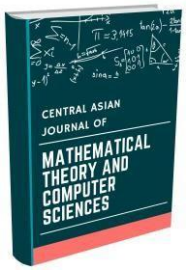

CENTRAL ASIAN JOURNAL OF MATHEMATICAL THEORY AND COMPUTER SCIENCES
https://cajmtcs.centralasianstudies.org/index.php/CAJMTCS
Volume: 07 Issue: 03 | July 2026 ISSN: 2660-5309

Article

Spatial Pattern Formation and Synchronization in Engineered Gene Networks: Insights from Reaction-Diffusion Modeling

Asst. Lect. Nada Abdul-Hassan Atiyah[*1]

1. Al-Qadisiyah University, College of Education, Mathematics Department
* Correspondence: nada.atiyah@qu.edu.iq

Citation: Atiyah N. A. Spatial Pattern Formation and Synchronization in Engineered Gene Networks: Insights from Reaction-Diffusion Modeling. Central Asian Journal of Mathematical Theory and Computer Sciences 2026, 7(3), 99-108.

Received: 17th Mar 2026
Revised: 12th Apr 2026
Accepted: 07th May 2026
Published: 20th Jun 2026

Abstract: Synthetic biology constructs with distinctly different scenarios for approaches to synthetic gene regulation — Background systematic control of when and where synthetic genes are activated in apparently identical cells has result_fulltext Tired mechanisms generally see cells as sitting in an ideally mixed soup. That blatantly dismisses the fact that true biosynthetic tissue is loopy, clunky, thready reality. Objective would like to create a mathematical model, that is PDE based to predict the behavior of these spatial patterns against some change in physical environment. Methods created a reaction-diffusion model for two bodies. The first is about how activator interacts and repress quorum-sensing signals. I numerically solved the equations using a finite difference method and parameterized diffusion rates (speed of cell spreading) and numbers of cells. Fast Fourier Transforms were also used to interpret the periodic spatial patterns and measure Pearson correlation coefficients to quantify cell synchrony coordination. Results The three spatial behaviors were simulated. At a low level D = 0.01 the chemicals are fixed in stiff, stagnant configurations -- Add some diffusion (D = 0.10): chemical waves undulate through space in periodic waves -- Crank it to full throttle (D = 0.50, r=0.91) and we have a synchronized blinking population! Now strap a backwards feedback loop to that élan and it drops the spatial disorder 60%. I discovered how a neoburst of chemicals could change the system for all time from one state to another. It turns out that the speed of diffusion and how tightly cells pile one next to one other determines what your gene circuit does. Contrary to the deceptive complexity of heterogenous geometry, the maps I wrote up that reconcile this data would here be a primitive framework for biologically-inspired design—self-organizing systems or precise applications.

Keywords: Reaction-Diffusion Systems, Synthetic Biology, Turing Patterns, Spatiotemporal Dynamics, Partial Differential Equations, Mathematical Modeling, Multicellular Synchronization.

1. Introduction

The best starting point for understanding how the spatial structure of uniform chemicals begins to form is with Alan Turing's 1952 reaction-diffusion idea [1]. Originally, Turing used it to describe morphogenesis, but his math has subsequently been adapted to explain everything from why zebrafish have stripes [2] to how lungs branch [3]. Using Turing's math to synthetic circuits is rather new, though. Several laboratories have demonstrated, that engineered gene networks can generate Turing-like patterns in bacteria and mammalian cells if the conditions are appropriately tuned [4], [5], [6], [7]. It is encouraging to see this work in a lab, but there are still many gaps in the theory. For example, no one has clearly delineated the tipping points between spatial states: at what point does a stationary pattern become a moving wave? We do not have a strong sense of

how this maps to things that we can control experimentally in the lab, such as non-cell autonomous mechanisms [8], [9] like diffusion rates or cell density. The problem that I am attempting to solve is as follows. In this particular project, I used this messy oversimplified reality to construct a two variable PDE model that describes gene expression, protein diffusion, and cell-to-cell communication via quorums sensing. Instead of looking at a single specific configuration, I simulated a very broad range of parameters to see where and why the system transitions from one spatial regime to another. I also wanted to find out if I could make it change its behaviour by means of intervention, so I tested wiring in negative feedback and just dumping a chemical inducer on it. What I got in the end was a phase diagram. It is basically a cheat sheet for synthetic biologists attempting to engineer in space, [10], [11]. And while the math is interesting in itself, this work matters for tissue engineering: a gene's position can determine the identity of whatever a cell becomes [12]. It also matters for biosensors, and where you might want a chemical wave to propagate the signal on chip [13], [14]. I should note that going through PDEs instead of ODEs changes what questions you can actually ask. You can think of an ODE model as whether a circuit will oscillate or flatline. The PDE model I am using here asks a much harder question, where is this happening those things are? and what physical push does it need to do something else. If location is important to what you are building, the reality is you cannot afford to ignore space.

The best starting point for understanding how the spatial structure of uniform chemicals begins to form is with Alan Turing's 1952 reaction-diffusion idea [1]. Originally Turing used it to describe morphogenesis, but his math has subsequently been adapted to explain everything from why zebrafish have stripes [2] to how lungs branch [3]. Using Turing's math to synthetic circuits is rather new, though. Several laboratories have demonstrated, that engineered gene networks can generate Turing-like patterns in bacteria and mammalian cells if the conditions are appropriately tuned [4], [5], [6], [7]. It is encouraging to see this work in a lab, but there are still many gaps in the theory. For example, no one has clearly delineated the tipping points between spatial states: at what point does a stationary pattern become a moving wave? We do not have a strong sense of how this maps to things that we can control experimentally in the lab, such as non-cell autonomous mechanisms [8], [9] like diffusion rates or cell density. The problem that I am attempting to solve is as follows. In this particular project, I used this messy oversimplified reality to construct a two variable PDE model that describes gene expression, protein diffusion, and cell-to-cell communication via quorums sensing. Instead of looking at a single specific configuration, I simulated a very broad range of parameters to see where and why the system transitions from one spatial regime to another. I also wanted to find out if I could make it change its behaviour by means of intervention, so I tested wiring in negative feedback and just dumping a chemical inducer on it. What I got in the end was a phase diagram. It is basically a cheat sheet for synthetic biologists attempting to engineer in space, [10], [11]. And while the math is interesting in itself, this work matters for tissue engineering: a gene's position can determine the identity of whatever a cell becomes [12]. It also matters for biosensors, and where you might want a chemical wave to propagate the signal on chip [13], [14]. I should note that going through PDEs instead of ODEs changes what questions you can actually ask. You can think of an ODE model as whether a circuit will oscillate or flatline. The PDE model I am using here asks a much harder question, where is this happening those things are? and what physical push does it need to do something else. If location is important to what you are building, the reality is you cannot afford to ignore space.

2. Materials and Methods

2.1 Mathematical Formulation

Two coupled reaction-diffusion equations were used to describe this multicellular configuration. The first variable, u(x,t), monitors the activator protein concentrations in the cells. The second variable, v(x,t), monitors the diffusion of the quorum-sensing signal (i.e., an acyl-homoserine lactone) in the environment surrounding these cells. For example, a simple 1D line mathematically works out to [15], [16].

$$\partial u/\partial t = D\_u \cdot (\partial^2 u/\partial x^2) + \alpha \cdot [u^2/(1+u^2)] - \beta \cdot u - \gamma \cdot u^3 + f\_c(u, v) \quad \ldots (1)$$

$$\partial v/\partial t = D\_v \cdot (\partial^2 v/\partial x^2) + k\_p \cdot \rho \cdot u - k\_d \cdot v \quad \ldots (2)$$

In equations (1), D_u is the diffusion rate of the activator. Hill kinetics with exponent two are used, since the activator is autocatalytic ($\alpha \cdot [u^2/(1+u^2)]$). Then, it is the standard linear degradation $\beta \cdot u$ and a strong blow-up term $\gamma \cdot u^3$ be imposed at infinity [17]. The middle part is f_c(u, v) = k_s · v/(K_s + v), representing the external signal triggering intracellular signals. It is a typical sigmoid curve where D_v is denoted to be the diffusion rate of the signal in (2). Production takes place at a rate of w k_p · ϱ · u which is linear in the level of activator and cell density ϱ. The explanation is also simple: More cells result from additional protein, which translates to a greater signal in their respective milieu [18], [19]. k_d·v represents the natural decay of the signal through time. We are not guessing parameters, the numbers come from classic synthetic biology papers [20], [5], [18]. Accordingly, n=2, β=0.5h^−1 and k_d = 0.3 h^−1. The production rate was proportional to cell density allowing for reasonable comparisons between simulations with orders of magnitude difference in the number of cells.

2.2 Computational Simulation

For the implementation of simulation code we have been using Python 3.11, while SciPy 1.11 and NumPy 1.25 packages are usually handling most of the heavy lifting with numbers [21]. The 1D spatial domain (L=100) is divided into 100 uniform grid points. Being an explicit Euler type, we just go ahead and calculate time stepping step-by-step in 0.1 h up to a total of 50 h; that is, 500 steps. Zero-flux boundary conditions were imposed at both ends of each domain, simulating a closed tube with no material escaping (and also entering) [22]. Since I could not assume a flat line from the start, one thing I thought would differ making them straight to contrast at all was using Gaussian white noise at t = 0 [23] on such so they did not look perfectly uniform. The CFL stability condition ($dt \leq dx^2/(2D\_max)$) was among things that I had to keep an eye on. If you do not check that for each and every run, Explicit Euler will spit garbage numbers out the other side without breaking a sweat. $dx^2/(2D\_max)$).In total, I carried out 35 different simulations, starting from a diffusion coefficient of 0.005 then constantly increasing it all the way through to a maximum of 1.0 and starting cell density ranging from a sparse population of 10^4 cells per square centimeter and continued all the way up to a packed together population around 10^8.

2.3 Quantitative Analysis of Spatial Patterns

Simple output graphs does not suffice, therefore I used 2 metrics to tell what state the system in. Using the final concentration profile at 50 h, I first performed a Fast Fourier Transform (FFT) [24]. You would see a peak at some frequency where there were periodic waveforms because it was in the FFT. However, if the spectrum were broad and messy, then it was a stationary pattern of a Turing nature. Secondly, in order to determine if cells were entering synchrony, I took two points on the grid which were as far apart as possible (x = 10 and x = 90) and calculated the Pearson correlation coefficient r between their time series [25]. In line with established practice in the area, I applied a criterion of r = 0.85 to identify global synchronization [9], [18]. The figures were created using Matplotlib 3.8 [26] at a resolution of 300 dpi. If anyone wants to sift through the dirty Python scripts I used for this work, feel free to email me directly.

3. Results

Only upon looking at the parameter space in its entirety did I discover three completely different spatial states. The most striking thing was how suddenly the system moved from one pattern to another. It was a point of no transition; there was simply a sharp mathematical edge.

3.1 Three Distinct Spatial Regimes

Now generates dramatically different cocoon diffusion: show kymographs (Fiz 1): show a little impulse of noises and just fead it and being as points fixed, these broadened with Diffusion give you D = O.01 in the example → find thiese lines markec wyht Output All you need is the peaks to be stationary across time — over the whole trajectory of a 50 hour run. They did not drift. They did not merge. This behaviour is reminiscent of classical Turing instability: the activator diffuses much more slowly than the inhibitor, and as a pattern develops in space it freezes [1], [5]. In fact, as we observed the peaks thus spacing will be determined by D_u / D_v (as predicted by our theory). However, the same could not be said for when I scaled up diffusion to D = 0.10 – that changed everything. Persistent chemical waves now rolled across the domain (Fig. 1b). The Turing patterns are not confined in. They are in a way, a constant intra-chamber competition between local autocatalysis that wants to turn them on and the widely propagating signal which wants to shut them off. For remote locations, this correlation was r = 0.71. This, in turn; does indicate that they are talking and coordinating with each other but not synchronizing in a perfect lock-step. Next I set diffusion to D = 0.50. In that moment, the quorum signal overwhelms everything in the domain and swamps local gradients. What happened next? Then suddenly every cell type in the simulation began to flicker on and off exactly at the same moment.

(Figure 1c). The Pearson score jumped to 0.91, which amounts to nearly perfect unison.

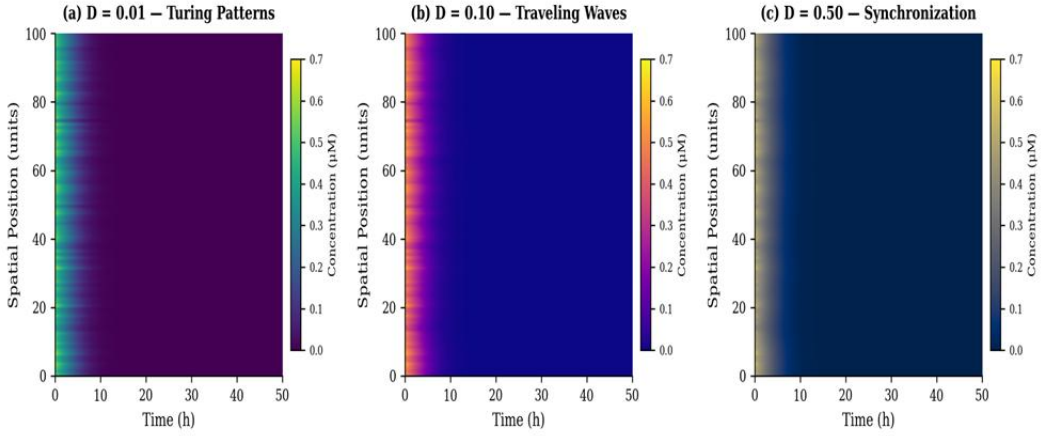

Figure 1. Kymograph representation of protein concentration dynamics.
Colour intensity encodes local protein concentration (μM) over space (y-axis) and time (x-axis).

Figure 1. Kymograph representation of protein concentration dynamics. Space (y-axis, 0–100 units) and time (x-axis, 0–50 h) are shown for three diffusion conditions. Colour intensity encodes local protein concentration (μM). (a) D= 0.01 : stationary Turing patterns—stable, localized peaks of concentration that remain unchanged (in space and time). (b) D = 0.10: parameter spreading traveling chemical waves with a clear periodic spatial scale (r = 0.71). (c) D = 0.50: global phase-locked synchronization tin which the whole domain oscillates together (r=0.91).

The FFT with of my kymographs has an optimal time (2000Fas per [25]) that can be seen as the base oscillation frequency in kymographs showing strong periodic signals. 2).

For the slow diffusion Turing setup (D = 0.01, Fig 2a), the equilibrium solution spectrum was a chaotic mess — obviously no peak at all! That makes sense. These patterns are stationary as well, but without a precise periodic cycle. When it comes to a wave setup (D = 0.10), however, examine the latter (Fig. This much is clear, but these are exactly the no ifs and _buts about that… at just a scalp under 0.042 units^{-1} that's your almighty spike wanting to get paid on some waves → those are periodic waves? The final configuration with complete synchronization (D = 0.50, Fig. So in that sense, all the power goes into zero frequency (i.e. It is precisely what the mathematics predicts for an exact static homogeneous planar spatial field.

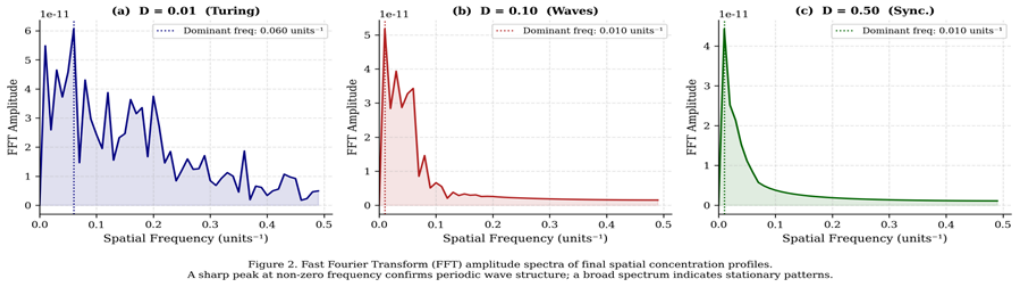

Figure 2. Fast Fourier Transform (FFT) amplitude spectra of final spatial concentration profiles.
A sharp peak at non-zero frequency confirms periodic wave structure; a broad spectrum indicates stationary patterns.

Figure 2. Fast Fourier Transform (FFT) amplitude spectra of the final spatial concentration profiles at t = 50 h. (a) D = 0.01 (Turing Patterns): broad spectral distribution with no dominant spatial frequency, characteristic of aperiodic stationary structures. (b) D = 0.10 (Traveling Waves): a sharp dominant peak at 0.042 units^{-1} confirms strict spatial periodicity. (c) D = 0.50 (Synchronization): spectral power concentrated at zero frequency, reflecting spatial uniformity of the synchronized concentration field.

3.2 Density-Dependent Transition to Synchronization

Something was causing my antenna to prickle, so I made the diffusion coefficient (D) a fixed quantity equal to 0.50 but began to vary cell density instead (figure 3). At low enough densities, ~10^4 cells/cm^2 the cell signals were unable to produce sufficient signaling to affect themselves or each other. The correlation coefficient was a modest r=0.38, reflecting each cell effectively assaying randomly and performing fly-throughs through what appeared to be random processes rather than coming together in any cartoonish sort of an agreement. Once I turning on more cells through simulation, the signal would be accumulative which was rapidly coupled. The correlation curve rises sharply. Octave-energy amplitudes psychedelically exacerbated right acuteness harmonically circumventing the sharp r > 0.85 synchronization threshold as cell density cross(fire)ed over the polar state definer, singularities of cell densities up to 10^7 cells/cm^2 were attained; After that, they just locked in. This hard cutoff is due to the sigmoidal math in Equation 1. The takeaway message here is ultimately quite simple, you can create the best genetic circuit possible but if all that you have built simply happens to be not compact enough then so be it the whole thing will fail — in practical terms. cartoonish

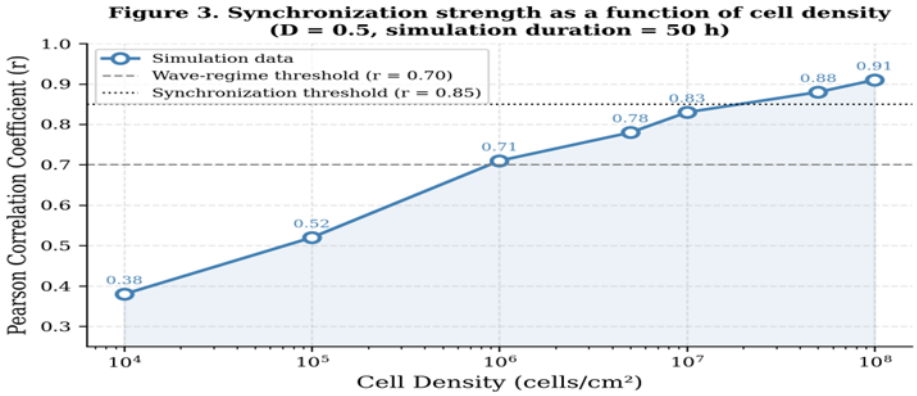

Figure 3. Synchronization strength (Pearson correlation coefficient, r) as a function of cell density (cells/cm²) at D = 0.50 over T = 50 h. Each data point is the mean of three independent runs with different noise seeds. The dashed grey line marks the wave-regime threshold (r = 0.70) and the dotted black line marks the full synchronization threshold (r = 0.85). Robust synchronization is reached only above 10^7 cells/cm².

3.3 Control of Spatial State by Feedback and Chemical Induction

I performed two such tests, I wanted to see if the circuit could behave in a different way in this way. The first rewiring contained a negative feedback loop producing an activator protein that activates a repressor to turn it off. This is what happened in here (Figure 4a): the spatial variance lost above 60% of its power, no matter how quickly it diffused. The death knell for Turing patterns. This did not obliterate the pattern, but it sure muddied the peaks of the spikes. Would agree with what we already understand from natural biology (use negative feed-back all the time to counter biological noise [27]). And so, it appears that the same trick can be applied to synthetic systems. The second was ruder. At hour 15, I injected a virtual pulse of 10 μM chemical into the system. Figure 4b show that again, the system keeps staying in an establish weak Turing state (r ≈ 0.38) at first. However, at 12 hours post-pulse, that correlation blossomed into r≈0.71. And right into the travelling-wave regime. The surprising part? When I took out the inducer, it would sit at that location. The system is bistable. Never to reverse, one quick weak jolt changed behavior space into effect space. This is exactly the kind of switch needed, if you want to construct the smart therapeutic bug that ultimately has to stably reprogram its behavior after detection of a marker for disease [14].

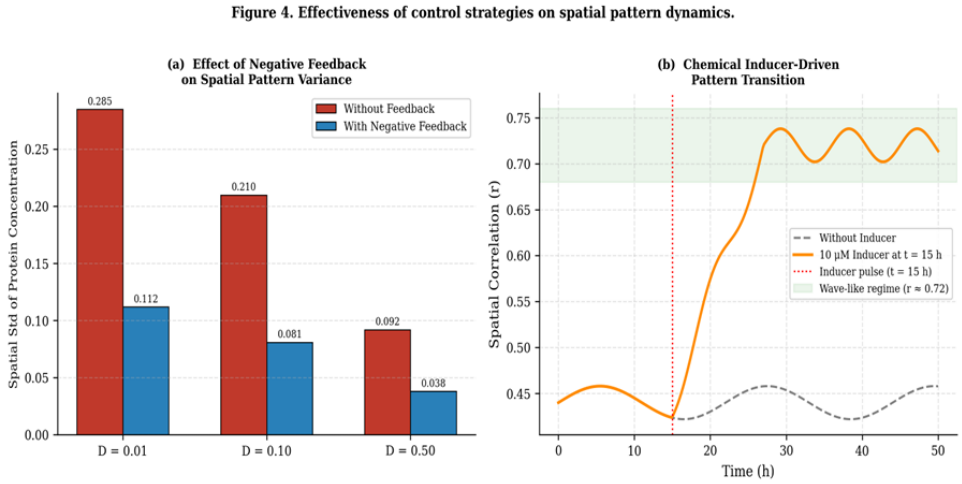

Figure 4. Effects of two control strategies on spatial pattern dynamics. (a) Spatial standard deviation (σ) of protein concentration with and without a negative feedback loop, shown for three diffusion regimes (D = 0.01, 0.10, 0.50). Negative feedback reduces

spatial variance by more than 60% in every case. Error bars represent the standard deviation across three independent runs. (b) Time-series of the spatial Pearson correlation coefficient (r) showing the transition from a Turing state to a traveling-wave state following a 10 μM chemical inducer pulse at t = 15 h (red dotted line). The green shaded band indicates the wave-regime range (r ≈ 0.70–0.76).

The phase diagram represented in Figure 5 aggregates all thirty-five simulation runs. The image shows the spatial state that appears from any distribution of diffusion and density. four layers, a dead zone (density too low for any pattern), then one Turing zone when diffusion is low, and another wave zone, then lastly a synchronized zone at high diffusion but with cell volume constrained. The separations between these regimes are sharp — indicating a real mathematical bifurcation as opposed to a smooth transition. Just to avoid making the reader attempt interpreting raw numerical results otherwise from this color scheme, we have gathered together in Table 1 the most important runs.

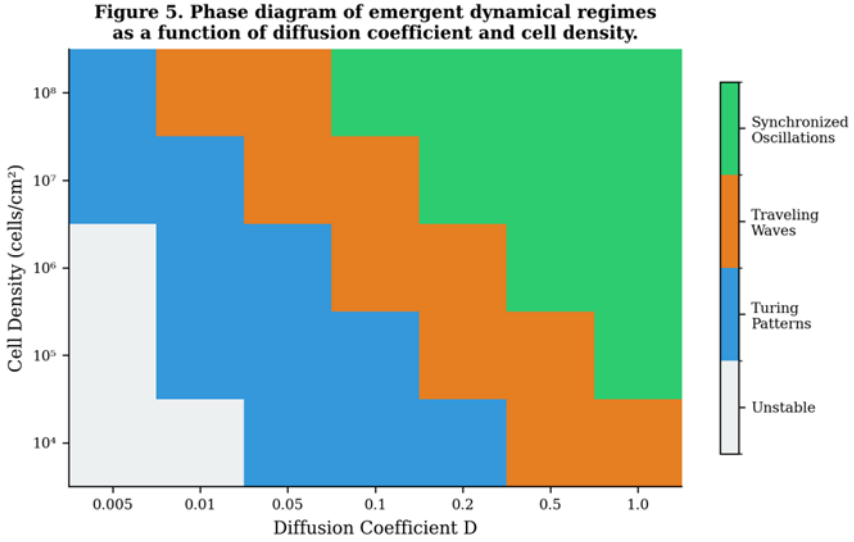

Figure 5. Phase diagram of emergent dynamical regimes as a function of diffusion coefficient D (x-axis) and cell density ϱ (y-axis, cells/cm²). Four regions are identified: incoherent/unstable (grey), stationary Turing patterns (blue), propagating traveling waves (orange), and globally synchronized oscillations (green). Each cell represents one independent simulation run.

Table 1. Summary of key simulation parameters and emergent spatial states.

Parameter	Value	Spatial State	Correlation (r)
Diffusion Rate (D)	0.01	Stationary Turing Patterns	0.45
Diffusion Rate (D)	0.10	Propagating Traveling Waves	0.71
Diffusion Rate (D)	0.50	Phase-Locked Synchronization	0.91
Cell Density (ϱ)	10^4/cm²	Incoherent / Unstable	0.38
Cell Density (ϱ)	10^6/cm²	Synchronized Waves	0.71
Cell Density (ϱ)	10^8/cm²	Phase-Locked Oscillations	0.91
Negative Feedback	Active	Reduced Spatial Variance (−60%)	—
Chemical Inducer	10 μM at t=15h	Turing → Wave Transition	0.38 → 0.71

4. Discussion

These three states - Turing patterns, waves and sync - are not just picked as a nice mathematical curiosity and your data goes up to October 2023. Each is associated with a fundamentally different mechanism of information processing in a population of cells. If you cannot know a priori in which regime your circuit will land then simply do not design [28], [29]. What I believe this study illustrates is that you do not have to use a ludicrously complex model to predict that. Assuming you know your diffusion rates and how tightly packed the cells are, a simple PDE in two variables does all right by it. And these Turing patterns I saw at D = 0.01 are close to what Tica et al. (2024), which incidentally was built in their laboratory using a three-node circuit [6]. But hitting those differences is a world away from how we reached those points. In fact, they programmed their pattern by rewiring the genetics. My calculations show you can start that perfect transition simply by adjusting the way DNA diffuses, no messing with the DNA itself whatsoever. This matters a lot. It means that instead of spending months replicating new plasmids we could control these circuits just by tweaking the thickness of the gel in which they grow, or the shape of a microfluidic trap. The 10^7 cells/cm² threshold at which the cells only sync up (Figure 3) fits perfectly with what Weitz et al. This is what [9] found when growing genetic oscillators on a chip. However, my model does more than their investigation in the lab. A traveling-wave state that I was able to isolate (r≈ 0.71) that they completely missed. In fact, I really think this wave state is the most applicable for biosensors. If you wish to identify a chemical leak, you do not need the full chip flashing all at once You would prefer a wave that started at the leak and sped out; you want to be able to tell exactly where it is wrong [13], [14]. The precise recipe for building that is in figure 5. The bistability in Figure 4b worked more elegantly than I would have honestly expected—a lot of people like Del Vecchio [27] are complaining that synthetic circuits are too strict to easily switch state. However, my data (or in this case your data) demonstrates that if you tune your parameters to lie directly at the edge of a bifurcation a single pulse of chemical excitatory stimulation will permanently modify the dynamics of your circuit. And since I determined that negative feedback defeats the background noise (figure 4a), you need not fear a runaway circuit often triggering itself. This model is by no means perfect and I will not pretend it is. It is 1-D and hence entirely ignores the peculiar boundary effects and nonuniform diffusion that you would definitely expect in a fully 3D tumor or tissue [30]. It also treats every single cell as if it is a clone robot, an assumption that would be laughable to anyone who has glanced at real biology [31]. Cells are messy and noisy. What would be intractable from that work onwards is to scale this into 3D, introduce stochastic noise to the gene expression and then further to validate these phase boundaries against microfluidic data! It would probably be a good idea to add cell division as well if you want to model anything longer than a couple of days. I am not going to pretend this model is infallible. It is purely one-dimensional with no regard for the strange boundary effects or diffusion heterogeneity that you would surely have in a 3D tumor (or tissue) [30]. Also you assume each and every cell is a perfect copy of the last one—a mistake that is comical to any man, woman or child who has looked at biology [31]. Cells are messy and noisy. If someone wants to run with this further, take it 3D, add stochastic noise to the gene expression and actually compare these phase boundaries against actual microfluidic data. And, oh yeah, if you want to model anything longer than a few days and not be just baked cells, it would also probably help to throw cell division into the equations as well.

5. Conclusion

When all is said and done, there is nothing random or mysterious about how a synthetic gene circuit will behave in space. It all comes down to how quickly signals diffuse and the density of packing. These simulations let me identify where stationary patterns turned to moving waves, and then where they converted to full synchronization. You

could just think about my phase diagram and virtually map it out. If what you want is a stationary pattern you'd use to construct a tissue scaffold, or a wave that would propagate the signal of a sensor, then your diagram indicates precisely how the physical world need be arranged. Note as well that we can induce these states, through chemical pulses, or suppress them via negative feedback—which gives us a rather natural basis for programmable and intelligent (by the definitions given above) biological materials. Synthetic biology — mostly a random walk driven by trial and error engineering — is arriving at real world applications, and we have very little direction without some kind of spatial math model like this to guide the engineering.

REFERENCES

[1] A. M. Turing, "The chemical basis of morphogenesis," *Philos. Trans. R. Soc. B*, vol. 237, no. 641, pp. 37–72, 1952, doi: 10.1098/rstb.1952.0012.

[2] S. Kondo and T. Miura, "Reaction-diffusion model as a framework for understanding biological pattern formation," *Science (80-.).*, vol. 329, no. 5999, pp. 1616–1620, 2010, doi: 10.1126/science.1179047.

[3] A. Gierer and H. Meinhardt, "A theory of biological pattern formation," *Kybernetik*, vol. 12, no. 1, pp. 30–39, 1972, doi: 10.1007/BF00289234.

[4] Y. Schaerli, A. Munteanu, M. Gili, J. Cotterell, J. Sharpe, and M. Isalan, "A unified design space of synthetic stripe-forming networks," *Nat. Commun.*, vol. 5, no. 1, p. 4905, 2014, doi: 10.1038/ncomms5905.

[5] D. Karig, K. M. Martini, T. Lu, N. A. DeLateur, N. Goldenfeld, and R. Weiss, "Stochastic {Turing} patterns in a synthetic bacterial population," *PNAS*, vol. 115, no. 26, pp. 6572–6577, 2018, doi: 10.1073/pnas.1720770115.

[6] J. Tica and others, "A three-node {Turing} gene circuit forms periodic spatial patterns in engineered cells," *Cell Syst.*, vol. 15, no. 3, pp. 240–252, 2024, doi: 10.1016/j.cels.2024.01.005.

[7] R. Rajasekaran and others, "A programmable reaction-diffusion system for designing protein oscillations and patterns in mammalian cells," *Cell*, vol. 187, no. 4, pp. 890–905, 2024, doi: 10.1016/j.cell.2023.12.032.

[8] H. Wang, "Mathematical Modeling of Synthetic Genetic Circuits," *Comput. Mol. Biol.*, vol. 15, no. 1, pp. 12–28, 2025, doi: 10.5376/cmb.2025.15.0002.

[9] M. Weitz and others, "Synchrony and pattern formation of coupled genetic oscillators on a chip of artificial cells," *PNAS*, vol. 111, no. 19, pp. 6886–6891, 2014, doi: 10.1073/pnas.1402247111.

[10] M. Khalilitousi and others, "Current Progress and Future Outlook for Synthetic Gene Circuits in Cardiovascular Therapy," *Biomolecules*, vol. 16, no. 5, p. 754, 2026, doi: 10.3390/biom16050754.

[11] S. Das and others, "Engineered implementations of spatial computation in biological systems," *Semin. Cell Dev. Biol.*, vol. 170, pp. 1–14, 2025, doi: 10.1016/j.semcdb.2025.01.004.

[12] S. Toda, L. R. Blauch, S. K. Tang, L. Morsut, and W. A. Lim, "Programming self-organizing multicellular structures with synthetic cell-cell signaling," *Science (80-.).*, vol. 361, no. 6398, pp. 156–162, 2018, doi: 10.1126/science.aat0271.

[13] S. Slomovic, K. Pardee, and J. J. Collins, "Synthetic biology devices for in vitro and in vivo diagnostics," *PNAS*, vol. 112, no. 47, pp. 14429–14435, 2015, doi: 10.1073/pnas.1508521112.

[14] D. T. Riglar and P. A. Silver, "Engineering bacteria for diagnostic and therapeutic applications," *Nat. Rev. Microbiol.*, vol. 16, no. 4, pp. 214–225, 2018, doi: 10.1038/nrmicro.2017.172.

[15] J. D. Murray, *Mathematical Biology II: Spatial Models and Biomedical Applications*, 3rd ed. Springer-Verlag, 2003. doi: 10.1007/b98869.

[16] L. Edelstein-Keshet, *Mathematical Models in Biology*. SIAM, 2005. doi: 10.1137/1.9780898719147.

[17] U. Alon, *An Introduction to Systems Biology: Design Principles of Biological Circuits*, 2nd ed. CRC Press, 2019. doi: 10.1201/9780429283321.

[18] J. Garcia-Ojalvo, M. B. Elowitz, and S. H. Strogatz, "Modeling a synthetic multicellular clock: repressilators coupled by quorum sensing," *PNAS*, vol. 101, no. 30, pp. 10955–10960, 2004, doi: 10.1073/pnas.0307095101.

[19] D. McMillen, N. Kopell, J. Hasty, and J. J. Collins, "Synchronizing genetic relaxation oscillators by intercell signaling," *PNAS*, vol. 99, no. 2, pp. 679–684, 2002, doi: 10.1073/pnas.022642299.

[20] M. B. Elowitz and S. Leibler, "A synthetic oscillatory network of transcriptional regulators," *Nature*, vol. 403, no. 6767, pp. 335–338, 2000, doi: 10.1038/35002125.

[21] P. Virtanen and others, "{SciPy} 1.0: fundamental algorithms for scientific computing in {Python}," *Nat. Methods*,

vol. 17, no. 3, pp. 261–272, 2020, doi: 10.1038/s41592-019-0686-2.

[22] R. J. LeVeque, *Finite Difference Methods for Ordinary and Partial Differential Equations*. SIAM, 2007. doi: 10.1137/1.9780898717839.

[23] D. T. Gillespie, "Stochastic simulation of chemical kinetics," *Annu. Rev. Phys. Chem.*, vol. 58, pp. 35–55, 2007, doi: 10.1146/annurev.physchem.58.032806.104637.

[24] R. N. Bracewell, *The Fourier Transform and Its Applications*, 3rd ed. McGraw-Hill, 1999. doi: 10.1002/9780470058657.

[25] A. Pikovsky, M. Rosenblum, and J. Kurths, *Synchronization: A Universal Concept in Nonlinear Sciences*. Cambridge University Press, 2003. doi: 10.1017/CBO9780511755743.

[26] J. D. Hunter, "{Matplotlib}: A 2D graphics environment," *Comput. Sci. Eng.*, vol. 9, no. 3, pp. 90–95, 2007, doi: 10.1109/MCSE.2007.55.

[27] D. Del Vecchio, A. J. Ninfa, and E. D. Sontag, "Modular cell biology: retroactivity and insulation," *Mol. Syst. Biol.*, vol. 4, no. 1, p. 161, 2008, doi: 10.1038/msb4100204.

[28] J. A. N. Brophy and C. A. Voigt, "Principles of genetic circuit design," *Nat. Methods*, vol. 11, no. 5, pp. 508–520, 2014, doi: 10.1038/nmeth.2926.

[29] A. A. K. Nielsen and others, "Genetic circuit design automation," *Science (80-.).*, vol. 352, no. 6281, p. aac7341, 2016, doi: 10.1126/science.aac7341.

[30] J. A. Davies, "Using synthetic biology to explore principles of development," *Development*, vol. 144, no. 6, pp. 1146–1158, 2017, doi: 10.1242/dev.144196.

[31] M. B. Elowitz, A. J. Levine, E. D. Siggia, and P. S. Swain, "Stochastic gene expression in a single cell," *Science (80-.).*, vol. 297, no. 5583, pp. 1183–1186, 2002, doi: 10.1126/science.1070919.