

One-Dimensional Reaction-Diffusion Equations: Numerical Solution and Algorithm

Garrett Higginbotham*

Department of Physics

The University of Alabama at Birmingham

(Dated: June 7, 2018)

Abstract

We consider a one-dimensional reaction diffusion process consisting of three particle types. Particles A and B may freely diffuse throughout the volume of the system, as well as diffuse onto the right-most boundary. Of those particles that diffuse onto the boundary, particles of type A and B may react to form particle C . Particle C is restricted to remain on the boundary, and it is allowed to disassociate back into A and B .

MODEL

Establish a coordinate system so that the left-edge of the volume is labeled with $x = 0$ and the right-edge is labeled with $x = L$. Within the volume, particles A and B are allowed to diffuse according to the standard diffusion equation:

$$\frac{\partial n_A(x, t)}{\partial t} = D_A \frac{\partial^2 n_A(x, t)}{\partial x^2} \quad (1a)$$

$$\frac{\partial n_B(x, t)}{\partial t} = D_B \frac{\partial^2 n_B(x, t)}{\partial x^2} \quad (1b)$$

where $n_l(x, t)$ denotes the concentration of particle l at space-time position (x, t) , and D_l represents the corresponding diffusion constant (we consider the particles to not be interacting with each other, and we may thus safely assume the diffusion constant to actually be constant). We assume that the $x = 0$ boundary enforces zero flux of the particles, which may be expressed as

$$\left. \frac{\partial n_A(x, t)}{\partial x} \right|_{x=0} = 0 \quad (2a)$$

$$\left. \frac{\partial n_B(x, t)}{\partial x} \right|_{x=0} = 0 \quad (2b)$$

Let $\sigma_l(t)$ denote the concentration of particle l once it has diffused onto the $x = L$ boundary, and let k_{on}^l and k_{off}^l denote the association and disassociation rates of particle l on and off the membrane, respectively. Finally, we assume that particles A and B bind together in equal proportions to create particle C with reaction rate k_{bind} , and that particle C may then disassociate back into A and B with rate k_{dis} . The concentration of particles diffused onto the $x = L$ boundary may then be expressed as

$$\frac{d\sigma_A(t)}{dt} = k_{on}^A n_A(L, t) - k_{off}^A \sigma_A(t) - k_{bind} \sigma_A(t) \sigma_B(t) + k_{dis} \sigma_C(t) \quad (3a)$$

$$\frac{d\sigma_B(t)}{dt} = k_{on}^B n_B(L, t) - k_{off}^B \sigma_B(t) - k_{bind} \sigma_A(t) \sigma_B(t) + k_{dis} \sigma_C(t) \quad (3b)$$

$$\frac{d\sigma_C(t)}{dt} = k_{bind} \sigma_A(t) \sigma_B(t) - k_{dis} \sigma_C(t) \quad (3c)$$

Finally, the flux at $x = L$ may be expressed as

$$D_A \left. \frac{\partial n_A(x, t)}{\partial x} \right|_{x=L} = -k_{on}^A n_A(L, t) + k_{off}^A \sigma_A(t) \quad (4a)$$

$$D_B \left. \frac{\partial n_B(x, t)}{\partial x} \right|_{x=L} = -k_{on}^B n_B(L, t) + k_{off}^B \sigma_B(t) \quad (4b)$$

In this model, we assume that particle C is only allowed to exist at $x = L$.

Talk about FRAPNO DON'T TALK ABOUT FRAP IT NEEDS ANOTHER PAPER BY ITSELF

STEADY-STATE SOLUTIONS

We will now consider the steady states of the system. For the steady-state solutions, we must have

$$\frac{d\sigma_A}{dt} = \frac{d\sigma_B}{dt} = \frac{d\sigma_C}{dt} = 0.$$

Substituting these into equation 3, we obtain the following steady-state ratios:

$$\frac{k_{on}^A}{k_{off}^A} = \frac{\sigma_A(t^*)}{n_A(L, t^*)} \quad (5a)$$

$$\frac{k_{on}^B}{k_{off}^B} = \frac{\sigma_B(t^*)}{n_B(L, t^*)} \quad (5b)$$

$$\frac{k_{bind}}{k_{dis}} = \frac{\sigma_C(t^*)}{\sigma_A(t^*)\sigma_B(t^*)} \quad (5c)$$

NUMERICAL PARADIGM

In order to simulate this system of equations, we will begin by slicing space-time in a grid-like manner. Let Δx and Δt denote the lengths of the spatial and temporal slices, respectively, and denote the coordinates by $x_i = i\Delta x, i \geq 0$ and $t_j = j\Delta t, j \geq 0$. Let $x_N = L$, so that we then have $\Delta x = \frac{L}{N}$. At each coordinate of space-time, we may associate a value for the particle concentrations, and we denote these values by $n_l^{i,j} = n_l(x_i, t_j)$ and $\sigma_l^j = \sigma_l(t_j)$. Because we have discretized the space, we may no longer use calculus. However, if we choose Δx and Δt correctly (the appropriate choice will be discussed later), we may approximate the derivatives in our system via finite differences.

Using finite differences, equation 1 may be expressed as

$$n_l^{i,j+1} = n_l^{i,j} + \frac{D_l \Delta t}{(\Delta x)^2} [n_l^{i+1,j} - 2n_l^{i,j} + n_l^{i-1,j}], 0 < i < N \quad (6)$$

Note (importantly) that this is inadequate for computing the values $n_l^{0,j}$ and $n_l^{N,j}$. We may compute $n_l^{0,j}$ via the discretized version of equation 2:

$$n_l^{0,j} = n_l^{1,j} \quad (7)$$

as well as compute $n_l^{N,j}$ from the discretized version of equation 4:

$$n_l^{N,j} = \frac{D_l}{D_l + k_{kon}^l \Delta x} n_l^{N-1,j} + \frac{k_{off}^l \Delta x}{D_l + k_{on}^l \Delta x} \sigma_l^j \quad (8)$$

Finally, we may compute the time evolution of σ_l from the discretized version of equation 3. For particles A and B , we have

$$\sigma_l^{j+1} = (1 - k_{off}^l \Delta t) \sigma_l^j + \Delta t \left[k_{on}^l n_l^{N,j} - k_{bind} \sigma_A^j \sigma_B^j + k_{dis} \sigma_C^j \right] \quad (9)$$

and for particle C we have

$$\sigma_C^{j+1} = (1 - k_{dis} \Delta t) \sigma_C^j + k_{bind} \Delta t \sigma_A^j \sigma_B^j \quad (10)$$

We are now ready to develop an algorithm to compute everything.

NUMERICAL ACCURACY

In free diffusion, the variance in position is related to time via $\langle x^2 \rangle = 2Dt$. Thus, the time step Δt must be taken to be small enough to ensure that adequate information is transferred in the spatial direction: $\Delta t << \frac{\Delta x^2}{2D}$.

Since we have two distinct particles, we must have

$$\Delta t << \min_{l=A,B} \left\{ \frac{\Delta x^2}{2D_l} \right\}$$

. From equations 9 and 10, we also see that we must have $\Delta t \gg 0$ and $\Delta t << \min_{l=A,B} \left\{ \frac{1}{k_{on}^l}, \frac{1}{k_{bind}}, \frac{1}{k_{dis}} \right\}$. Putting all these together, we obtain the convergence condition

$$0 << \Delta t << \min_{l=A,B} \left\{ \frac{\Delta x^2}{2D_l}, \frac{1}{k_{on}^l}, \frac{1}{k_{bind}}, \frac{1}{k_{dis}} \right\} \quad (11)$$

Have I made a mistake or overlooked something? It seems to me that k_{off}^l doesn't come into play with the convergence condition.

ALGORITHM

1. Choose the constants of the system.

- Be sure to set D_l , k_{on}^l , k_{off}^l , k_{bind} , and k_{dis} .

- Set the length L of the system, and set the number of spatial slices N . Then, we may take $\Delta x = \frac{L}{N}$.
- Set the time limit T of the system, and set the number of temporal slices M . Then, we may take $\Delta t = \frac{T}{M}$.

2. Initialize the concentrations of the system.

- For the volume concentrations, set $n_l^{i,0} = \tilde{n}_l^i$; i.e., functions only of position. Do this only for $0 < i < N$.
- For the surface concentrations, set $\sigma_l^0 = \sigma_l$; i.e., constant values.
- Set $\tilde{n}_l^1 = \tilde{n}_l^2$, due to equation 7.
- Use equation 8 to compute \tilde{n}_l^N .

3. We may now calculate the time evolution of the system. Assume without loss of generality that we are currently at time step j .

- Use equation 6 to compute $n_l^{i,j+1}$ for $0 < i < N$.
- In accordance with equation 7, set $n_l^{1,j+1} = n_l^{2,j+1}$.
- Use equations 9 and 10 to compute σ_l^{j+1} .
- Use equation 8 to compute $n_l^{N,j+1}$.

INTERMEDIATE QUESTIONS

1. What conservation laws could we use to test our program? In the case where we neglect reactions, we must have conservation of mass. But that isn't the case for a reaction simulation. Do conservation laws for this system even exist? Maybe we could verify some thermodynamic laws.

MATLAB SCRIPT

RESULTS

* ghiggie@uab.edu