**Custom code for bifurcation analysis, ODE, SDE and CA simulations related to Kamps et al.,2020**

Reference: Kamps et al., 2020, “Optogenetic tuning reveals Rho amplification-dependent dynamics of a cell contraction signal network”

**Instructions for code related to Kamps et al. 2020**

**Bifurcation analysis (Figures 2 H and 2I):**

Requirements:

XPP 7.0 or later

1. Start XPP 7.0 or higher

2. Run the script gef\_rho\_myo.ode

3. Find equilibria (Sing pts/Go)

4. Start bifurcation analysis (File/Auto)

5. Enter the following settings:

Parameters:

Par1:Gt

Par2:Mt

Axes:

hI-lo

Y-axis:RA

Xmin:0

Ymin:0

Xmax:3

Ymax:1

Numerics:

Ntst: 500

Nmax: 2500

Npr:500

Ds.-0.001

Dsmin: 0.0001

Dsmax:0.05

Par Max:3

5. Perform steady-state analysis (Run/steady-state).

6. Pick the Hopf bifurcation at 0.6216 (Grab/press Tab until HB at Gt=0.6216 is selected/press Enter).

7. Perform periodic analysis (Run/periodic) to finish 1-parameter bifurcation analysis.

8. Pick the Hopf bifurcation at 0.6216 (Grab/press Tab until HB at Gt=0.6216 is selected/press Enter).

9. Adjust the following settings:

Axes:

Two par

Xmin:0

Ymin:0

Xmax:3

Ymax:10

10. Perform 2-parameter analysis analysis (Run/Two param).

11. Pick the Hopf bifurcation at 0.6216 (Grab/press Tab until HB at Gt=0.6216 is selected/press Enter).

12. Adjust the following settings:

Numerics:

Ds.0.001

10. Perform 2-parameter analysis analysis (Run/Two param) to finish 2-parameter bifurcation analysis. Adjust axes (Xmax: 1 and Ymax:1) and click reDraw to zoom into most relevant parameter regions.

**ODE and SDE simulations (Figures 2 J and 3B, 4A-D, 4F-G):**

Requirements:

Matlab R2018 or later

1. Open Matlab R2018 or later

2. Open the directory which contains the ode\_sims.m and ode\_euler\_noise\_model2.m scripts.

3. Run ode\_sims.m

4. Choose, which data should be generated. Please note that only one repeat of stochastic simulations will be generated and that the parameter ranges are smaller than in the publication to facilitate rapid data generation. The parameter ranges can be adjusted in the ode\_sims.m script.

**CA simulations (Figures 5B,5C and 5E):**

Requirements:

Cellular automata code from (Schmick et al., 2014).

Matlab R2018 or later

1. Merge supplied code with cellular automata code (Schmick et al., 2014) into one folder

2. Compile C++ code by typing mex maltesCA.cpp vec3d.cpp in the MATLAB console.

3. Run launch\_CA.m

4. Choose, which data should be generated.