci, Niclas Jansson, Kaspar Müller, Murtazo Nazarov, and Jeanette H. Spühler. Unicorn: Parallel adaptive finite element simulation of turbulent flow and fluid-structure interaction for deforming domains and complex geometry. *Computer and Fluids*, in press, 2012.

[24] Johan Hoffman, Johan Jansson, Cem Degirmenci, Niclas Jansson, and Murtazo Nazarov. *Unicorn: a Unified Continuum Mechanics Solver*, chapter 18. Springer, 2012.

[25] Johan Hoffman, Johan Jansson, Niclas Jansson, C. Johnson, and Rodrigo V. de Abreu. *Turbulent Flow and Fluid–structure Interaction*, chapter 28. Springer, 2012.

[26] Niclas Jansson, Johan Jansson, and Johan Hoffman. Framework for massively parallel adaptive finite element computational fluid dynamics on tetrahedral meshes. *SIAM Journal on Scientific Computing*, 34(1):C24–C41, 2012.

[27] Alexandra Jilkine and Leah Edelstein-Keshet. A comparison of mathematical models for polarization of single eukaryotic cells in response to guided cues. *PLoS computational biology*, 7(4):e1001121, 2011.

[28] Alexandra Jilkine, Athanasius FM Marée, and Leah Edelstein-Keshet. Mathematical model for spatial segregation of the rho-family gtpases based on inhibitory crosstalk. *Bulletin of mathematical biology*, 69(6):1943–1978, 2007.

[29] Douglas I Johnson and John R Pringle. Molecular characterization of cdc42, a saccharomyces cerevisiae gene involved in the development of cell polarity. *The Journal of cell biology*, 111(1):143–152, 1990.

[30] Robert C. Kirby. Algorithm 839: Fiat, a new paradigm for computing finite element basis functions. *ACM Transactions on Mathematical Software*, 30(4):502–516, 2004.

[31] Robert C. Kirby. *FIAT: Numerical Construction of Finite Element Basis Functions,*, chapter 13. Springer, 2012.

[32] Robert C. Kirby, Matthew G. Knepley, Anders Logg, and L. Ridgway Scott. Optimizing the evaluation of finite element matrices. *SIAM Journal on Scientific Computing*, 27(3):741–758, 2005.

[33] Robert C. Kirby and Anders Logg. A compiler for variational forms. *ACM Transactions on Mathematical Software*, 32(3), 2006.

[34] Robert C. Kirby and Anders Logg. Efficient compilation of a class of variational forms. *ACM Transactions on Mathematical Software*, 33(3), 2007.

[35] Robert C. Kirby and Anders Logg. Benchmarking domain-specific compiler optimizations for variational forms. *ACM Transactions on Mathematical Software*, 35(2):1–18, 2008.

[36] Robert C. Kirby, Anders Logg, L. Ridgway Scott, and Andy R. Terrel. Topological optimization of the evaluation of finite element matrices. *SIAM Journal on Scientific Computing*, 28(1):224–240, 2006.

[37] Robert C. Kirby and L. Ridgway Scott. Geometric optimization of the evaluation of finite element matrices. *SIAM Journal on Scientific Computing*, 29(2):827–841, 2007.

[38] Anita T Layton, Natasha S Savage, Audrey S Howell, Susheela Y Carroll, David G Drubin, and Daniel J Lew. Modeling vesicle traffic reveals unexpected consequences for cdc42p-mediated polarity establishment. *Current Biology*, 21(3):184–194, 2011.

[39] Herbert Levine and Wouter-Jan Rappel. Membrane-bound turing patterns. *Physical Review E*, 72(6):061912, 2005.

[40] Daniel J Lew and Steven I Reed. Morphogenesis in the yeast cell cycle: regulation by cdc28 and cyclins. *The Journal of cell biology*, 120(6):1305–1320, 1993.

[41] Anders Logg. Automating the finite element method. *Archives of Computational Methods in Engineering*, 14(2):93–138, 2007.

[42] Anders Logg. Efficient representation of computational meshes. *International Journal of Computational Science and Engineering*, 4(4):283–295, 2009.

[43] Anders Logg, Kent-Andre Mardal, Garth N. Wells, et al. *Automated Solution of Differential Equations by the Finite Element Method*. Springer, 2012.

[44] Anders Logg, Kristian B. Ølgaard, Marie E. Rognes, and Garth N. Wells. *FFC: the FEniCS Form Compiler*, chapter 11. Springer, 2012.

[45] Anders Logg and Garth N. Wells. Dolfin: Automated finite element computing. *ACM Transactions on Mathematical Software*, 37(2), 2010.

[46] Anders Logg, Garth N. Wells, and Johan Hake. *DOLFIN: a C++/Python Finite Element Library*, chapter 10. Springer, 2012.

[47] Liqun Luo, Y Joyce Liao, Lily Yeh Jan, and Yuh Nung Jan. Distinct morphogenetic functions of similar small gtpases: Drosophila drac1 is involved in axonal outgrowth and myoblast fusion. *Genes & development*, 8(15):1787–1802, 1994.

[48] MATLAB. *version 9.4.0.813654 (R2018a)*. The MathWorks Inc., Natick, Massachusetts, 2018.

[49] Peter J Miller and Douglas I Johnson. Cdc42p gtpase is involved in controlling polarized cell growth in schizosaccharomyces pombe. *Molecular and cellular biology*, 14(2):1075–1083, 1994.

[50] Yoichiro Mori, Alexandra Jilkine, and Leah Edelstein-Keshet. Wave-pinning and cell polarity from a bistable reaction-diffusion system. *Biophysical journal*, 94(9):3684–3697, 2008.

[51] S Munemitsu, MA Innis, R Clark, F McCormick, A Ullrich, and P Polakis. Molecular cloning and expression of a g25k cdna, the human homolog of the yeast cell cycle gene cdc42. *Molecular and cellular biology*, 10(11):5977–5982, 1990.

[52] J. D. Murray. *Mathematical Biology II: Spatial Models and Biomedical Applications*, volume 18 of *Interdisciplinary Applied Mathematics*. Springer New York, 2003.

[53] Kristian B. Ølgaard, Anders Logg, and Garth N. Wells. Automated code generation for discontinuous galerkin methods. *SIAM Journal on Scientific Computing*, 31(2):849–864, 2008.

[54] Kristian B. Ølgaard and Garth N. Wells. Optimisations for quadrature representations of finite element tensors through automated code generation. *ACM Transactions on Mathematical Software*, 37, 2010.

[55] Mikiya Otsuji, Shuji Ishihara, Kozo Kaibuchi, Atsushi Mochizuki, Shinya Kuroda, et al. A mass conserved reaction–diffusion system captures properties of cell polarity. *PLoS computational biology*, 3(6):e108, 2007.

[56] Andreas Rätz. Turing-type instabilities in bulk–surface reaction–diffusion systems. *Journal of Computational and Applied Mathematics*, 289:142–152, 2015.

[57] Andreas Rätz and Matthias Röger. Turing instabilities in a mathematical model for signaling networks. *Journal of mathematical biology*, 65(6-7):1215–1244, 2012.

[58] Andreas Rätz and Matthias Röger. Symmetry breaking in a bulk–surface reaction–diffusion model for signalling networks. *Nonlinearity*, 27(8):1805, 2014.

[59] Marie E. Rognes, David A. Ham, Colin J. Cotter, and Andrew T. T. McRae. Automating the solution of pdes on the sphere and other manifolds in fenics 1.2. *Geoscientific Model Development*, 6:2099–2119, 2013.

[60] Marie E. Rognes, Robert C. Kirby, and Anders Logg. Efficient assembly of h(div) and h(curl) conforming finite elements. *SIAM Journal on Scientific Computing*, 31(6):4130–4151, 2009.

[61] Natasha S Savage, Anita T Layton, and Daniel J Lew. Mechanistic mathematical model of polarity in yeast. *Molecular biology of the cell*, 23(10):1998–2013, 2012.

[62] J Schnakenberg. Simple chemical reaction systems with limit cycle behaviour. *Journal of theoretical biology*, 81(3):389–400, 1979.

[63] Natalie S Scholes, David Schnoerr, Mark Isalan, and Michael PH Stumpf. A comprehensive network atlas reveals that turing patterns are common but not robust. *Cell systems*, 9(3):243–257, 2019.

[64] Katsuhiro Shinjo, John G Koland, Matthew J Hart, Vikram Narasimhan, Douglas I Johnson, Tony Evans, and Richard A Cerione. Molecular cloning of the gene for the human placental gtp-binding protein gp (g25k): identification of this gtp-binding protein as the human homolog of the yeast cell-division-cycle protein cdc42. *Proceedings of the National Academy of Sciences*, 87(24):9853–9857, 1990.

[65] Michael Trogdon, Brian Drawert, Carlos Gomez, Samhita P Banavar, Tau-Mu Yi, Otger Campàs, and Linda R Petzold. The effect of cell geometry on polarization in budding yeast. *PLoS computational biology*, 14(6):e1006241, 2018.

[66] Michael Trogdon, Brian Drawert, Carlos Gomez, Samhita P. Banavar, Tau-Mu Yi, Otger Campàs, and Linda R. Petzold. The effect of cell geometry on polarization in budding yeast. *PLOS Computational Biology*, 14(6):1–22, 06 2018.

[67] A. M. Turing. The chemical basis of morphogenesis. *Bulletin of Mathematical Biology*, 52(1):153–197, Jan 1952.

[68] John J Tyson. Modeling the cell division cycle: cdc2 and cyclin interactions. *Proceedings of the National Academy of Sciences*, 88(16):7328–7332, 1991.

Supplementary material for the article

Classic and non-classic Turing instability can drive cell polarisation in a bulk-surface model

Johannes Borgqvist[1], Adam Malik[1], Carl Lundholm[1], Anders Logg[1], Philip Gerlee[1], Marija Cvijovic[1]

1 Department of Mathematical Sciences, Chalmers University of Technology and the University of Gothenburg, Gothenburg, Sweden.

Date: January 29, 2020

*Corresponding Author: Marija Cvijovic, Phone: +46 31 772 53 21, e-mail: `marija.cvijovic@chalmers.se`

Contents

S1 Analytical results **1**
S1.1 Non-dimensionalisation . . . 1
S1.2 Diffusion driven instability . . . 5
S1.2.1 Steady states . . . 7
S1.2.2 Steady states enabling diffusion-driven instability . . . 8

S2 Numerical implementations **11**
S2.1 Mapping of the parameter space . . . 11
S2.2 Solving the RD-system numerically . . . 12
S2.2.1 Setting up the problem . . . 12
S2.2.2 FEM-discretisation in space . . . 14
S2.2.3 FD-discretisation in time . . . 16

List of Figures

S7 Steady states . . . 9
S8 The spatial Domain . . . 13
S9 Validation of FEM-FD implementation . . . 19

List of Tables

S1 Analytical results

The analytical results consists of two parts. Firstly, the details of the non-dimensionalisation is presented and secondly the proof of Theorem 1 on page 10 in the article is presented.

S1.1 Non-dimensionalisation of the model

The aim to render the following model

$$
\begin{aligned}
\frac{\mathrm{d}A}{\mathrm{d}t} &= \underbrace{k_2 I - k_{-2}A + k_3 A^2 I}_{=F(A,I)} + D_A \Delta A \\
&= F(A,I) + D_A \Delta A \\
\frac{\mathrm{d}I}{\mathrm{d}t} &= \underbrace{-F(A,I) + Q(A,I,G)}_{=G(A,I)} + D_I \Delta I \\
&= G(A,I) + D_I \Delta I \\
\frac{\mathrm{d}G}{\mathrm{d}t} &= D_G \Delta G \\
-D\left((\nabla G)^T \cdot \mathbf{n}\right) &= k_1 V (k_{\max} - (A+I)) - k_1 I \\
&= Q(A,I,G)
\end{aligned}
\tag{S9}
$$

dimensionless using the proposed scalings of the states ((4) on page 6) and the variables ((5) on page 7). We start with the left hand side and the time derivatives in order to obtain,

$$
\begin{aligned}
\mathrm{d}A &= \sqrt{\frac{k_{-2}}{k_3}}\mathrm{d}u \\
\mathrm{d}I &= \sqrt{\frac{k_{-2}}{k_3}}\mathrm{d}v \\
\mathrm{d}G &= \frac{1}{R}\sqrt{\frac{k_{-2}}{k_3}}\mathrm{d}V
\end{aligned}
$$

where the time variable is the following.

$$
\mathrm{d}t = \frac{R^2}{D_A}\mathrm{d}\tau
$$

These expressions result in the following left hand side in the PDEs (S9)

$$
\begin{aligned}
\frac{\mathrm{d}A}{\mathrm{d}t} &= \left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\frac{\mathrm{d}u}{\mathrm{d}\tau} \\
\frac{\mathrm{d}I}{\mathrm{d}t} &= \left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\frac{\mathrm{d}v}{\mathrm{d}\tau} \\
\frac{\mathrm{d}G}{\mathrm{d}t} &= \left(\frac{D_A}{R^3}\sqrt{\frac{k_{-2}}{k_3}}\right)\frac{\mathrm{d}V}{\mathrm{d}\tau}
\end{aligned}
$$

and thus the aim is to factor out "$\left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)$" from the remaining terms in (S9): $F(A,I)$, $G(A,I)$, and the diffusion terms. For the diffusion terms, in the one dimensional case[S1], i.e. $x \in [0, L]$, it follows that the following holds.

$$
\begin{aligned}
D_A \Delta A &= D_A \frac{\partial^2}{\partial x^2} A \\
&\quad \{\text{Use the non-dimensional assignment for the spatial variable } x \text{ ((5) on page 7)}\} \\
&= \frac{D_A}{L^2}\frac{\partial^2}{\partial x^2} A = \frac{D_A}{L^2}\Delta A \\
&\quad \{\text{Substitute the non-dimensional state } u \text{ in place of } A \text{ ((4) on page 6)}\} \\
&= \left(\frac{D_A}{L^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\Delta u
\end{aligned}
$$

Analogously, in the spherical case with the spherical Laplace operator ((**??**) on page **??**) the factor L^2 in the denominator in the one-dimensional case is replaced by the square of the radius R^2 which is summarised as follows.

$$
D_A \Delta A = \left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\Delta u
$$

Similarly, for I

$$
D_I \Delta I = \left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\frac{D_I}{D_A}\Delta v
$$

and for G the following holds.

$$
D_G \Delta G = \left(\frac{D_A}{R^3}\sqrt{\frac{k_{-2}}{k_3}}\right)\frac{D_G}{D_A}\Delta V
$$

For the reaction term $F(A,I)$, we have

$$
\begin{aligned}
F(A,I) &= k_2 I - k_{-2} A + k_3 A^2 I \\
&= \left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\left(\frac{R^2}{D_A}\sqrt{\frac{k_3}{k_{-2}}}\right)\left(k_2 I - k_{-2} A + k_3 A^2 I\right) \\
&= \left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\left(\frac{R^2}{D_A}\right)\left(k_2 \underbrace{\left[\sqrt{\frac{k_3}{k_{-2}}} I\right]}_{=v} - k_{-2}\underbrace{\left[\sqrt{\frac{k_3}{k_{-2}}} A\right]}_{=u} + k_3 \underbrace{A^2}_{=\left(\frac{k_{-2}}{k_3}\right)u^2} \underbrace{\left[\sqrt{\frac{k_3}{k_{-2}}} I\right]}_{=v}\right) \\
&= \left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\left(\frac{R^2}{D_A}\right)\left(k_2 v - k_{-2} u + k_3\left(\frac{k_{-2}}{k_3}\right)u^2 v\right) \\
&= \left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\left(\frac{R^2 k_{-2}}{D_A}\right)\left(\frac{k_2}{k_{-2}} v - u + u^2 v\right)
\end{aligned}
$$

[S1] The spherical case is analogous.

which is summarised as follows.

$$F(A,I) = \left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\left(\frac{R^2 k_{-2}}{D_A}\right)\left(\frac{k_2}{k_{-2}}v - u + u^2 v\right) = \left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\left(\frac{R^2 k_{-2}}{D_A}\right) f(u,v)$$

Lastly, we would like to factor out "$\left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\left(\frac{R^2 k_{-2}}{D_A}\right)$" from the above expression of $F(A,I)$ from $G(A,I)$. However, as $G(A,I) = -F(A,I) + Q(A,I,G)$ it suffices to look at the the transfer function "$Q(A,I,G)$". In a similar fashion, we obtain

$$\begin{aligned}
Q(A,I,G) &= k_1 G(k_{\max} - (A+I)) - k_{-1} I \\
&= \left[\left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\left(\frac{R^2 k_{-2}}{D_A}\right)\right]\left(\frac{1}{k_{-2}}\sqrt{\frac{k_3}{k_{-2}}}\right)\{k_1 G(k_{\max} - (A+I)) - k_{-1} I\} \\
&= \left[\left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\left(\frac{R^2 k_{-2}}{D_A}\right)\right]\left\{\frac{k_1}{k_{-2}}\left[\underbrace{\sqrt{\frac{k_3}{k_{-2}}}G}_{=V}\right](k_{\max} - (A+I)) - \frac{k_{-1}}{k_{-2}}\left[\underbrace{\sqrt{\frac{k_3}{k_{-2}}}I}_{=v}\right]\right\} \\
&= \left[\left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\left(\frac{R^2 k_{-2}}{D_A}\right)\right]\left\{\frac{k_1}{k_{-2}}V(k_{\max} - (A+I)) - \frac{k_{-1}}{k_{-2}}v\right\} \\
&= \left[\left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\left(\frac{R^2 k_{-2}}{D_A}\right)\right]\left\{\frac{k_1}{k_{-2}}\sqrt{\frac{k_{-2}}{k_3}}V\left(k_{\max}\sqrt{\frac{k_3}{k_{-2}}} - \left[\underbrace{\sqrt{\frac{k_3}{k_{-2}}}(A+I)}_{=(u+v)}\right]\right) - \frac{k_{-1}}{k_{-2}}v\right\} \\
&= \left[\left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\left(\frac{R^2 k_{-2}}{D_A}\right)\right]\left\{\frac{k_1}{k_{-2}}\sqrt{\frac{k_{-2}}{k_3}}V\left(k_{\max}\sqrt{\frac{k_3}{k_{-2}}} - (u+v)\right) - \frac{k_{-1}}{k_{-2}}v\right\}
\end{aligned}$$

which yields the following result.

$$\begin{aligned}
Q(A,I,G) &= \left[\left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\left(\frac{R^2 k_{-2}}{D_A}\right)\right]\left\{\frac{k_1}{k_{-2}}\sqrt{\frac{k_{-2}}{k_3}}V\left(k_{\max}\sqrt{\frac{k_3}{k_{-2}}} - (u+v)\right) - \frac{k_{-1}}{k_{-2}}v\right\} \\
&= \left[\left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\left(\frac{R^2 k_{-2}}{D_A}\right)\right] q(u,v,V)
\end{aligned}$$

Now, summarising all these terms yields the following

$$
\begin{aligned}
\left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\frac{\mathrm{d}u}{\mathrm{d}\tau} &= \left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\left\{\left(\frac{R^2k_{-2}}{D_A}\right)\left(\frac{k_2}{k_{-2}}v-u+u^2v\right)+\Delta u\right\}\\
\left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\frac{\mathrm{d}v}{\mathrm{d}\tau} &= \left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\left\{\left(\frac{R^2k_{-2}}{D_A}\right)\left[-\left(\frac{k_2}{k_{-2}}v-u+u^2v\right)+\right.\right.\\
&\qquad\left.\left.\left\{\frac{k_1}{k_{-2}}\sqrt{\frac{k_{-2}}{k_3}}V\left(k_{\max}\sqrt{\frac{k_3}{k_{-2}}}-(u+v)\right)-\frac{k_{-1}}{k_{-2}}v\right\}\right]+\frac{D_I}{D_A}\Delta v\right\}\\
\left(\frac{D_A}{R^3}\sqrt{\frac{k_{-2}}{k_3}}\right)\frac{\mathrm{d}V}{\mathrm{d}\tau} &= \left(\frac{D_A}{R^3}\sqrt{\frac{k_{-2}}{k_3}}\right)\frac{D_G}{D_A}\Delta G\\
\left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\left\{-\frac{D_G}{D_A}\left((\nabla V)^T\cdot\mathbf{n}\right)\right\} &= \left[\left(\frac{D_A}{R^2}\sqrt{\frac{k_{-2}}{k_3}}\right)\left(\frac{R^2k_{-2}}{D_A}\right)\right]\\
&\qquad\left\{\frac{k_1}{k_{-2}}\sqrt{\frac{k_{-2}}{k_3}}V\left(k_{\max}\sqrt{\frac{k_3}{k_{-2}}}-(u+v)\right)-\frac{k_{-1}}{k_{-2}}v\right\}
\end{aligned}
$$

and cancelling the common factor in both sides results in the following equation.

$$
\begin{aligned}
\frac{\mathrm{d}u}{\mathrm{d}\tau} &= \left(\frac{R^2k_{-2}}{D_A}\right)\left(\frac{k_2}{k_{-2}}v-u+u^2v\right)+\Delta u\\
\frac{\mathrm{d}v}{\mathrm{d}\tau} &= \left(\frac{R^2k_{-2}}{D_A}\right)\left[-\left(\frac{k_2}{k_{-2}}v-u+u^2v\right)+\left\{\frac{k_1}{k_{-2}}\sqrt{\frac{k_{-2}}{k_3}}V\left(k_{\max}\sqrt{\frac{k_3}{k_{-2}}}-(u+v)\right)-\frac{k_{-1}}{k_{-2}}v\right\}\right]+\frac{D_I}{D_A}\Delta v\\
\frac{\mathrm{d}V}{\mathrm{d}\tau} &= \frac{D_G}{D_A}\Delta V\\
\left\{-\frac{D_G}{D_A}\left((\nabla V)^T\cdot\mathbf{n}\right)\right\} &= \left(\frac{R^2k_{-2}}{D_A}\right)\left\{\frac{k_1}{k_{-2}}\sqrt{\frac{k_{-2}}{k_3}}V\left(k_{\max}\sqrt{\frac{k_3}{k_{-2}}}-(u+v)\right)-\frac{k_{-1}}{k_{-2}}v\right\}
\end{aligned}
$$

By introducing the following parameters

$$
c_1=\frac{k_1}{k_{-2}}\sqrt{\frac{k_{-2}}{k_3}},\quad c_{\max}=k_{\max}\sqrt{\frac{k_3}{k_{-2}}},\quad c_{-1}=\frac{k_{-1}}{k_{-2}},\quad c_2=\frac{k_2}{k_{-2}},\quad \gamma=\frac{R^2k_{-2}}{D_A},\quad D=\frac{D_G}{D_A},\quad \text{and}\quad d=\frac{D_I}{D_A}
$$

results in the following equations

$$
\begin{aligned}
\frac{\mathrm{d}u}{\mathrm{d}\tau} &= \gamma\left(c_2v-u+u^2v\right)+\Delta u\\
\frac{\mathrm{d}v}{\mathrm{d}\tau} &= \gamma\left(-\left(c_2v-u+u^2v\right)+\left(c_1V(c_{\max}-(u+v))-c_{-1}v\right)\right)+d\Delta v\\
\frac{\mathrm{d}V}{\mathrm{d}\tau} &= D\Delta V\\
-D\left((\nabla V)^T\cdot\mathbf{n}\right) &= \gamma\left[c_1V(c_{\max}-(u+v))-c_{-1}v\right]
\end{aligned}
$$

which is the desired result.

□

S1.2 Diffusion driven instability in the limit $D \to \infty$

We reduce the complexity of the system ((6) on page 7) by considering the limit $D \to \infty$. This implies that the cytosolic concentration of GDI-bound Cdc42 is approximated as homogeneous motivated by the fact that the internal diffusion D is much faster compared to the diffusion in the membrane. In this case, the mass conservation property ((**??**) on page **??**) is described by the *non-local functional* $V[u+v]$ below

$$V[u+v] = V_0 - \frac{1}{|\Omega|}\int_\Gamma (u+v)\,\mathrm{d}s \tag{S10}$$

and the RD-system ((6) on page 7) gets reduced to the following two-state system.

$$\begin{aligned} \partial_t u &= \Delta u + \gamma f(u,v) \\ \partial_t v &= d\Delta v + \gamma(-f(u,v) + q(u,v,V[u+v])) \end{aligned} \tag{S11}$$

This system ((S10) and (S11)) is first described in Röger and Rätz [S35] and we will use the same notation as the one introduced in this work here. Now, the stability analysis concerns both the homogeneous system without spatial effects and the inhomogeneous system accounting for spatial effects. In the former case, the non-local functional is transformed to the non-local function $V_1(u+v)$ defined as follows.

$$\begin{gathered} V_1(u+v) = V_0 - a(u+v) \\ \text{where} \\ a = \frac{|\Gamma|}{|\Omega|} \end{gathered} \tag{S12}$$

Note that in our case the domain Ω is the unit ball after the non-dimensionalisation and Γ is the unit sphere. Consequently, the parameter a (S12) has the value $a = \dfrac{4\pi}{((4\pi)/(3))} = 3$.

Now, let $(u^\star, v^\star, V^\star)$ denote a steady state of the system (S11) and let f_u, f_v, q_u, q_v, q_V and $V_1' = -a$ denote the partial derivatives evaluated at this steady state. Then, the stability of the homogeneous system is given by the following conditions.

$$0 > f_u - f_v + q_v + q_V V_1^{'} \tag{S13}$$

$$0 < f_u(q_v + q_V V_1^{'}) - f_v(q_u + q_V V_1^{'}) \tag{S14}$$

As in Proposition 3.1 by Röger and Rätz [S35], it is possible to obtain symmetry breaking in two ways provided that the above conditions ((S13) and (S14)) are satisfied. The first way is by the classic diffusion-driven instability proposed by Alan Turing [S38] corresponding to what we will refer to as the classic Turing conditions.

$$f_u q_v - f_v q_u \geq 0 \tag{S15}$$

$$d f_u - f_v + q_v > 0 \tag{S16}$$

$$Q := (d f_u - f_v + q_v)^2 - 4d(f_u q_v - f_v q_u) \geq 0 \tag{S17}$$

Here, the eigenvalues $\lambda_\pm$ of the membrane bound Laplace operator Δ_Γ often referred to as the *wave numbers* are given by the following equation.

$$\lambda_{\pm} = \frac{1}{2d}(df_u - f_v + q_v \pm \sqrt{Q}) \tag{S18}$$

In fact, two conclusions can be drawn from the classic conditions ((S13) and (S16)). Firstly, the diffusion ratio must satisfy $d > 1$ implying that the state v diffuses faster than u. Secondly, if we denote the sign of the diagonal elements of the Jacobian matrix must be opposite. In linear stability analysis, the Jacobian matrix for a general system with reaction terms determined by f and g consists of the partial derivatives of f and g with respect to the states u and v where these derivatives are evaluated at the steady-state (u^*, v^*) of interest. As a consequence of the latter, it follows that the elements of the Jacobian matrix must have the following signs [S31, Fig 2.6 Page 88].

$$J = \begin{pmatrix} f_u & f_v \\ g_u & g_v \end{pmatrix} \quad \text{with signs} \quad \propto \begin{cases} \begin{pmatrix} + & - \\ + & - \end{pmatrix} \\ \\ \begin{pmatrix} + & + \\ - & - \end{pmatrix} \end{cases} \tag{S19}$$

The other way by which symmetry breaking can be achieved which we will call *non-classic* Turing conditions are formulated as follows.

$$f_u q_v - f_v q_u < 0 \tag{S20}$$

$$\lambda_+ > 0 \tag{S21}$$

In the article, we state sufficient conditions for these two types of symmetry breaking by means of four theoretical results. The first lemma states conditions that are sufficient for a steady state that can give rise to Turing instability. The second lemma states conditions sufficient for a negative trace of the homogeneous system (S13). Given these two lemmas, the first theorem states sufficient conditions for Turing instability while the second theorem states sufficient conditions for saddle point instability. It is these conditions that we check numerically by firstly calculating a steady-state and then evaluating the conditions at the steady-state at hand. The details of how this is done will be presented in the next part of this document (S2.1). Before this is done, we prove Theorem 1 on page 10 stating the existence of steady-states and their characterisations.

S1.2.1 Existence of a steady state to the homogeneous system

Recall that a steady state is a solution $(u, v) = (u^*, v^*)$ to the following two equations.

$$f(u, v) = c_2 v - u + u^2 v = 0$$
$$q(u, v) = c_1(V_0 - a(u + v))(c_{\max} - (u + v)) - c_{-1} v = 0$$

The first equation is satisfied for all v of the form

$$v = \frac{u}{c_2 + u^2}$$

which can be inserted into the second equation, resulting in

$$c_1 \left(V_0 - a \left(u + \frac{u}{c_2 + u^2} \right) \right) \left(c_{\max} - \left(u + \frac{u}{c_2 + u^2} \right) \right) - c_{-1} \frac{u}{c_2 + u^2} = 0$$
$$\Longrightarrow c_1 V_0 (c_2 + u^2) - c_1 a \left(u(c_2 + u^2) + u \right) \left(c_{\max} - \left(u + \frac{u}{c_2 + u^2} \right) \right) - c_{-1} u = 0$$
$$\Longleftrightarrow$$
$$\boxed{p(u) = c_1 V_0 (c_2 + u^2)^2 - c_1 a \left(u(c_2 + u^2) + u \right) \left[c_{\max}(c_2 + u^2) - u(c_2 + u^2) + u \right] - c_{-1} u (c_2 + u^2)} = 0 \tag{S22}$$

which is a 6th-degree polynomial in u denoted $p(u)$. This polynomial can be simplified and written as follows.

$$\begin{aligned}
& c_1 V_0 c_2^2 \\
& -u \left(a c_1 c_2 c_{\max}(1 + c_2) + c_2 c_{-1} \right) \\
& +u^2 (a c_1 (1 + c^2)^2 + 2 c_1 c_2 V_0) \\
& -u^3 (a c_1 c_{\max}(1 + 2c_2) + c_{-1}) \\
& +u^4 (2 a c_1 (1 + c_2) + c_1 V_0) \\
& -u^5 (a c_1 c_{\max}) \\
& +u^6 (a c_1)
\end{aligned}$$

The solutions to $p(u) = 0$ (S22) subject to one additional condition (S23) which will be introduced in the subsequent section S1.2.2 are the steady-states of the original system.

Notice, that since all parameters in the expression above are positive, the coefficients of the polynomial are all positive in the case of even degree terms, and negative in the case of odd degree terms. We can therefore refer to Descartes' rule of signs, to conclude that the polynomial has no negative real roots. We see this by observing that $p(-u)$ has 0 sign changes between consecutive pairs of terms, and therefore there are no negative real roots. Being a 6th-degree polynomial, there can therefore be 0, 2, 4 or 6 positive real roots to the polynomial. However, it is worth emphasising that we are only interested in the roots of the polynomial p that satisfy a particular constraint as we mentioned earlier. This additional constraint states that an upper bound of the steady-states is "$u + v \leq \min\left(c_{\max}, \frac{V_0}{a}\right)$", and therefore we analyse whether the system will have any steady-states satisfying this bound or not.

First, observe that $p(0) = c_1 V_0 c_2^2 > 0$. Next, we let u_1 and v_1 be such that $u_1 + v_1 = \min\left(c_{\max}, \frac{V_0}{a}\right)$ where $u_1 \geq 0$ and $v_1 \geq 0$. Given these values, we investigate

$$q(u, v) = c_1(V_0 - a(u + v))(c_{\max} - (u + v)) - c_1 v$$

evaluated at $u = u_1, v = v_1$. We see that one of the two factors in the parentheses in the first term will be zero, and hence the whole expression is $q(u_1, v_1) = -c_1 v_1 \leq 0$. Therefore, the sign

of the polynomial at $u = u_1$ will also be negative, i.e. $p(u_1) \leq 0$. By the intermediate value theorem, we can now deduce that at least once is p zero, inside the interval $[0, u_1]$.

□

S1.2.2 Characterisation of the steady-states enabling diffusion-driven instability

As we are interested in positive steady states, the mass conservation equation for the cytosolic component (S12) yields the following

$$\begin{aligned} 0 \leq & V = V_0 - a(u+v) \\ \implies (u+v) \leq & \frac{V_0}{a} \end{aligned}$$

and in addition the total amount of membrane-bound proteins is also constrained by $c_{\max}$, i.e. $(u+v) \leq c_{\max}$. Therefore, following the analysis by Röger and Rätz [S34], we are interested in solutions within the following region of the state space.

$$\begin{aligned} \mathcal{A} &:= \left\{ \begin{pmatrix} u \\ v \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \end{pmatrix} : u + v \leq \min\left(c_{\max}, m\right) \right\} \\ \text{where } m &= \frac{V_0}{a} \end{aligned} \tag{S23}$$

This region has the property that an initial condition (u_0, v_0) that starts within $\mathcal{A}$ remains in it. Our functions satisfy the following criteria

$$\begin{aligned} & f(0,v) \geq 0, q(u,0) \geq 0 \text{ for all } 0 \leq u, v \leq \min(1, m) \\ & f(u,v) + (-f(u,v) + q(u,v)) = q(u,v) \leq 0 \text{ for all } u, v \geq 0 : u + v = \min(c_{\max}, m) \end{aligned}$$

where $-f(u,v) + q(u,v,V(u+v))$ is the right hand side of the second equation $\partial_t v$.

Now, we seek a homogeneous steady state $(u^*, v^*) \in \mathcal{A}$ satisfying the following equations.

$$f(u,v) = f(u^*, v^*) = 0 \tag{S24}$$

$$q(u,v) = q(u^*, v^*) = 0 \tag{S25}$$

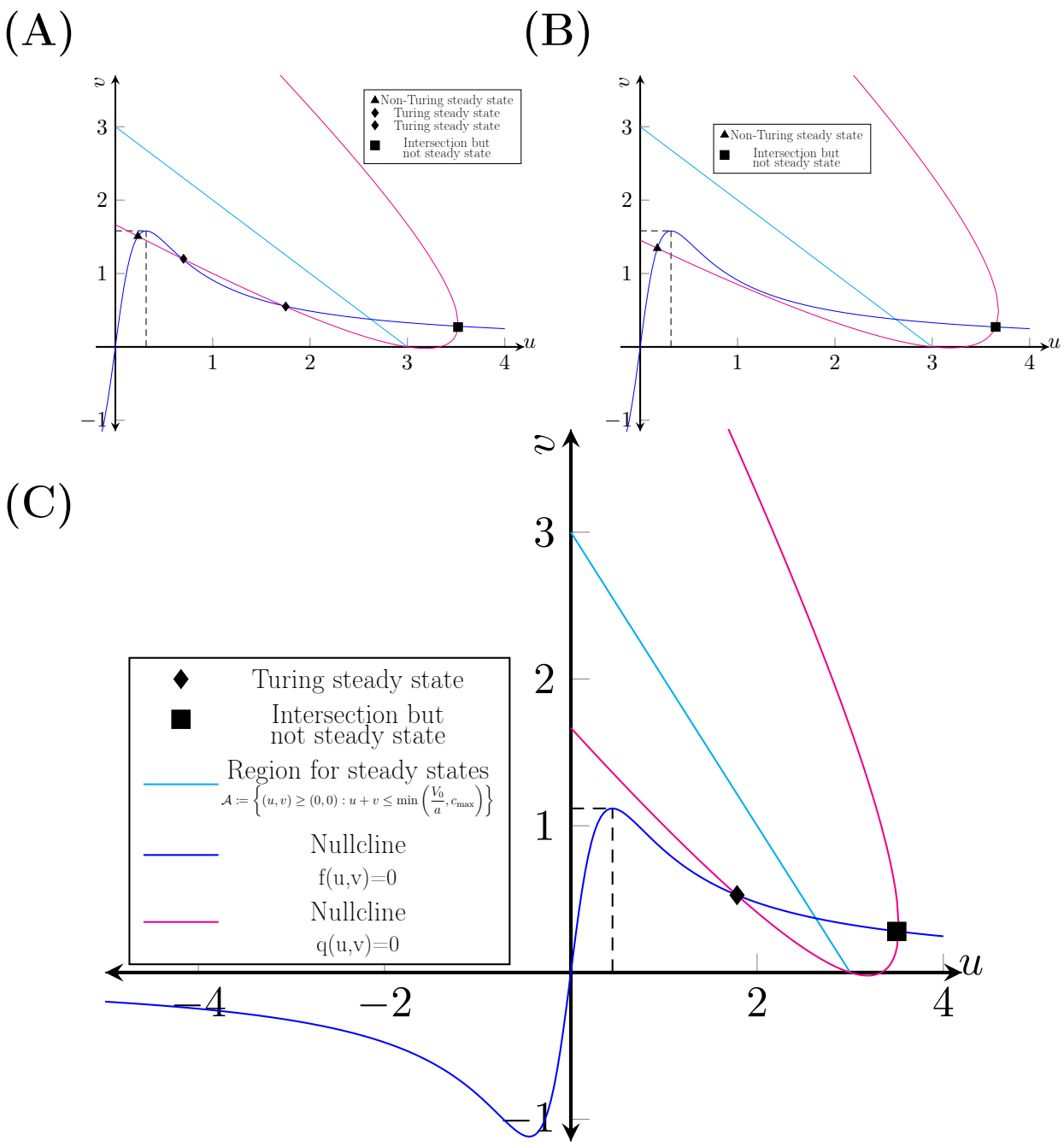

Figure S7: Steady States. The steady states of the system are located below within the region $\mathcal{A}$ in the first quadrant (the cyan curve). The intersections of two curves define the steady states: the nullcline corresponding to the activation-inactivation reaction $f(u,v) = c_2 v - u + u^2 v$ (the blue curve) and the nullcline corresponding to the transfer of inactive Cdc42 between the cytosol and the membrane $q(u,v) = c_1 (V_0 - a(u+v)) (c_{\max} - (u+v)) - c_{-1} v$ (the magenta curve). The number of steady states determined by the intersections of the two nullclines are visualised for three cases. **(A)** *Four steady states* $(c_{-1}, c_2) = (0.30, 0.10)$: one non-Turing steady state ▲, two Turing steady states ♦ and one intersection but not a steady state ■. **(B)** *Two steady states* $(c_{-1}, c_2) = (0.30, 0.20)$: one non-Turing steady state ▲, and one intersection but not a steady state ■. **(C)** *Two steady states* $(c_{-1}, c_2) = (0.20, 0.20)$: one Turing steady state ♦, and one intersection but not a steady state ■. The other parameters are: $V_0 = 10$, $a = 3$, $c_{\max} = 3$ and $c_1 = 0.05$.

The first nullcline (S24) implies that the steady state of interest lies on the curve

$$v(u) = \frac{u}{c_2 + u^2} \tag{S26}$$

and thus we are interested in a u-component such that $\Phi(u) \coloneqq q(u, v(u)) = 0$. We next differentiate $v(u)$ and find that

$$v'(u) = \frac{c_2 - u^2}{(c_2 + u^2)^2} = \frac{(\sqrt{c_2} - u)(\sqrt{c_2} + u)}{(c_2 + u^2)^2}$$

which has a positive root[S2] at $u_0 = \sqrt{c_2}$, where $v(u_0) = 1/(2\sqrt{c_2})$. We also find that $v'(u) < 0$ for all $u > u_0$ and that $v'(u) > 0$ for all $u < u_0$. It is the switch of signs around the critical point u_0 which we will use to characterise the desired steady-states.

Since, $f(u, v) = c_2 v - u + u^2 v$, the corresponding partial derivatives are

$$\begin{aligned} f_v &= c_2 + u^2 > 0 \\ f_u &= 2uv - 1 \end{aligned}$$

and since these terms are evaluated at the steady state we can substitute $v(u)$ (S26) into the latter expression for f_u above to obtain the following.

$$\begin{aligned} f_u &= 2uv - 1 \\ &= 2u\left(\frac{u}{c_2 + u^2}\right) - 1 \\ &= \frac{2u^2 - (c_2 + u^2)}{c_2 + u^2} \\ &= \frac{u^2 - c_2}{c_2 + u^2} \\ &= -\left(\underbrace{\frac{c_2 - u^2}{(c_2 + u^2)^2}}_{=v'(u)}\right)\underbrace{(c_2 + u^2)}_{=f_v} \\ &= -v'(u) f_v \end{aligned}$$

Now, the partial derivatives are evaluated at the steady-state (u^*, v^*) which can be characterised based on the sign of the derivative v_u at the steady-state. It is known that a steady-state allowing for diffusion-driven instability has a Jacobian matrix with elements with particular signs (S19). As a consequence of the negativity of the partial derivatives of q, the only way to obtain opposite signs of the diagonal elements $J(1,1)$ and $J(2,2)$ is if the following holds.

$$\begin{aligned} f_v &= c_2 + u^2 > 0 \\ f_u &= -v'(u) f_v > 0 \end{aligned} \tag{S27}$$

It is clear that $f_v > 0$ however $f_u > 0$ implies that $u^* > u_0$ for the critical point $u_0 = \sqrt{c_2}$. This follows from the fact that we have that $v'(u) < 0 \forall u > u_0$, and $-v'(u) > 0 \forall u > u_0$. Combining this with the fact that the steady-states should be located withing the region $\mathcal{A}$ (S23) yields the following characterisation of the desired steady-state.

[S2] The critical point u_0 is a local max-point of the function $v(u)$.

$$u^* \in (u_0, \min(c_{\max}, m)) \tag{S28}$$

In fact, this requirement indicate that the signs of the Jacobian must satisfy the lower of the two cases in terms of the signs of the elements (S19). Also, a necessary requirement to ensure that such a steady-state exists is that the local maximum

$$(u_0, v_0) = \left(\sqrt{c_2}, \frac{1}{2\sqrt{c_2}}\right)$$

of $v(u)$ (S26) is located within the region $\mathcal{A}$ (S23). This requirement is exactly translated to the condition

$$\sqrt{c_2} + \frac{1}{2\sqrt{c_2}} \leq \min\{c_{\max}, m\}$$

which completes the proof of Theorem 1.

□

S2 Numerical implementations

The numerical implementations consist of two parts. Firstly, the pseudo-code for the visualisation of the parameter space (Fig 2) giving rise to diffusion-driven instability is presented. Secondly, the method for finding the approximate solutions of the dimensionless model (6) is presented.

S2.1 Numerical mapping of the parameter space

The numerical mapping of the parameter space consists of three steps. For the sake of simplicity, assume that the (c_{-1}, c_2)-space[S3] is to be calculated. Then, the initial step is to discretise both the c_{-1}- and c_2-line into a finite number, $N \in \mathbb{N}_+$, of nodes. Using this partitioning, allocate memory for the solution matrix $C \in \mathbb{R}_+^{N\times N}$, i.e. $C \leftarrow \mathbf{0}^{N\times N}$. Then loop over the nodes in the partitioning and do the following two steps for all nodes.

1. Calculate the steady states $(u^*, v^*) \in \mathcal{A}$
2. Check the desired conditions for diffusion-driven instability
   (a) **Classic case**: Check the conditions (S13), (S14), (S15), (S16) and (S17). If they are satisfied, assign $C(i, j) \leftarrow 1.0$ for the specific indices $i, j \in \{1, \ldots, N\}$ of interest
   (b) **Non-classic case**: Check the conditions (S13), (S14), (S20) and (S21). If they are satisfied, assign $C(i, j) \leftarrow 0.5$ for the specific indices $i, j \in \{1, \ldots, N\}$ of interest

Depending on the specific case, a matrix with the value 1.0 for all parameters giving rise to classic diffusion-driven instability is obtained or a corresponding matrix with the value 0.5 in the non-classic case.

To calculate the steady-states, we used a Newton-iteration with a given start guess. To this end, we take the function v (S26) corresponding to the nullcline $f(u, v) = 0$. Similarly, we rewrite the nullcline $q(u, v) = 0$ as a function of u in order to obtain the following two segments.

[S3] The (c_{-1}, c_2)-space is generated completely analogously.

$$v_{q,1}(u) = \frac{1}{2}\left(K + \sqrt{K^2 - 4L}\right) \tag{S29}$$

$$v_{q,2}(u) = \frac{1}{2}\left(K - \sqrt{K^2 - 4L}\right) \tag{S30}$$

where

$$A = c_{\max} - u$$
$$B = V_0 - a\,u$$
$$K = \frac{1}{a}\left(aA + B + \frac{c_{-1}}{c_1}\right)$$
$$L = \frac{AB}{a}$$

Now, given a start-guess $u = u_0$, we solve the equations $v(u) - v_{q,1}(u) = 0$ and $v(u) - v_{q,2}(u) = 0$ numerically using the function "`fzero`" in Matlab [S30]. In order to make sure that the solver actually finds the steady-states, we take a large number of start guesses (in fact, we have used 50 start guesses) in the interval $[u_0, \min(c_{\max}, m)]$ and we control that each value u^* that the solver converges to satisfies the equation $q\left(u^*, \frac{u^*}{c_2 + (u^*)^2}\right) = 0$. If so, the value $(u^*, v(u^*))$ is a steady state and it is consequently saved.

Lastly, given the steady state we can check the conditions involving the partial derivatives. The partial derivatives for our model are the following.

$$f_u = 2uv - 1$$
$$f_v = c_2 + u^2$$
$$q_u = (-c_1 a)(m - (u + v))$$
$$q_v = q_u - c_{-1}$$
$$q_V = c_1(c_{\max} - (u + v))$$
$$V' = -a$$

S2.2 Numerical solutions to the RD-model of cell polarisation

The numerical implementation of the solution to the problem is divided into three parts. Firstly, we set up the problem by formulating the equations of the model and the corresponding domains to the various equations. It is worth emphasising that in the analysis, it is assumed that the cytosolic diffusion goes to infinity, i.e. $D \to \infty$ while this is not the case for the numerical solutions. Moreover, as the variables in the PDE-problem are $\mathbf{x} \in \mathbb{R}^3$ corresponding to the *spatial* dimension and $t \in \mathbb{R}_+$ corresponding to time, two discretisations corresponding to these variables are required. Firstly, a *Finite Element Method* (*FEM*) is implemented corresponding to the spatial discretisation. Secondly, a *Finite Difference* (*FD*)-method is implemented corresponding to the discretisation in time. These two discretisations are presented subsequently after the numerical setting is introduced.

S2.2.1 Introduction to the numerical implementation: Setting up the problem

The numerical solution to the original RD problem (Eq (6) on page 7) accounts for the domains for the problem (Fig S8). These are the cytosol Ω corresponding to the interior of the spherical cell and the cell-membrane Γ being its surface (Fig S8).

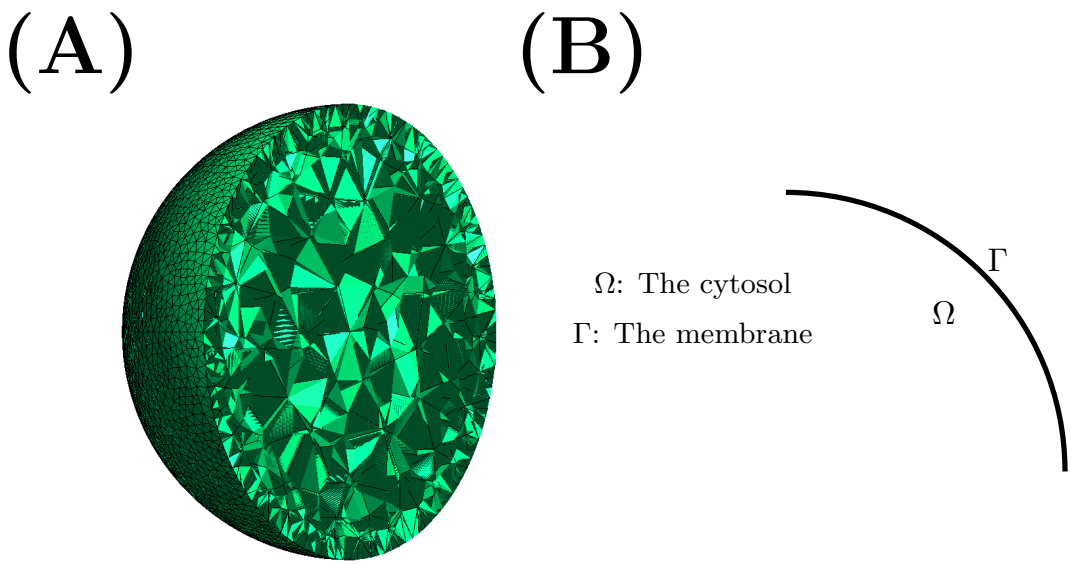

Figure S8: The detailed Spatial Domain. (**A**) *Spatial mesh for the numerical solution.* The discretisation of the spatial domain represented as a mesh. The mesh is non-uniformly discretised in the sense that the node-density is higher at the membrane than in the cytosol. (**B**) *Geometric domain.* By letting the membrane thickness shrink to zero, the geometric description is simplified to one domain Ω corresponding to the cytosol and one boundary Γ corresponding to the membrane

The domain (S31) and interfaces (S32) are important to define as the various equations and boundary conditions are based on the spatial description. Note that the domains are the unit sphere and its surface which correspond to the geometric description of the dimensionless model.

$$\Omega := \left\{ \mathbf{x} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \in \mathbb{R}^3 : x^2 + y^2 + z^2 < 1 \right\} \tag{S31}$$

$$\Gamma := \left\{ \mathbf{x} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \in \mathbb{R}^3 : x^2 + y^2 + z^2 = 1 \right\} \tag{S32}$$

Now, given the definition of the spatial domain Ω the formulation of the problem is the following. In the membrane Γ the active u and inactive v form undergo activation-inactivation reactions determined by the function $f(u, v) = c_2 v - u + u^2 v$ and diffusion. The interface Γ has a flux of the inactive membrane bound component v from the membrane to the cytosol, a flux of the GDI-bound cytosolic component V from the cytosol to the membrane and no flux of the active membrane bound component u. The flux over the membrane is determined by the function $q(u, v, V) = c_1 V(c_{\max} - (u+v)) - c_1 v$ and it is translated into a contribution to the reaction term for the inactive membrane-bound species v and a Robin-boundary condition for the cytosolic component V at Γ. Lastly, in the cytosol, Ω, the GDI-bound component V undergoes diffusion. Using these equations and boundary conditions it is possible to formulate the detailed model of Cdc42 activation (S33).

$$
\left.\begin{aligned}
\frac{\mathrm{d}V}{\mathrm{d}t} &= D\Delta V
\end{aligned}\right\}\mathbf{x}\in\Omega,\ \tau\in\mathbb{R}_+
$$

$$
\begin{aligned}
\left.\begin{aligned}
-D\left[(\nabla V)^T\cdot\mathbf{n}\right] &= \gamma\left\{c_1V\left(c_{\max}-(u+v)\right)-c_{-1}v\right\}\\
&= \gamma q(u,v,V)
\end{aligned}\right\}\mathbf{x}\in\Gamma,\ \tau\in\mathbb{R}_+\\
\left.\begin{aligned}
\frac{\mathrm{d}u}{\mathrm{d}\tau} &= \gamma\left(c_2v-u+u^2v\right)+\Delta u\\
&= \gamma f(u,v)+\Delta u\\
\frac{\mathrm{d}v}{\mathrm{d}\tau} &= \gamma\left(-f(u,v)+q(u,v,V)\right)+d\Delta v
\end{aligned}\right\}\mathbf{x}\in\Gamma,\ \tau\in\mathbb{R}_+
\end{aligned}
\tag{S33}
$$

The various notations above are the following: $\nabla = \begin{pmatrix}\frac{\partial}{\partial x}\\ \frac{\partial}{\partial y}\\ \frac{\partial}{\partial z}\end{pmatrix}$ is the gradient operator, "^{T}" is the transpose operator, $\mathbf{n}\in\Omega\subset\mathbb{R}^3$ is the *outward normal* at a specific position on a given surface and $\Delta = \frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}+\frac{\partial^2}{\partial z^2}$ is the Laplace operator. In order to solve the problem (S33) numerically, the spatial domain must be discretised.

The spatial discretisation of the domain is corresponds to a *mesh* (Fig S8A). As the interesting parts of the model regards the reactions and diffusion of the membrane-bound species u and v it is advantageous from a computational view to use a non-uniform mesh. More precisely, we have implemented a mesh with higher node density close to the membrane and a low node density in the interior of the cell. To generate the mesh over the domain, we have used the three dimensional finite element mesh generator called "Gmsh" [S10]. Given the generated mesh, it is possible to numerically solve the problem (S33) using the finite element method.

For the FD- and FEM-implementations, the computing platform FEniCS [S15, S17, S18, S21, S22, S19, S24, S32, S20, S25, S37, S4, S6, S28, S33, S11, S14, S36, S5, S2, S26, S29, S27, S16, S7, S3, S1, S12, S13] has been used. In FEniCS, it suffices to provide the so called *variational formulation* of the problem (S33) which is a reformulation of the problem. The approximation to this problem is then obtained by projecting the variational formulation onto a space of piece wise continuous functions or in our case a test function space of piece-wise linear functions. In the next chapter, we will merely present the derivation of the variational formulation regarding the spatial discretisation, and after that we will present the finite difference implementation for the temporal discretisation based on this variational formulation.

S2.2.2 An implementation of the Finite Element Method in space

Define the following function space for *test functions* ϕ on the domain Ω.

$$
H^1(\Omega) := \left\{\phi\in\mathcal{C}^1(\Omega) : \|\phi\|^2_{\mathcal{L}_2(\Omega)}+\|\phi'\|^2_{\mathcal{L}_2(\Omega)}<\infty\right\} \tag{S34}
$$

This space consists of continuous functions with continuous first derivatives with compact support.

As is standard in the formulation of the variational formulation, we multiply the three PDEs (S33) with three test functions $\phi_1,\phi_2,\phi_3\in H^1(\Omega)$ and integrate over the domain, Ω.

$$
\begin{aligned}
\int_\Gamma \frac{\mathrm{d}u}{\mathrm{d}\tau}\phi_1(\mathbf{x})\mathrm{d}\mathbf{x} &= \int_\Gamma \Delta u\phi_1(\mathbf{x})\mathrm{d}\mathbf{x} + \gamma\int_\Gamma f(u,v)\phi_1(\mathbf{x})\mathrm{d}\mathbf{x}\\
\int_\Gamma \frac{\mathrm{d}v}{\mathrm{d}\tau}\phi_2(\mathbf{x})\mathrm{d}\mathbf{x} &= d\int_\Gamma \Delta v\phi_2(\mathbf{x})\mathrm{d}\mathbf{x} + \gamma\int_\Gamma \left(-f(u,v)+q(u,v,V)\right)\phi_2(\mathbf{x})\mathrm{d}\mathbf{x}\\
\int_\Omega \frac{\mathrm{d}V}{\mathrm{d}\tau}\phi_3(\mathbf{x})\mathrm{d}\mathbf{x} &= \int_\Omega D\ \Delta V\phi_3(\mathbf{x})\mathrm{d}\mathbf{x}
\end{aligned}
$$

Now, to rewrite the diffusion terms, we will use *Green's first identity*[S23]

$$
\int_\Omega v\Delta u\mathrm{d}\mathbf{x} = \int_\Gamma v\ ((\nabla u)^T\cdot\mathbf{n})\mathrm{d}s - \int_\Omega (\nabla v)^T\cdot(\nabla u)\mathrm{d}\mathbf{x}
$$

where $\nabla = \begin{pmatrix}\frac{\partial}{\partial x}\\ \frac{\partial}{\partial y}\\ \frac{\partial}{\partial z}\end{pmatrix}$ is the gradient, $\mathbf{n}\in\mathbb{R}^3$ is the outward normal and ^{T} is the transpose operator.

Using this identity on the diffusion terms above one obtains

$$
\begin{aligned}
\int_\Gamma \frac{\mathrm{d}u}{\mathrm{d}\tau}\phi_1(\mathbf{x})\mathrm{d}\mathbf{x} &= -\int_\Gamma (\nabla\phi_1(\mathbf{x}))^T\cdot(\nabla u)\mathrm{d}\mathbf{x} + \gamma\int_\Gamma f(u,v)\phi_1(\mathbf{x})\mathrm{d}\mathbf{x}\\
\int_\Gamma \frac{\mathrm{d}v}{\mathrm{d}\tau}\phi_2(\mathbf{x})\mathrm{d}\mathbf{x} &= -d\int_\Gamma (\nabla\phi_2(\mathbf{x}))^T\cdot(\nabla v)\mathrm{d}\mathbf{x} + \gamma\int_\Gamma (-f(u,v)+q(u,v,V))\phi_2(\mathbf{x})\mathrm{d}\mathbf{x}\\
\int_\Omega \frac{\mathrm{d}V}{\mathrm{d}\tau}\phi_3(\mathbf{x})\mathrm{d}\mathbf{x} &= D\int_\Gamma \phi_3(\mathbf{x})\ ((\nabla V)^T\cdot\mathbf{n})\mathrm{d}s - \int_\Omega D(\nabla\phi_3(\mathbf{x}))^T\cdot(\nabla V)\mathrm{d}\mathbf{x}
\end{aligned}
$$

and simplifying the above yields the following.

$$
\begin{aligned}
\int_\Gamma \frac{\mathrm{d}u}{\mathrm{d}\tau}\phi_1(\mathbf{x})\mathrm{d}\mathbf{x} &= -\int_\Gamma (\nabla\phi_1(\mathbf{x}))^T\cdot(\nabla u)\mathrm{d}\mathbf{x} + \gamma\int_\Gamma f(u,v)\phi_1(\mathbf{x})\mathrm{d}\mathbf{x}\\
\int_\Gamma \frac{\mathrm{d}v}{\mathrm{d}\tau}\phi_2(\mathbf{x})\mathrm{d}\mathbf{x} &= \gamma\int_\Gamma q(u,v,V)\phi_2(\mathbf{x})\mathrm{d}s - d\int_\Gamma (\nabla\phi_2(\mathbf{x}))^T\cdot(\nabla v)\mathrm{d}\mathbf{x} - \gamma\int_\Gamma f(u,v)\phi_2(\mathbf{x})\mathrm{d}\mathbf{x}\\
\int_\Omega \frac{\mathrm{d}V}{\mathrm{d}\tau}\phi_3(\mathbf{x})\mathrm{d}\mathbf{x} &= -\gamma\int_\Gamma q(u,v,V)\phi_3(\mathbf{x})\mathrm{d}s - \int_\Omega D(\nabla\phi_3(\mathbf{x}))^T\cdot(\nabla V)\mathrm{d}\mathbf{x}
\end{aligned}
$$

Lastly, moving all terms in the right hand side to the left hand side and adding all three equations yields the *variational formulation* of the detailed problem (S33).

Find $u(\tau), v(\tau), V(\tau) \in H_0^1(\Omega)$ for a fixed $\tau \in \mathbb{R}_+$ such that

$$F(u(\tau), v(\tau), V(\tau), \phi_1, \phi_2, \phi_3) = 0 \;\; \forall \phi_1, \phi_2, \phi_3 \in H_0^1(\Omega) \tag{S35}$$

where

$$\begin{aligned} F(u(\tau), v(\tau), V(\tau), \phi_1, \phi_2, \phi_3) = & \\ & \int_\Gamma \frac{\mathrm{d}u}{\mathrm{d}\tau}\phi_1(\mathbf{x})\mathrm{d}\mathbf{x} + \int_\Gamma (\nabla\phi_1(\mathbf{x}))^T \cdot (\nabla u(\tau))\mathrm{d}\mathbf{x} \\ & + \int_\Gamma \frac{\mathrm{d}v}{\mathrm{d}\tau}\phi_2(\mathbf{x})\mathrm{d}\mathbf{x} + d\int_\Gamma (\nabla\phi_2(\mathbf{x}))^T \cdot (\nabla v(\tau))\mathrm{d}\mathbf{x} \\ & + \gamma \int_\Gamma f(u(\tau), v(\tau))(\phi_2(\mathbf{x}) - \phi_1(\mathbf{x}))\mathrm{d}\mathbf{x} \\ & + \int_\Omega \frac{\mathrm{d}V}{\mathrm{d}\tau}\phi_3(\mathbf{x})\mathrm{d}\mathbf{x} + D\int_\Omega (\nabla\phi_3(\mathbf{x}))^T \cdot (\nabla V(\tau))\mathrm{d}\mathbf{x} \\ & + \gamma \int_\Gamma q(u(t), v(t), V(t))(\phi_3(\mathbf{x}) - \phi_2(\mathbf{x}))\mathrm{d}s \end{aligned}$$

The numerical solution of this problem (S35) is written as linear combinations of piece-wise continuous basis functions of order 1, i.e. linear functions. Assume that a spatial discretisation (Fig S8A) is given and denote this by τ_h. To describe the finite element method, denote the space of piece-wise linear functions on τ_h by $\mathfrak{H}(\Omega)$. These tetrahedron-like functions take the value one on each node in the grid (Fig S8A) and they take the value zero on the neighbouring nodes. Then, the finite-element solution $\mathfrak{u}(\tau), \mathfrak{v}(\tau), \mathfrak{V}(\tau) \in \mathfrak{H}(\Omega)$ for a fixed $\tau \in \mathbb{R}_+$ to the variational formulation (S35) consists of the orthogonal projections of $u(\tau), v(\tau), V(\tau) \in H_0^1(\Omega)$ onto $\mathfrak{H}(\Omega)$ for a fixed $\tau \in \mathbb{R}_+$. Here, we mean projection in the sense of the classic $\mathcal{L}_2$-*inner product*.

$$\langle f, g\rangle = \int_\Omega f(\mathbf{x})g(\mathbf{x})\,\mathrm{d}\mathbf{x}$$

It is this variational formulation (S35) that is solved in FEniCS approximately using the finite element method just described. Subsequently, the time derivatives will be approximated using a finite difference scheme.

S2.2.3 An implementation of the Finite Difference Method in time

For the implementation of the finite difference scheme for solving the problem (S35) numerically, we use an *mixed Implicit-Explicit scheme*. The solution of the ODE problem

$$\frac{\mathrm{d}y}{\mathrm{d}t} = f(y(t))$$

with $f \in \mathcal{C}(\mathbb{R})$ is the function $y \in \mathcal{C}(\mathbb{R}_+)$ with the time as variable $t \in \mathbb{R}_+$. Initially, the general methodology for finite differences involves *discretising* the time-line into various nodes, where the partitioning of the time-line is denoted as follows $\mathcal{T}([0, t_{\max}]) = \{t_0, t_1, t_2, \ldots, t_n, \ldots, t_{\max}\}$. Moreover, denote each of these nodes by $t_n \in \mathbb{R}_+$ for some index $n \in \mathbb{Z}_+$ where the previous node on the discretised time line is denoted t_{n-1}. Then, the backward-Euler algorithm for the above problem entails solving solve the following equation iteratively

$$\frac{y(t_n) - y(t_{n-1})}{k_n} = f(y(t_n))$$

where $k_n = t_n - t_{n-1} \in \mathbb{R}_+$ is the so called *step size*. The solution to the above equation corresponds to the solution $y(t_n)$ in the current node given the solution in the previous node $y(t_{n-1})$. Note that both the left and the right hand side in the above equation depend on the solution $y(t_n)$ in the current node which implies that this algorithm is *implicit*. The forward version of the Euler algorithm consists of solving

$$\frac{y(t_n) - y(t_{n-1})}{k_n} = f(y(t_{n-1}))$$

which yields the iterative scheme $y(t_n) = y(t_{n-1}) + k_n f(y(t_{n-1}))$ which is *explicit* and can be solved directly. However, the implicit version is more computationally expensive and has to be solved by using for example Newton's method [S9, S8]. Despite the implicit algorithm being more computationally expensive compared to the forward or explicit version of the Euler scheme for ODEs, the advantage of it is that it is *unconditionally stable* as oppose the the explicit algorithm which is merely *conditionally stable*. The numerical solution of the detailed problem (S33) is given by combining the finite element method based on the previous variational formulation (S35) with a mixed explicit-implicit Euler finite difference scheme described above, which results in the following variational formulation (S36).

$$\begin{aligned}
&\text{Find } \; u(\tau_n), v(\tau_n), V(\tau_n) \in H_0^1(\Omega) \\
&\quad F(u(\tau_n), v(\tau_n), V(\tau_n), u(\tau_{n-1}), v(\tau_{n-1}), V(\tau_{n-1}), \phi_1, \phi_2, \phi_3) = 0 \;\; \forall \phi_1, \phi_2, \phi_3 \in H_0^1(\Omega) \\
&\qquad\qquad \text{where} \\
&\quad F(u(\tau_n), v(\tau_n), V(\tau_n), u(\tau_{n-1}), v(\tau_{n-1}), V(\tau_{n-1}), \phi_1, \phi_2, \phi_3) = \\
&\qquad \int_\Gamma \left(\frac{u(\tau_n) - u(\tau_{n-1})}{\tau_n - \tau_{n-1}} \right) \phi_1(\mathbf{x}) \mathrm{d}\mathbf{x} + \int_\Gamma (\nabla \phi_1(\mathbf{x}))^T \cdot (\nabla u(\tau_n)) \mathrm{d}\mathbf{x} \\
&\qquad + \int_\Gamma \left(\frac{v(\tau_n) - v(\tau_{n-1})}{\tau_n - \tau_{n-1}} \right) \phi_2(\mathbf{x}) \mathrm{d}\mathbf{x} + d \int_\Gamma (\nabla \phi_2(\mathbf{x}))^T \cdot (\nabla v(\tau_n)) \mathrm{d}\mathbf{x} \\
&\qquad + \gamma \int_\Gamma f(u(\tau_n), v(\tau_n), u(\tau_{n-1}), v(\tau_{n-1}))(\phi_2(\mathbf{x}) - \phi_1(\mathbf{x})) \mathrm{d}\mathbf{x} \\
&\qquad + \int_\Omega \left(\frac{V(\tau_n) - V(\tau_{n-1})}{\tau_n - \tau_{n-1}} \right) \phi_3(\mathbf{x}) \mathrm{d}\mathbf{x} + D \int_\Omega (\nabla \phi_3(\mathbf{x}))^T \cdot (\nabla V(\tau_n)) \mathrm{d}\mathbf{x} \\
&\qquad + \gamma \int_\Gamma q(u(\tau_n), v(\tau_n), u(\tau_{n-1}), v(\tau_{n-1}), V(\tau_{n-1}))(\phi_3(\mathbf{x}) - \phi_2(\mathbf{x})) \mathrm{d}s
\end{aligned} \tag{S36}$$

The above approach is referred to as an *mixed Implicit-Explicit* approach. This approach is implicit in the sense that the linear terms, i.e. the time derivatives approximated by finite differences and the diffusive terms containing the gradients, are implicit. These will result in the classic mass- and stiffness-matrices for the time derivatives and diffusive terms respectively when the finite element method is implemented. By "trial and error" we found that a *mixed implicit-explicit* methodology worked in order to get accurate simulations.

$$f(u(\tau_n), v(\tau_n), u(\tau_{n-1}), v(\tau_{n-1})) = c_2 v(\tau_n) - u(\tau_n) + u(\tau_{n-1})^2 v(\tau_{n-1})$$

$$\begin{aligned}
&q(u(\tau_n), v(\tau_n), u(\tau_{n-1}), v(\tau_{n-1}), V(\tau_{n-1})) = \\
&c_1(V(\tau_{n-1}) - a(u(\tau_{n-1}) + v(\tau_{n-1})))(c_{\max} - (u(\tau_{n-1}) + v(\tau_{n-1}))) - c_{-1} v(\tau_n)
\end{aligned} \tag{S37}$$

As can be seen, all purely linear terms are treated implicitly while the purely non-linear terms are treated explicitly. Furthermore, we implemented an adaptive explicit time-stepping procedure. The time-stepping procedure is based on calculating the residual Res based on Eq

S36. In each time-step, we calculate a possible maximum time step $\mathrm{d}\tau$ by setting it inversely proportional to R

$$\delta\tau \leftarrow \frac{\mathrm{TOL}}{\mathrm{Res}}$$

where the tolerance TOL is set beforehand. We also save the previous time step $\Delta\tau$

$$\Delta\tau \leftarrow (\tau_n - \tau_{n-1})$$

and then the new time step is calculated by using the arithmetic mean according to the following rule.

$$k \leftarrow \min\left(\frac{2\cdot\Delta\tau\cdot\delta\tau}{\Delta\tau+\delta\tau}, k_{\max}\right)$$

In similarity with the tolerance TOL, another numerical parameter corresponding to the maximum step length $k_{\max}$ is defined.

To test the validity of our implementation, a test-problem was constructed. Here we solved the homogeneous ODE-problem

$$\begin{aligned}
\frac{\mathrm{d}u}{\mathrm{d}\tau} &= \gamma\left(c_2 v - u + u^2 v\right) \\
&= \gamma f(u,v) \\
\frac{\mathrm{d}v}{\mathrm{d}\tau} &= \gamma\left(f(u,v) + [c_1(V_0 - a(u+v))(c_{\max} - (u+v)) - c_{-1}v]\right) \\
&= \gamma(-f(u,v) + q(u,v))
\end{aligned}$$

with trivial initial conditions.

$$u_0 = v_0 = 0$$

Then, compared the spatial averages of the PDE-solutions to Eq S36 given by

$$\overline{u}(\tau) = \frac{1}{|\Gamma|}\int_\Gamma u(\mathbf{x},\tau)\,\mathrm{d}s \;\;\&\;\; \overline{v}(\tau) = \frac{1}{|\Gamma|}\int_\Gamma v(\mathbf{x},\tau)\,\mathrm{d}s$$

with the ODE-solutions. Again the initial conditions of u and v in the PDE-setting where set to 0 everywhere in the domain while the initial conditions of V was set to $V(\mathbf{x},0) = V_0 \forall \mathbf{x}\in\Omega$. The spatial averages $\overline{u}(\tau), \overline{v}(\tau)$ corresponded to the ODE-solutions $u(\tau), v(\tau)$ for the mixed implementation (Eq (S37)) of the reaction term (Fig S9) which validates our implementation.

However, the non-linear reaction terms f and q are evaluated at the previous time node and thereby the algorithm is explicit with respect to the "complicated terms". A consequence of this is that a large value of the relative diffusion d will render the solution algorithm relatively more implicit (and thus more stable) while a large value of the reaction strength γ will render the algorithm relatively more explicit (and thereby less stable). An advantage of the above implementation is that is faster to solve compared to a purely implicit algorithm, however a small time-step $k_n = \tau_n - \tau_{n-1}$ is required to ensure stability. Moreover, the initial condition for the above time stepping procedure is set to

$$u(x_i,\tau_0) = u(x_i,0) = u^\star + \epsilon_{1,i},\;\; v(x_i,\tau_0) = v(x_i,0) = v^\star + \epsilon_{2,i} \;\text{ and }\; V(x_i,\tau_0) = V(x_i,0) = V^\star + \epsilon_{3,i}$$

for all spatial nodes $x_i, i = 1,\ldots,N$ where $N\in\mathbb{N}_+$ is the total number of nodes. Above, we set the initial conditions for all spatial nodes to the steady-state value (u^*, v^*, V^*) at which

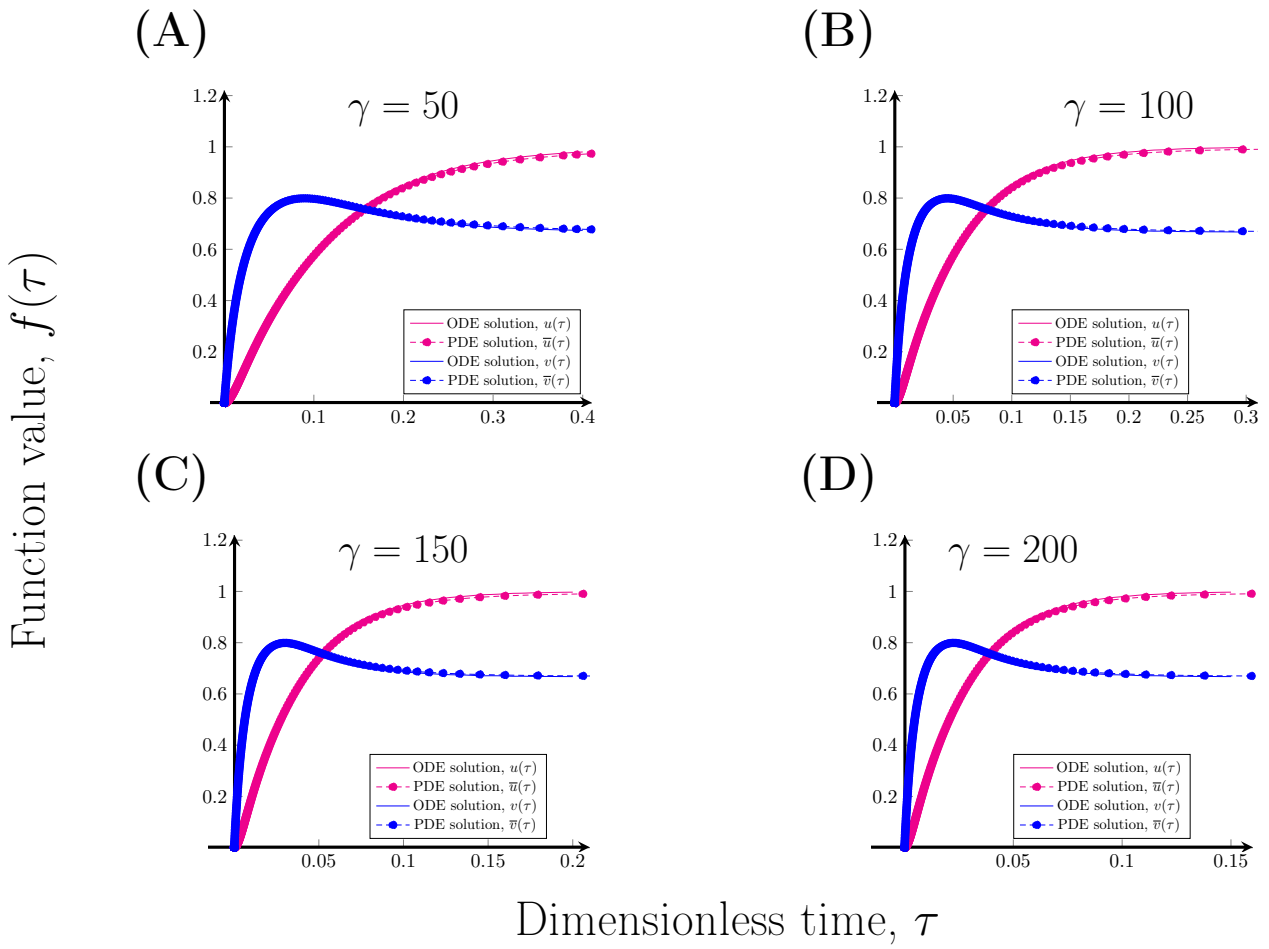

Figure S9: Validation of FEM-FD implementation. The states are plotted over time where the active state corresponds to the blue graph and the magenta graph to the inactive state. The ODE solutions u, v are represented withe whole lines while the spatial averages of the PDE solutions $\overline{u}, \overline{v}$ are represented by the dashed lines. The comparison is made in four cases corresponding to **(A)** $\gamma = 50$, **(B)** $\gamma = 100$, **(C)** $\gamma = 150$, and **(D)** $\gamma = 200$. The other parameters used in the simulations are: $c_1 = 0.05$, $c_{-1} = 0.10$, $c_1 = 0.50$, $D = 1000$, $V_0 = 6.0$ and $c_{\max} = 3.0$.

diffusion-driven instability occurs plus a noise term $\epsilon_{k,i} \in \mathcal{N}(0, \sigma)$, $k \in \{1, 2, 3\}$ which is normally distributed with zero mean and standard deviation σ. For the implementation, we have implemented a noise term of $\sigma = 0.1$.

The presented simulations corresponds to the numerical solutions to the problem (S33) given by this combined approach. In summary, our methodology entails applying the FEM on the above variational formulation (S36) repeatedly while advancing in time using FDs corresponding to an implicit Euler scheme. Thus, given a spatial disretisation in terms of a mesh (Fig S8B), a partitioning of the time-line $\mathcal{T}([0, \tau_{\max}]) = \{\tau_0, \tau_1, \tau_2, \ldots, \tau_n, \ldots, \tau_{\max}\}$ and initial conditions $\{\mathfrak{u}_0, \mathfrak{v}_0, \mathfrak{V}_0\} = \{\mathfrak{u}(\tau_0), \mathfrak{v}(\tau_0), \mathfrak{V}(\tau_0)\} \in \mathfrak{H}$ corresponding to the initial concentration profiles of the three states it is possible to solve the problem numerically using the finite element method based on (S36) in order to obtain the finite element solutions in each time node in $\mathcal{T}([0, \tau_{\max}])$. Regarding the initial conditions $\{\mathfrak{u}_0, \mathfrak{v}_0, \mathfrak{V}_0\} \in \mathfrak{H}$, we have initiated the three states inhomogeneously, as described above, in their respective domains (i.e. Ω for $\mathfrak{V}$ and Γ for $\mathfrak{u}$ and $\mathfrak{v}$) with a small perturbation around each steady state value.

References Supplementary

[S1] Martin S. Alnæs. *UFL: a Finite Element Form Language*, chapter 17. Springer, 2012.

[S2] Martin S. Alnæs, Jan Blechta, Johan Hake, August Johansson, Benjamin Kehlet, Anders Logg, Chris Richardson, Johannes Ring, Marie E. Rognes, and Garth N. Wells. The fenics project version 1.5. *Archive of Numerical Software*, 3(100), 2015.

[S3] Martin S. Alnæs, Anders Logg, and Kent-Andre Mardal. *UFC: a Finite Element Code Generation Interface*, chapter 16. Springer, 2012.

[S4] Martin S. Alnæs, Anders Logg, Kent-Andre Mardal, Ola Skavhaug, and Hans Petter Langtangen. Unified framework for finite element assembly. *International Journal of Computational Science and Engineering*, 4(4):231–244, 2009.

[S5] Martin S. Alnæs, Anders Logg, Kristian B. Ølgaard, Marie E. Rognes, and Garth N. Wells. Unified form language: A domain-specific language for weak formulations of partial differential equations. *ACM Transactions on Mathematical Software*, 40(2), 2014.

[S6] Martin S. Alnæs and Kent-Andre Mardal. On the efficiency of symbolic computations combined with code generation for finite element methods. *ACM Transactions on Mathematical Software*, 37(1), 2010.

[S7] Martin S. Alnæs and Kent-Andre Mardal. *SyFi and SFC: Symbolic Finite Elements and Form Compilation*, chapter 15. Springer, 2012.

[S8] Niclas Andréasson, Anton Evgrafov, Michael Patriksson, Emil Gustavsson, and Magnus Önnheim. *An introduction to continuous optimization: foundations and fundamental algorithms*, volume 28. Studentlitteratur Lund, 2005.

[S9] Ward Cheney and David Kincaid. Numerical mathematics and computation, 2004.

[S10] Christophe Geuzaine and Jean-François Remacle. Gmsh: A 3-d finite element mesh generator with built-in pre-and post-processing facilities. *International journal for numerical methods in engineering*, 79(11):1309–1331, 2009.

[S11] Johan Hoffman, Johan Jansson, Rodrigo V. de Abreu, Cem Degirmenci, Niclas Jansson, Kaspar Müller, Murtazo Nazarov, and Jeanette H. Spühler. Unicorn: Parallel adaptive finite element simulation of turbulent flow and fluid-structure interaction for deforming domains and complex geometry. *Computer and Fluids*, in press, 2012.

[S12] Johan Hoffman, Johan Jansson, Cem Degirmenci, Niclas Jansson, and Murtazo Nazarov. *Unicorn: a Unified Continuum Mechanics Solver*, chapter 18. Springer, 2012.

[S13] Johan Hoffman, Johan Jansson, Niclas Jansson, C. Johnson, and Rodrigo V. de Abreu. *Turbulent Flow and Fluid–structure Interaction*, chapter 28. Springer, 2012.

[S14] Niclas Jansson, Johan Jansson, and Johan Hoffman. Framework for massively parallel adaptive finite element computational fluid dynamics on tetrahedral meshes. *SIAM Journal on Scientific Computing*, 34(1):C24–C41, 2012.

[S15] Robert C. Kirby. Algorithm 839: Fiat, a new paradigm for computing finite element basis functions. *ACM Transactions on Mathematical Software*, 30(4):502–516, 2004.

[S16] Robert C. Kirby. *FIAT: Numerical Construction of Finite Element Basis Functions,*, chapter 13. Springer, 2012.

[S17] Robert C. Kirby, Matthew G. Knepley, Anders Logg, and L. Ridgway Scott. Optimizing the evaluation of finite element matrices. *SIAM Journal on Scientific Computing*, 27(3):741–758, 2005.

[S18] Robert C. Kirby and Anders Logg. A compiler for variational forms. *ACM Transactions on Mathematical Software*, 32(3), 2006.

[S19] Robert C. Kirby and Anders Logg. Efficient compilation of a class of variational forms. *ACM Transactions on Mathematical Software*, 33(3), 2007.

[S20] Robert C. Kirby and Anders Logg. Benchmarking domain-specific compiler optimizations for variational forms. *ACM Transactions on Mathematical Software*, 35(2):1–18, 2008.

[S21] Robert C. Kirby, Anders Logg, L. Ridgway Scott, and Andy R. Terrel. Topological optimization of the evaluation of finite element matrices. *SIAM Journal on Scientific Computing*, 28(1):224–240, 2006.

[S22] Robert C. Kirby and L. Ridgway Scott. Geometric optimization of the evaluation of finite element matrices. *SIAM Journal on Scientific Computing*, 29(2):827–841, 2007.

[S23] Stig Larsson and Vidar Thomée. *Partial differential equations with numerical methods*, volume 45. Springer Science & Business Media, 2008.

[S24] Anders Logg. Automating the finite element method. *Archives of Computational Methods in Engineering*, 14(2):93–138, 2007.

[S25] Anders Logg. Efficient representation of computational meshes. *International Journal of Computational Science and Engineering*, 4(4):283–295, 2009.

[S26] Anders Logg, Kent-Andre Mardal, Garth N. Wells, et al. *Automated Solution of Differential Equations by the Finite Element Method*. Springer, 2012.

[S27] Anders Logg, Kristian B. Ølgaard, Marie E. Rognes, and Garth N. Wells. *FFC: the FEniCS Form Compiler*, chapter 11. Springer, 2012.

[S28] Anders Logg and Garth N. Wells. Dolfin: Automated finite element computing. *ACM Transactions on Mathematical Software*, 37(2), 2010.

[S29] Anders Logg, Garth N. Wells, and Johan Hake. *DOLFIN: a C++/Python Finite Element Library*, chapter 10. Springer, 2012.

[S30] MATLAB. *version 9.4.0.813654 (R2018a)*. The MathWorks Inc., Natick, Massachusetts, 2018.

[S31] J. D. Murray. *Mathematical Biology II: Spatial Models and Biomedical Applications*, volume 18 of *Interdisciplinary Applied Mathematics*. Springer New York, 2003.

[S32] Kristian B. Ølgaard, Anders Logg, and Garth N. Wells. Automated code generation for discontinuous galerkin methods. *SIAM Journal on Scientific Computing*, 31(2):849–864, 2008.

[S33] Kristian B. Ølgaard and Garth N. Wells. Optimisations for quadrature representations of finite element tensors through automated code generation. *ACM Transactions on Mathematical Software*, 37, 2010.

[S34] Andreas Rätz and Matthias Röger. Turing instabilities in a mathematical model for signaling networks. *Journal of mathematical biology*, 65(6-7):1215–1244, 2012.

[S35] Andreas Rätz and Matthias Röger. Symmetry breaking in a bulk–surface reaction–diffusion model for signalling networks. *Nonlinearity*, 27(8):1805, 2014.

[S36] Marie E. Rognes, David A. Ham, Colin J. Cotter, and Andrew T. T. McRae. Automating the solution of pdes on the sphere and other manifolds in fenics 1.2. *Geoscientific Model Development*, 6:2099–2119, 2013.

[S37] Marie E. Rognes, Robert C. Kirby, and Anders Logg. Efficient assembly of h(div) and h(curl) conforming finite elements. *SIAM Journal on Scientific Computing*, 31(6):4130–4151, 2009.

[S38] A. M. Turing. The chemical basis of morphogenesis. *Bulletin of Mathematical Biology*, 52(1):153–197, Jan 1952.