A feedback quenched repressilator produces Turing pattern with one diffuser

Justin Hsia, William J. Holtz, Daniel C. Huang, Murat Arcak, and Michel M. Maharbiz

Department of Electrical Engineering and Computer Sciences, University of California, Berkeley

Introduction

Objective: To design a synthetic gene network that generates spatio-temporal patterning. We propose a new design using the repressilator and a quenching feedback loop.

Reaction-Diffusion Equation: $\frac{\partial c}{\partial t} = f(c) + D\nabla^2 c$, subject to zero-flux boundary condition.

Diffusion-Driven Instability: Stability of steady state c^* in reaction system does not imply stability of reaction-diffusion system.

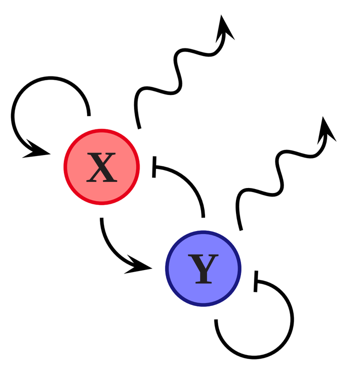
Conditions for Turing Pattern Formation

Here λ_k is the eigenvalue of Laplacian for wave number k and $J = \frac{\partial f}{\partial c}|_{c=c^*}$ is a linearization matrix.

1. Essential structural property for Turing phenomenon is an unstable subsystem.
2. This subsystem must be stabilized by the rest of the system so that J is stable.
3. The diffusion matrix D must be such that $J + \lambda_k D$ is unstable for some wave number $k \geq 1$.

Activator-Inhibitor Models

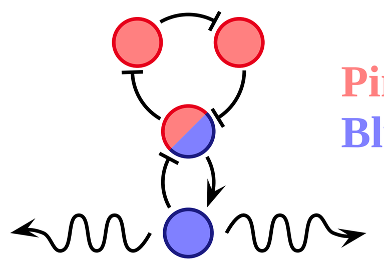
Canonical example proposed by Turing:



X: activator
Y: inhibitor

These have proven to be notoriously difficult to engineer!

New Quenched Oscillator Network



Pink: oscillator loop
Blue: quenching loop

Oscillator circuit now serves as the unstable subsystem.

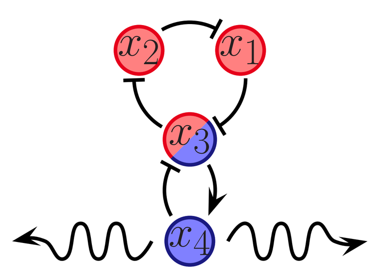
Advantages:

- Can leverage existing biological oscillators
- No diffusible species in the oscillator
- Do not have to worry about saturation regions

Toy Model

Use to show plausibility of Turing phenomenon:

$$\frac{\partial}{\partial t}x_1 = \frac{v_1}{1+x_2^p} - x_1$$
$$\frac{\partial}{\partial t}x_2 = \frac{v_2}{1+x_3^p} - x_2$$
$$\frac{\partial}{\partial t}x_3 = \frac{v_3}{1+x_1^p} + \frac{v_4}{1+x_4^p} - x_3$$
$$\frac{\partial}{\partial t}x_4 = \frac{v_5 x_3^p}{1+\alpha x_3^p} - x_4 + d_4\frac{\partial^2 x_4}{\partial \xi^2}$$



Oscillator loop (x_1, x_2, x_3) and quenching loop (x_3, x_4)

- Quenching loop must have smaller phase lag than the oscillator loop in order to stabilize the system

Toy Model Analysis

Jacobian linearization at steady state $(\bar{x}_1, \bar{x}_2, \bar{x}_3, \bar{x}_4)$:

$$J = \left[\begin{array}{ccc|c} -1 & -b_1 & 0 & 0 \\ 0 & -1 & -b_2 & 0 \\ -b_3 & 0 & -1 & -c_4 \\ \hline 0 & 0 & c_5 & -1 \end{array}\right], \quad J_{osc} = \begin{bmatrix} -1 & -b_1 & 0 \\ 0 & -1 & -b_2 \\ -b_3 & 0 & -1 \end{bmatrix}$$

$$b_1 = \frac{pv_1\bar{x}_2^{p-1}}{(1+\bar{x}_2^p)^2}, b_2 = \frac{pv_2\bar{x}_3^{p-1}}{(1+\bar{x}_3^p)^2}, b_3 = \frac{pv_3\bar{x}_1^{p-1}}{(1+\bar{x}_1^p)^2}, b_4 = \frac{pv_4\bar{x}_4^{p-1}}{(1+\bar{x}_4^p)^2}, c_5 = \frac{pv_5\bar{x}_3^{p-1}}{(1+\alpha\bar{x}_3^p)^2}.$$

Turing Condition #1: Oscillator loop is unstable

- $\det(\lambda I - J_{osc}) = (\lambda+1)^3 + b_1 b_2 b_3$
- Instability requires $B \triangleq b_1 b_2 b_3 > 8$

Turing Condition #2: Overall system J is stable

- $\det(\lambda I - J) = (\lambda+1)[(\lambda+1)^3 + B + c_4 c_5(\lambda+1)]$
- Stability requires $C \triangleq c_4 c_5 > (B-8)/2$

Turing Condition #3: Diffusion destabilizes some system modes

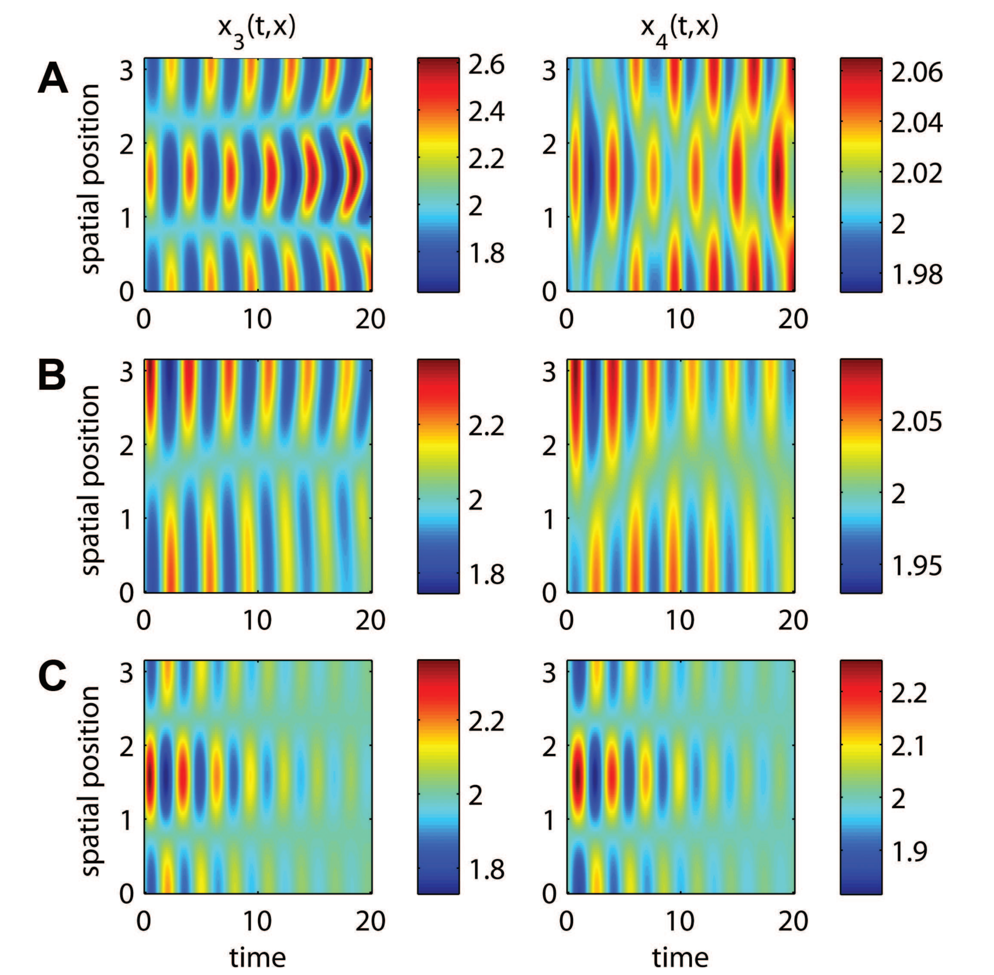
- Check stability of $J + \lambda_k D$
- For a 1-D spatial domain $\Omega = [0, L]$, $\lambda_k = -(k\frac{\pi}{L})^2$.
- Only x_4 diffuses, so $D = \text{diag}\{0, 0, 0, d_4\}$.
- $\det(\lambda I - J - \lambda_k D) = (\lambda+1)[(\lambda+1)^3 + B + C(\lambda+1)] - \lambda_k d_4[(\lambda+1)^3 + B]$

Note that if $\exists k^*$ such that $J + \lambda_{k^*}D$ is unstable, then $J + \lambda_k D$ is unstable $\forall k > k^*$

- $\exists$ a min wave number (or max wavelength) for instability

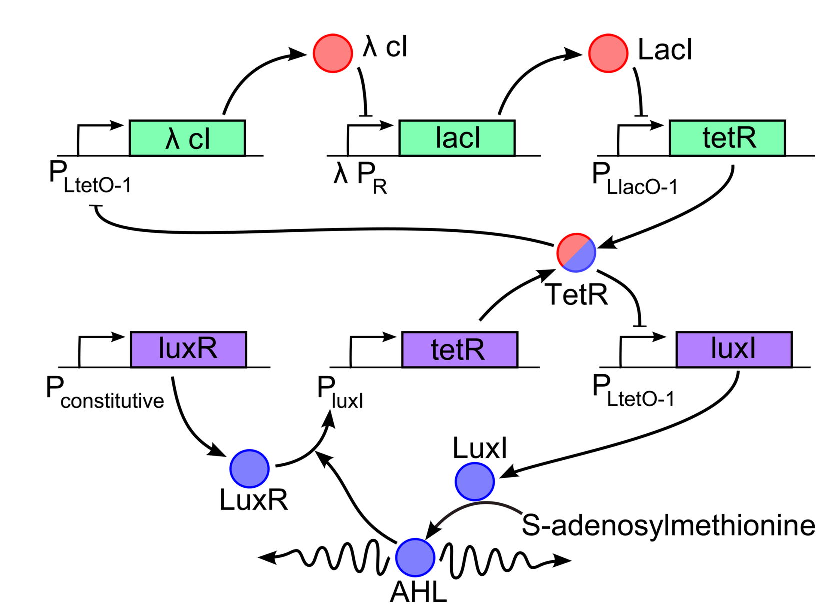
Toy Model Results

Parameters: $p = 3$, $v_1 = 18$, $v_2 = 18$, $v_3 = 9$, $v_4 = 9$, $v_5 = 0.45$, $\alpha = 0.1$



(A) For $k = 2$ and $d_4 = 6$, unstable mode grows.
(B) For $k = 1$ and $d_4 = 6$, stable mode decays.
(C) For $k = 2$ and $d_4 = 0$, all cells stabilize.

Proposed Synthetic Implementation



Pink: oscillator loop molecule
Green: oscillator loop gene
Blue: quenching loop molecule
Purple: quenching loop gene

This implementation based on the repressilator (Elowitz), but other oscillators possible.

Analysis follows same methodology. Model and parameter values not shown. Continuous, deterministic simulations show similar behavior to that of the toy model.

Stochastic Simulations

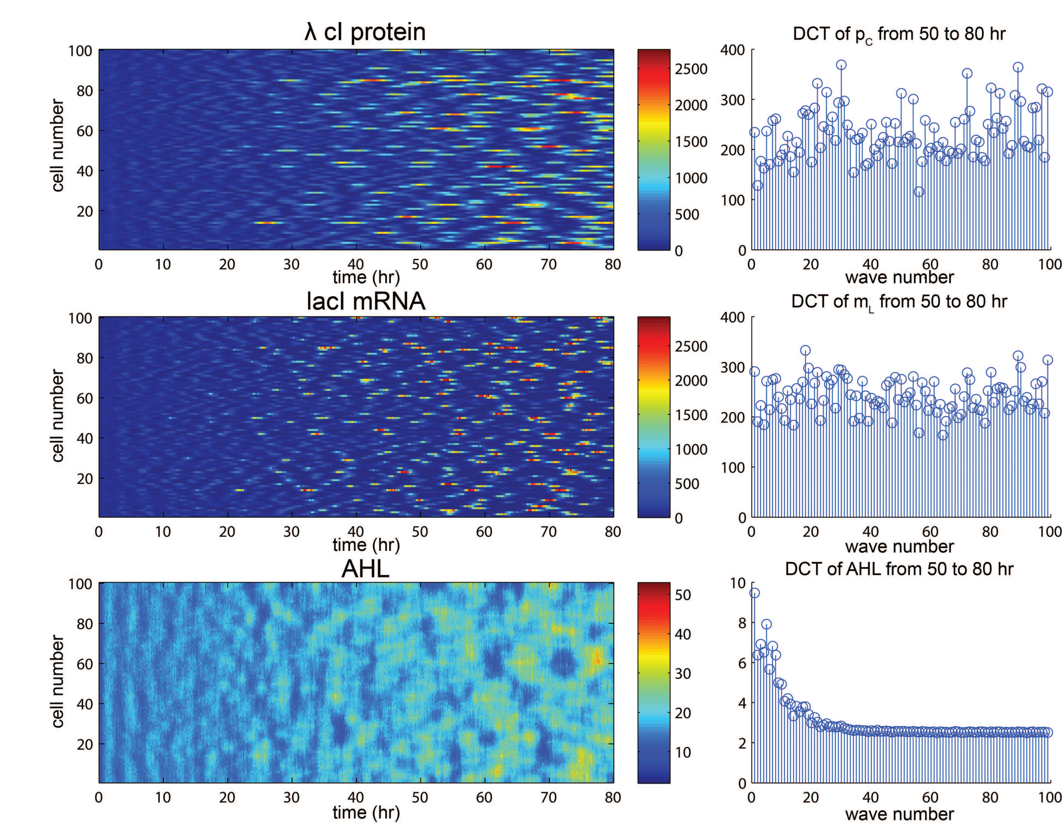
- Realistic parameters put many species concentrations in stochastic region.
- Better for testing experimental plausibility.

We use the **Stochastic Simulation Compiler (SSC)** developed at MIT (http://web.mit.edu/irc/ssc/).

- Allows for spatially heterogeneous systems.
- Claims to have an advantage in runtime as combinatorial complexity scales up.

Stochastic Results

Used a stochastic reaction set designed to match our PDE model. We generated the following plots using a physically possible parameter set.



Presence of patterning quantified using discrete cosine transform (DCT).

Patterning emerges spontaneously over time due to noise.

Future Work

- **Experimental implementation:** While our parameter set is physically possible, exact parts that match are not all currently known. As more biological parts are characterized or created, parts are likely to be found that match our chosen parameter values.
- **Other network topologies:** The repressilator is known to be difficult to implement, but worked well as a theoretical starting point. We will try different oscillators (Hasty's), quenching loops, and diffusible molecules.

Contact: Justin Hsia, 253 Cory Hall, Department of EECS, University of California, Berkeley, CA 94720; **E-mail:** *jhsia@eecs.berkeley.edu*
Acknowledgments: This work was supported in part by the National Science Foundation. Presented June 15-16, 2011 at SB5.0.