Simulated Cellular Evolution

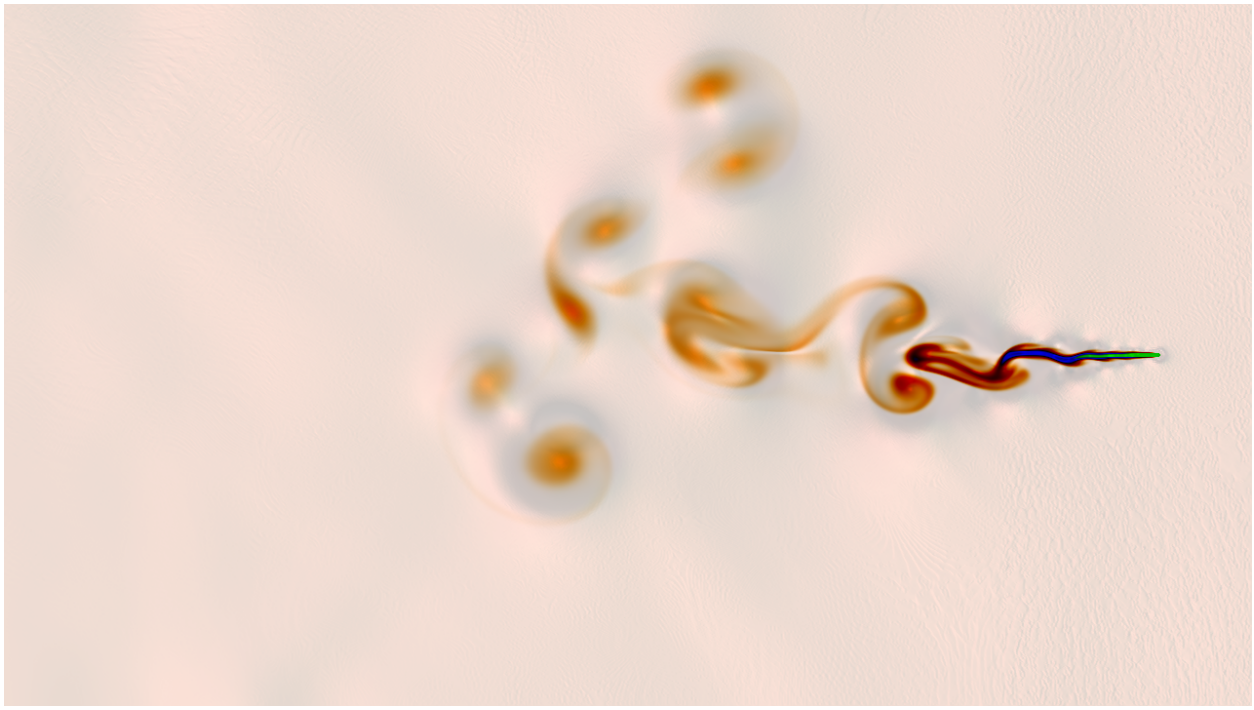
Wyatt Flanders

1. Fluid Dynamics
2. Cell Model
3. Cell-Fluid Interaction
4. Emergence & Evolution

Introduction

I have a lattice of numbers. The numbers can represent whatever I want. The numbers can communicate with each other however I program them. I want to see if I can simulate evolution with 11 values per pixel on a 2D lattice.

1. Fluid Model



If you learned that the Navier Stokes equations describe fluid dynamics, then we would live in a silent, cold world without sound or temperature and you would think that

solving the equations was difficult and complicated. Here we will use a simpler, more physical model of fluid dynamics based on the wave equation alone :

$$\frac{d^2}{dt^2}\psi = \frac{d^2}{dx^2}\psi$$

As you may know, we can separate this second order equation into two first order equations :

$$\frac{d}{dt}\psi' = -\frac{d}{dx}\psi$$

$$\frac{d}{dt}\psi = -\frac{d}{dx}\psi'$$

Now, allow our intuition to guide us, we will need **V***elocity* and **P***ressure*. Pressure is a scalar quantity of Energy/Volume. Velocity is a vector quantity of Distance/Time. Let us rewrite our wave equation with this in mind, converting to vector derivatives.

$$\frac{d}{dt}\hat{V} = -\hat{\nabla}P$$

$$\frac{d}{dt}P = -\hat{\nabla}\cdot\hat{V}$$

And finally, let us recall that velocity implies movement, and let us advect our wave through the velocity field. There are many ways to do this, but here let us simply linearly interpolate values between lattice points to move all 3 variables (Vx, Vy, and P) through the fluid. This is written simply as :

$$V,P(coord) = V,P\ (coord - dt * V)$$

Code it up in 3 stages :

Forever :

Each Pixel :

V_t *(coord)=* V_{t-dt} *(coord) - dt * gradient (*P_{t-1} *(coord))*

Each Pixel :

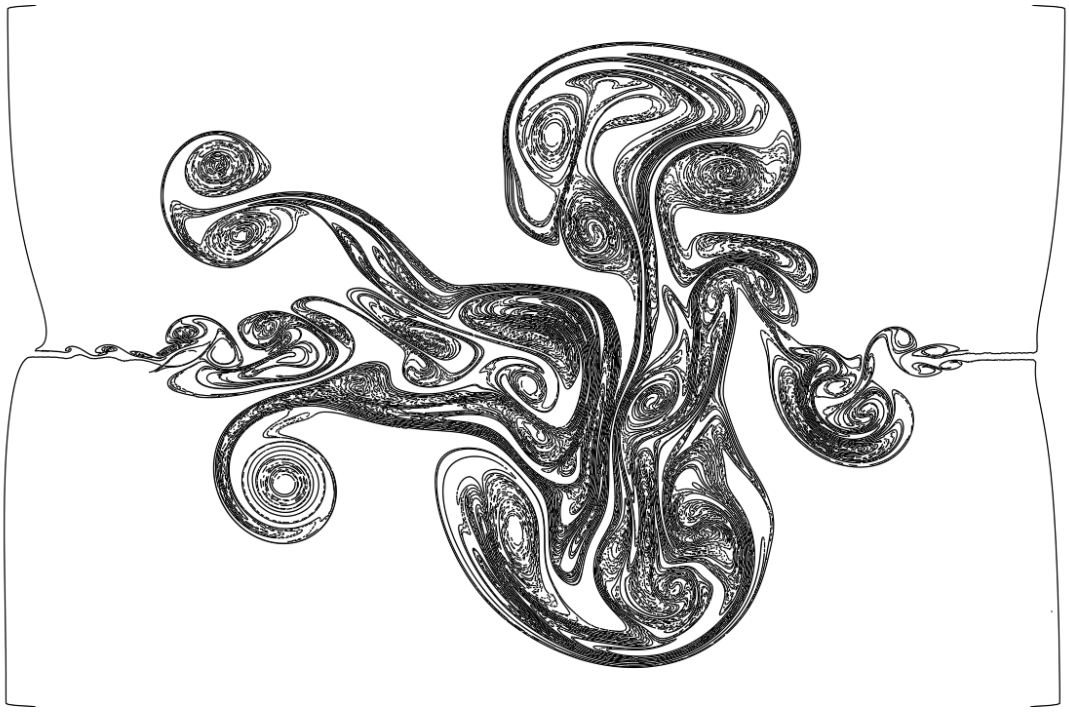
P_t *(coord) =* P_{t-dt} *(coord) - dt * divergence (*V_{t-1} *(coord))*

Each Pixel (linear interpolation on) :

P,V_t *=* P,V_{t-dt} *(coord - dt * V_{t-dt} *(coord))*

Now that we have our fluid model, we can submerge whatever we want. We can apply initial and boundary conditions in any manner we chose.

2. Cell Model



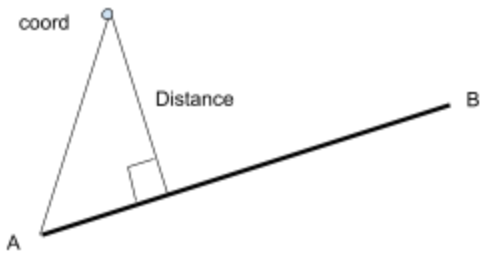
Let us assume that our cells are little line segments. They are very light and at the mercy of the movement of the fluid around them. When the line segment grows to an arbitrary length, it splits into two new cells joined at the midpoint.

Now we can introduce the next 4 variables on our lattice. Our line segment is represented by two points :

A_{xy} and B_{xy}. Each point, A and B, move independently. Each pixel in our lattice stores the closest line segment. Each pixel reads its

neighbors, continuously sorting for the closest line segment. The distance to a line segment is given by the formula :

$$Distance = \|coord - A - (B - A) * clamp(\frac{(coord-A)\cdot(B-A)}{(B-A)\cdot(B-A)}, 0, 1)\|$$

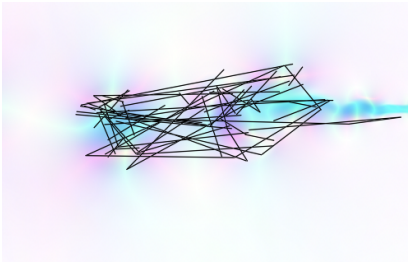


Now that we have each pixel tracking the nearest line, we can advect the lines through the fluid :

A_t (coord) = A_{t-dt} (coord) + dt * V (A_{t-dt} (coord))

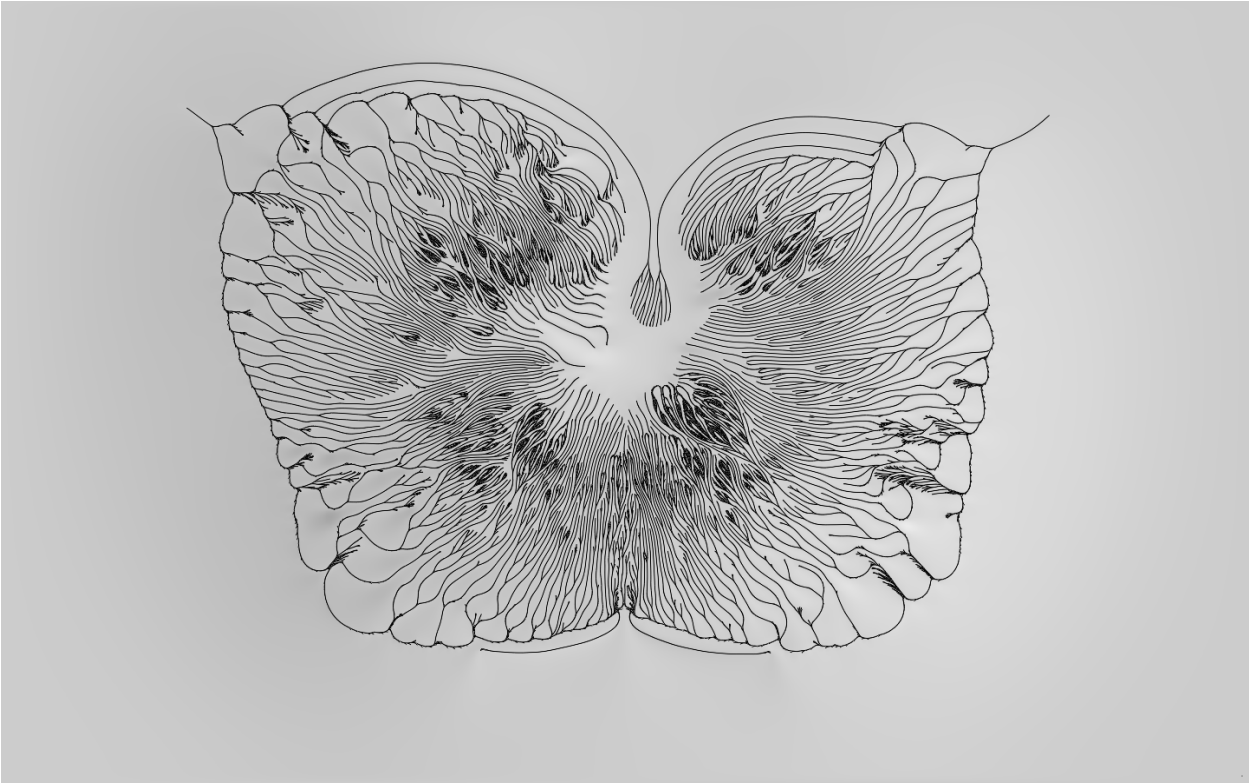
B_t (coord) = B_{t-dt} (coord) + dt * V (B_{t-dt} (coord))

Here, we are simply translating the coordinate of each point by the velocity of the fluid at its location. Over time, A and B will be separated due to the chaos of the fluid. Evidently, it is time to split the lines in half. This is a achieved with a simple ‘if’ statement :



IF || a - b || < max_length THEN b = (a+b)/2

3. Cell-Fluid Interaction



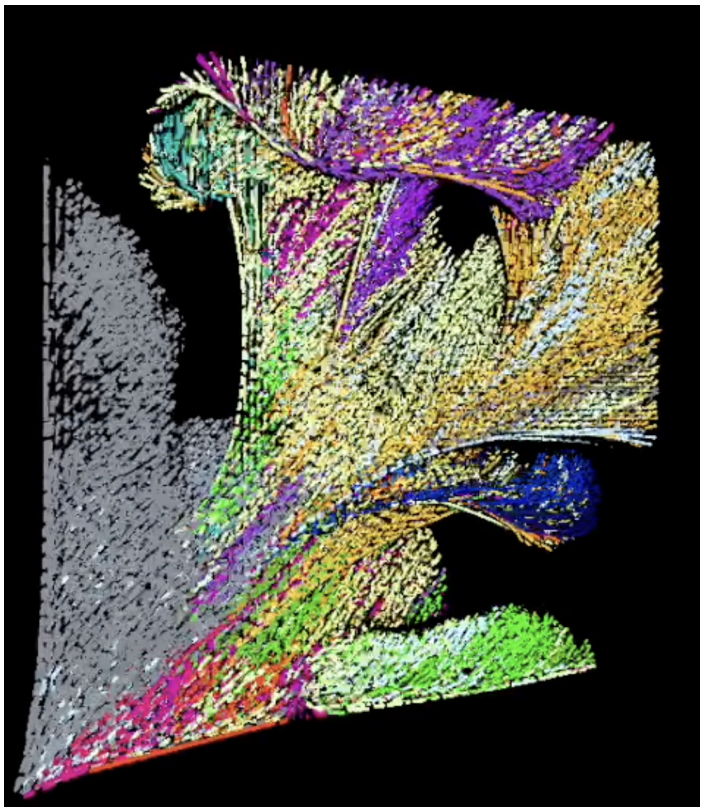
Now, let us bring our cells to life by giving them some metabolic activity and a little curiosity.

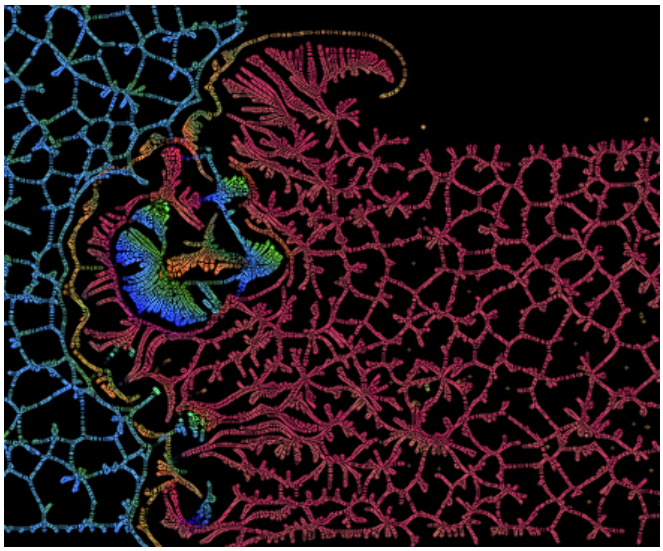
We can achieve this by using our line segments as boundary conditions in our fluid. We can put a force on our fluid in the direction A→B and we can add or remove material from the fluid by putting a force on the Pressure variable.

$$V_t(coord) = V_{t-dt}(coord) + cell_speed * \delta(distance(coord, A, B))$$

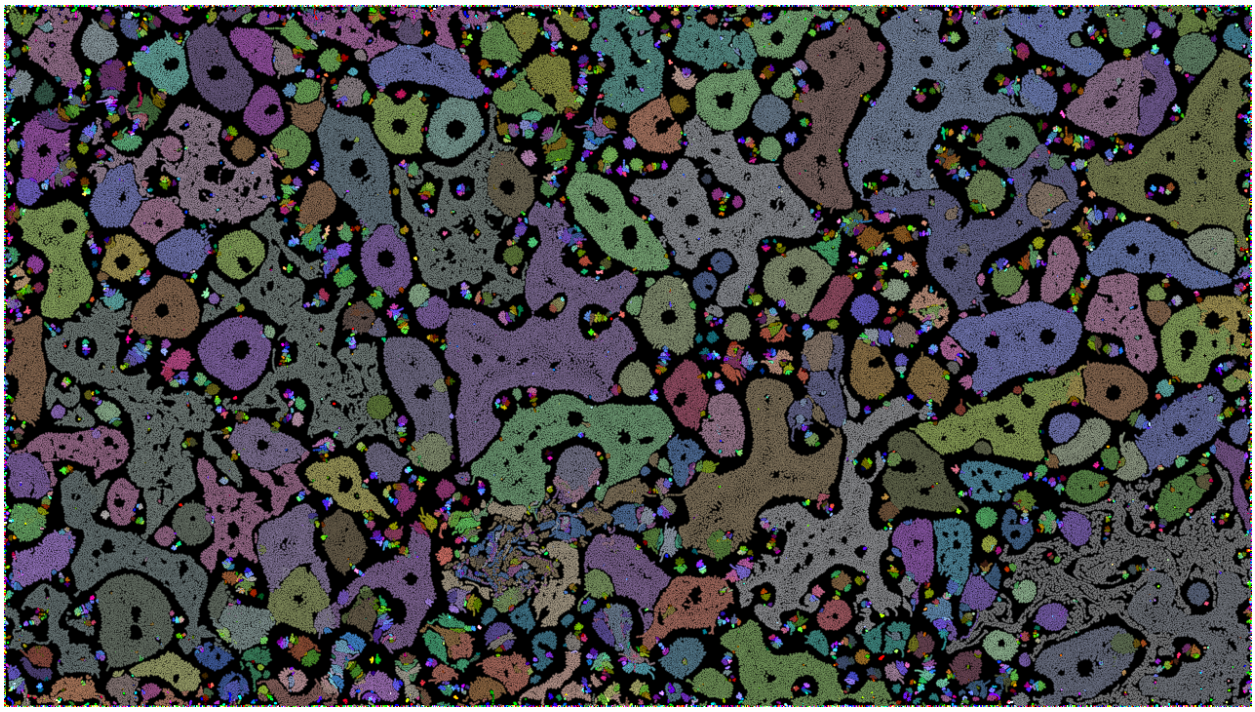
$$P_t(coord) = P_{t-dt}(coord) + cell_metabolism * \delta(distance(coord, A, B))$$

Now that the cells can move around and push or pull on the fluid around them, we can also introduce some imperfection to the advection by making it so that each pixel does not agree about how exactly A and B moved. This is achieved simply through sloppy coding and was a happy accident that brought this project to life.



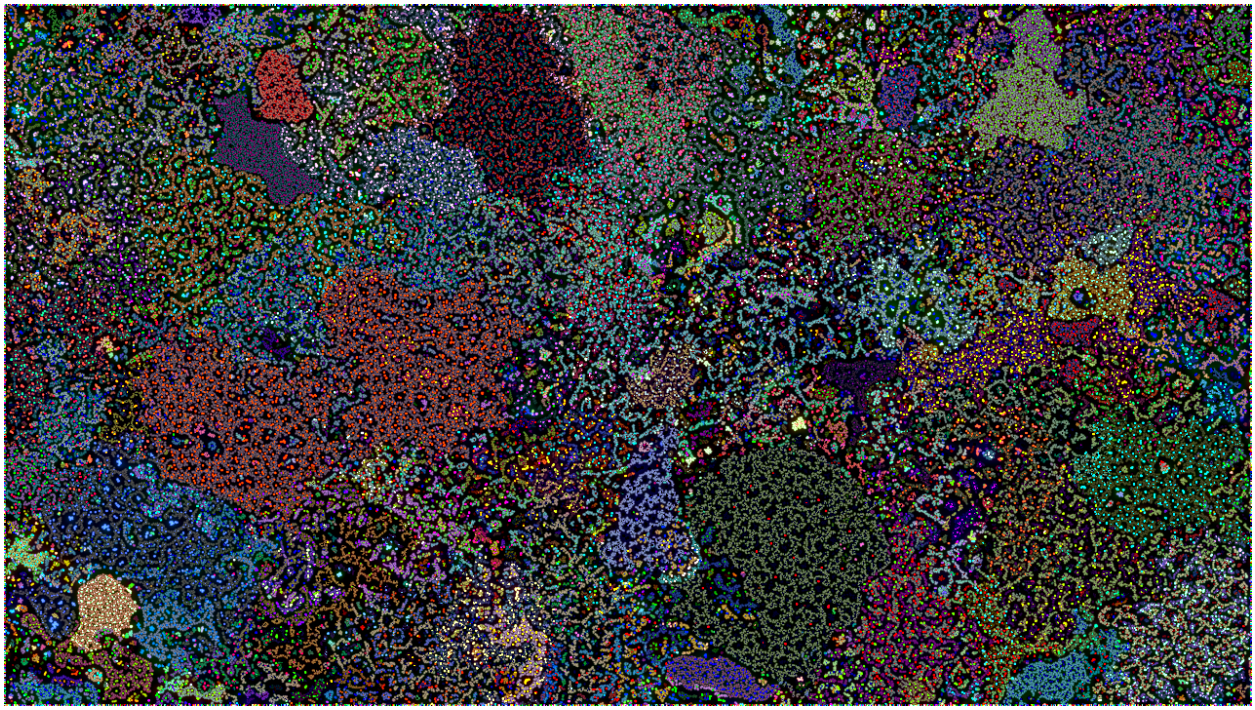


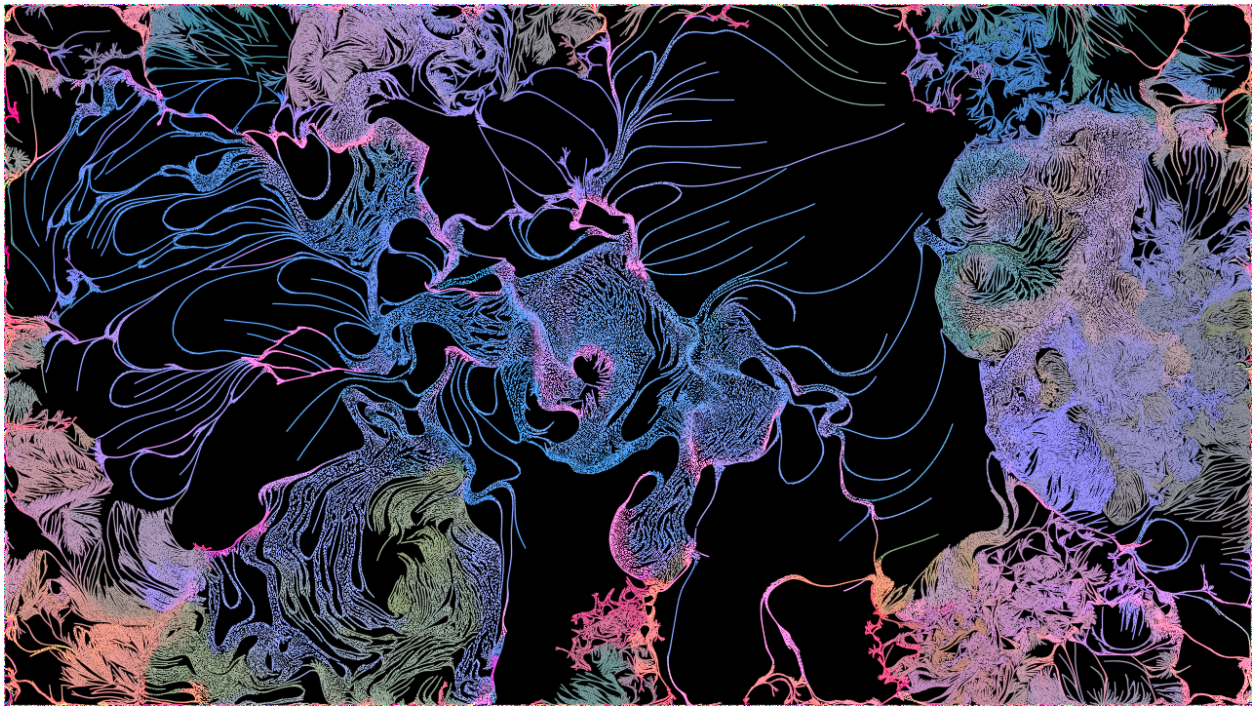
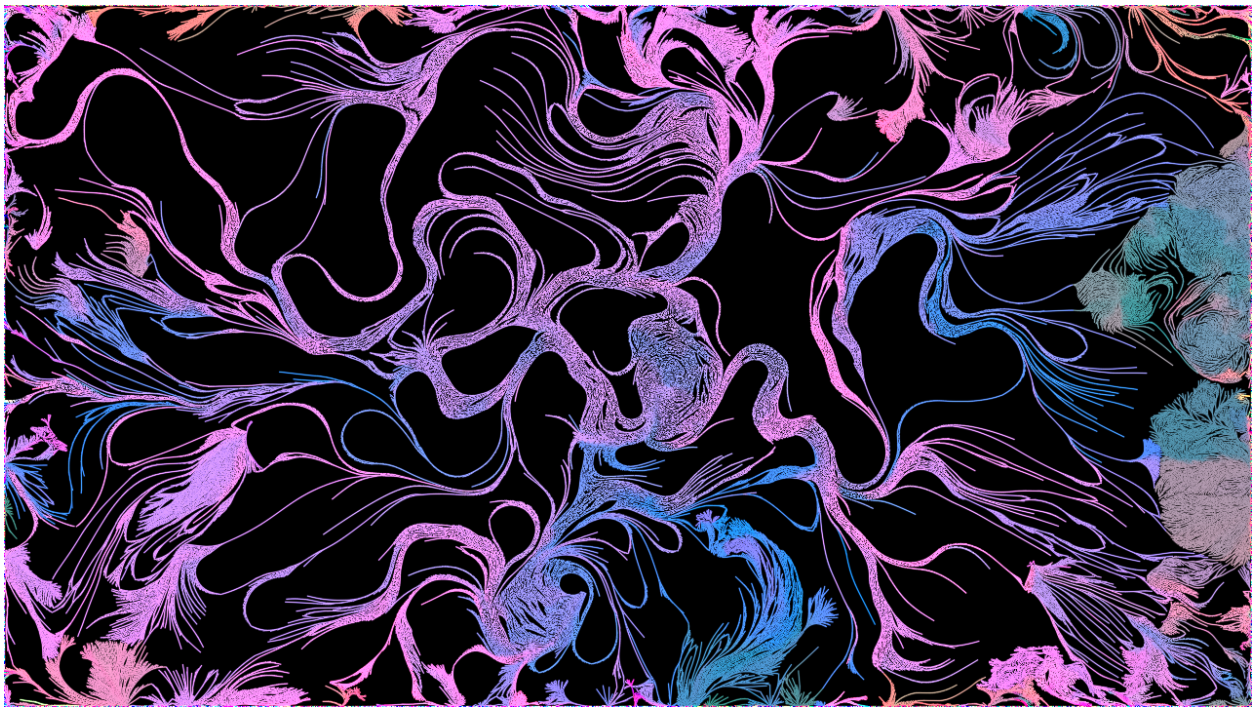
4. Emergence & Evolution



Now that we have some parameters, we can introduce 4 more variables to our lattice. This time, the values represent the cell's strategy. We can make the strategy correspond to any parameter in the cell's interaction with its surroundings. We have a lot of room for emergence - there's no telling what will happen given a parameter set.

We can simulate natural selection by initializing a random set of strategies. The best strategies will dominate the lattice and take over the finite world.



We can also simulate evolution by diffusing the strategy. This acts like sexual reproduction and allows the dominant species to have a strategy that did not necessarily exist at the beginning of the simulation. Try it yourself! It's a new story every time you restart the simulation.

https://www.shadertoy.com/view/wttfDs