FYS4150 - Computational Physics - Project 5

Candidate number: 68
Email: eimundsm@fys.uio.no
28th November, 2014

Abstract

1 Diffusion of neurotransmitters

I will study diffusion as a transport process for neurotransmitters across synaptic cleft separating the cell membrane of two neurons, for more detail see [1]. The diffusion equation is the partial differential equation

$$\frac{\partial u(\mathbf{x},t)}{\partial t} = \nabla \cdot \left(D(\mathbf{x},t)\, \boldsymbol{\nabla} u(\mathbf{x},t) \right),$$

where u is the concentration of particular neurotransmitters at location $\mathbf{x}$ and time t with the diffusion coefficient D. In this study I consider the diffusion coefficient as constant, which simplify the diffusion equation to the heat equation

$$\frac{\partial u(\mathbf{x},t)}{\partial t} = D\nabla^2 u(\mathbf{x},t) \ .$$

I will look at the concentration of neurotransmitter u in two dimensions with x_1 parallel with the direction between the presynaptic to the postsynaptic across the synaptic cleft, and x_2 is parallel with both presynaptic to the postsynaptic. Hence we have the differential equation

$$\frac{\partial u\left(\{x_i\}_{i=1}^2, t\right)}{\partial t} = D \sum_{j=1}^{2} \frac{\partial^2 u\left(\{x_i\}_{i=1}^2, t\right)}{\partial {x_j}^2}, \tag{1}$$

where $\{x_i\}_{i=1}^2 = (x_1, x_2) = \mathbf{x}$. The boundary and initial condition that I'm going to study is

$$\begin{aligned} \exists \{d, w\} \subseteq \mathbb{R}_{0+} \exists \{w_i\}_{i=1}^2 \subseteq \mathbb{R}_{0+}^{w-} \Big(& \forall t \in \mathbb{R}_0 : \forall x_2 \in \mathbb{R}_{w_1}^{w_2} : u(0, x_2, t) = u_0 \\ & \wedge \forall t \in \mathbb{R} \Big(\forall x_2 \in \mathbb{R}_{0+}^{w-} : u(d, x_2, t) = 0 \wedge \forall x_1 \in \mathbb{R}_0^d : \Big(u(x_1, 0, t) = 0 \wedge u(x_1, w, t) = 0 \Big) \Big) \\ & \wedge \forall x_1 \in \mathbb{R}_{0+}^{d-} \forall x_2 \in \mathbb{R}_{0+}^{w-} : u\left(\{x_i\}_{i=1}^2, 0\right) = 0 \wedge \forall x_2 \in \mathbb{R}_0^w \setminus \mathbb{R}_{w_1}^{w_2} : u(0, x_2, 0) = 0 \Big) \end{aligned} \tag{2}$$

where d is the distance between the presynaptic and the postsynaptic, and w is the width of the presynaptic and postsynaptic. Note that the notation $\forall x \in \mathbb{R}_{a+}^{b-} \Leftrightarrow a < x < b$, where as $\forall x \in \mathbb{R}_a^b \Leftrightarrow a \leq x \leq b$. Note also that these boundary conditions implies that the neurotransmitters are transmitted from presynaptic at $x_1 = 0$ and $w_1 \leq x_2 \leq w_2$ with constant concentration u_0; the neurotransmitters are immediately absorbed at the postsynaptic $x_1 = d$; there are no neurotransmitters at boundary width $x_2 = 0$ and $x_2 = w$ of the synaptic cleft; and we have the initial condition at $t = 0$ where there are no neurotransmitters between the pre- and postsynaptic as well on the side of the

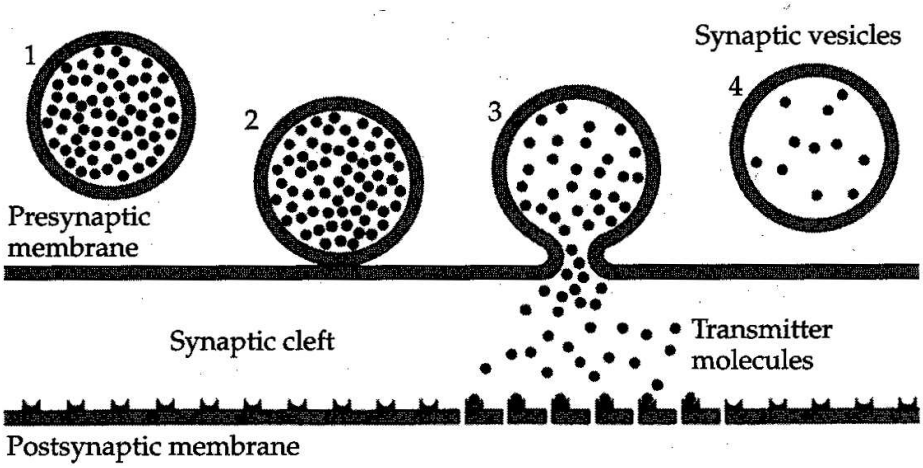

Figure 1.1: *Left: Schematic drawing of the process of vesicle release from the axon terminal and release of transmitter molecules into the synaptic cleft. (From Thompson: "The Brain", Worth Publ., 2000). Right: Molecular structure of the two important neurotransmitters glutamate and GABA.*

synaptic vesicles $x_1 = 0, 0 \leq x_2 < w_1$ and $w_2 < x_2 \leq w$.

To solve the differential equation (1) with the boundary and initial condition (2) we make an ansatz that the solution is unique, which is the case for a deterministic system. We recognize the heat equation as part of the class of partial differential equation spanned by the Poisson's equation for each time instance. The Uniqueness theorem for the Poisson's equation $\nabla^2 u = f$ [8] says that the Poisson's equation has a unique solution with the Dirichlet boundary condition, where Dirichlet boundary condition is here defined as a boundary that specifies the values the solution must have at the boundary.

Unfortunately the boundary condition in (2) is not a Dirichlet boundary, since the boundary is not specified on the side of the synaptic vesicles $x_1 = 0, 0 \leq x_2 < w_1$ and $w_2 < x_2 \leq w$, and closer investigation will show that the boundary condition in (2) does not provide a unique solution. So we need to add further condition to make the solution unique, and I make an assumption that the total concentration u in an infinitesimal area has uniform concentration per length u_1 and u_2 in each direction x_1 and x_2 accordingly. Hence $u = u_1 u_2$ at every point and therefore we can write

$$\forall x_1 \in \mathbb{R}_0^d \forall x_2 \in \mathbb{R}_0^w \forall t \in \mathbb{R}_0 : u\left(\{x_i\}_{i=1}^2, t\right) = u_1(x_1, t)\, u_2(x_2, t) \,. \tag{3}$$

Putting this into the heat equation (1) we get

$$u_2(x_2, t)\frac{\partial u_1(x_1, t)}{\partial t} + u_1(x_1, t)\frac{\partial u_2(x_2, t)}{\partial t} = D\left(u_2(x_2, t)\frac{\partial^2 u_1(x_1, t)}{\partial {x_1}^2} + u_1(x_1, t)\frac{\partial^2 u_2(x_2, t)}{\partial {x_2}^2}\right),$$

which can be written as two heat equations

$$\forall i \in \mathbb{N}_1^2 : \frac{\partial u_i(x_i, t)}{\partial t} = D\frac{\partial^2 u_i(x_i, t)}{\partial {x_i}^2} \,. \tag{4}$$

I make another assumption that the the boundary at $u(0, x_2, t)$ determined by $u_2(x_2, t)$ alone, and by satisfying initial condition $u(0, x_2, 0)$ for u_2

$$\forall t \in \mathbb{R}_0 \left(: u_2(0, t) = u_2(w, t) = 0 \wedge \forall x_2 \in \mathbb{R}_{w_1}^{w_2} : u_2(x_2, t) = u_0\right) \wedge \forall x_2 \in \mathbb{R}_0^w \setminus \mathbb{R}_{w_1}^{w_2} : u_2(x_2, 0) = 0 \tag{5}$$

we have now determined the values on all the boundaries have therefore Dirichlet boundary condition, and therefore a unique solution of u. We found the analytical solution to the heat equation in (4) for $i = 2$ with similar boundary and initial condition, and I will therefore not show the derivation here, but just state the solutions

$$u_2(x_2,t) = u_0 \begin{cases} \frac{x_2}{w_1} - \sum_{n=1} \frac{2}{n\pi} \sin\left(n\pi\left(1-\frac{x_2}{w_1}\right)\right) \exp\left(-D\left(\frac{n\pi}{w_1}\right)^2 t\right) & : x_2 \in \mathbb{R}_0^{w_1-} \\ 1 & : x_2 \in \mathbb{R}_{w_1}^{w_2} \\ \frac{w-x_2}{w-w_2} - \sum_{n=1} \frac{2}{n\pi} \sin\left(n\pi\frac{x_2-w_2}{w-w_2}\right) \exp\left(-D\left(\frac{n\pi}{w-w_2}\right)^2 t\right) & : x_2 \in \mathbb{R}_{w_2+}^{w} \,. \end{cases} \tag{6}$$

Since I have established that the boundary at $u(0,x_2,t)$ determined by $u_2(x_2,t)$ alone, means that u_1 has the following boundary and initial condition

$$\forall t \in \mathbb{R}_0 \left(: u_1(0,t) = 1 \wedge u_1(d,t) = 0\right) \wedge \forall x_1 \in \mathbb{R}_{0+}^{d-} : u_1(x_1,0) = 0\,. \tag{7}$$

And using the analytical solution from project 4 to heat equation (4) for $i = 1$ with boundary and initial condition in (7), we have

$$u_1(x_1,t) = 1 - \frac{x_1}{d} - \sum_{n=1} \frac{2}{n\pi} \sin\left(n\pi\frac{x_1}{d}\right) \exp\left(-D\left(\frac{n\pi}{d}\right)^2 t\right). \tag{8}$$

To summarize the solution to the concentration u is given by (3) with (6) and (8).

2 Numerical methods

2.1 The θ-rule

The Taylor expansion is given by

$$u(x) = \sum_{n=0} \frac{\overset{(n)}{u}(x_0)}{n!} (x-x_0)^n \tag{9}$$

where $\overset{(n)}{u} = \frac{\mathrm{d}^n u}{\mathrm{d}t^n}$ and x_0 is a initial value where we step from to x . If we now use the first order approximation

$$u(x) \approx u(x_0) + \overset{(1)}{u}(x_0)(x-x_0)\,.$$

The first order differential equation $\overset{(1)}{u}(x) = f(x)$ is determined when we have the initial condition $u(x_0)$, however $\overset{(1)}{u}(x_0)$ is not an initial condition, and it depends on how we calculate it numerically from the initial condition. Now note that $\overset{(1)}{u}(x_0)$ is the same for different values of x in the approximation above and lets say that we calculate it as given from the approximation above;

$$\overset{(1)}{u}(x_0) \approx \frac{u(x) - u(x_0)}{x - x_0}\,. \tag{10}$$

So now use this in another point $x_\theta = \theta x + (1-\theta)\, x_0$ which we also approximate to the first order, and if we use the expression above for $\overset{(1)}{u}(x_0)$ we get

$$\begin{aligned} u(x_\theta) &\approx u(x_0) + \overset{(1)}{u}(x_0)(x_\theta - x_0) = u(x_0) + \theta \overset{(1)}{u}(x_0)(x - x_0) \\ &\approx u(x_0) + \frac{u(x) - u(x_0)}{x - x_0}\theta(x - x_0) = \theta u(x) + (1-\theta)\, u(x_0)\,, \end{aligned} \tag{11}$$

this is known as the θ-rule. The θ-rule can be used to approximate the solution of the following first order differential equation

$$\overset{(1)}{u}(x) = f(u(x))\,, \tag{12}$$

where we use (10) to approximate the expression $\overset{(1)}{u}(x)$ and given an even better or worse approximation to the solution $u(x)$ by approximating $f(x) \approx f(x_\theta)$;

$$\frac{u(x) - u(x_0)}{x - x_0} \approx f\Big(u(x_\theta)\Big) = f\Big(\theta u(x) + (1-\theta)\, u(x_0)\Big)\,,$$

which discretize to

$$\frac{u_{i+1} - u_i}{x_{i+1} - x_i} = f\Big(\theta u_{i+1} + (1-\theta)\, u_i\Big) \qquad \text{where } i \in \mathbb{N}_0 \text{ and } u_0 \text{ is an initial contidion.} \tag{13}$$

We can find the the next step in the numerical solution to (12) by solving this difference equation with regard to u_{i+1}. Note the above discretization is known as Forward Euler scheme (Explicit) when $\theta = 0$, Backward Euler scheme (Implicit) when $\theta = 1$ and Crank-Nicolson scheme when $\theta = \frac{1}{2}$.

2.2 Second order derivative

We approximated the first order derivative in (10), but we need to approximate the second order derivative to be able to solve the diffusion in (1). I start by expanding the Taylor series in (9) around the point $x_0 \pm \Delta x$ accordingly;

$$u(x_0 \pm \Delta x) = \sum_{n=0} \frac{\overset{(n)}{u}(x_0)}{n!}(\pm\Delta x)^n\,. \tag{14}$$

Now adding these two expansions

$$u(x_0 + \Delta x) + u(x_0 - \Delta x) = 2\sum_{n=0} \frac{\overset{(2n)}{u}(x_0)}{(2!)}\Delta x^{2n} = 2u(x_0) + \overset{(2)}{u}(x_0)\,\Delta x^2 + 2\sum_{n=2} \frac{\overset{(2n)}{u}(x_0)}{(2!)}\Delta x^{2n}\,,$$

and solve it with

$$\begin{aligned} \overset{(2)}{u}(x_0) &= \frac{u(x_0 + \Delta x) - 2u(x_0) + u(x_0 - \Delta x)}{\Delta x^2} - 2\sum_{n=2} \frac{\overset{(2n)}{u}(x_0)}{(2!)}\Delta x^{2(n-1)} \\ &= \frac{u(x_0 + \Delta x) - 2u(x_0) + u(x_0 - \Delta x)}{\Delta x^2} + O\Big(\Delta x^2\Big)\,, \end{aligned}$$

So the second order derivative can be approximated with

$$\overset{(2)}{u}(x_0) \approx \frac{u(x_0+\Delta x) - 2u(x_0) + u(x_0-\Delta x)}{\Delta x^2} \tag{15}$$

with the local truncation error $O\left(\Delta x^2\right)$.

2.3 The heat equation

We want to discretize the dimensionless heat equation from (1), where we use $D = 1$, $u_0 = 1$ and $d = 1$,

$$\frac{\partial u\left(\{x_i\}_{i=1}^2, t\right)}{\partial t} = \sum_{\ell=1}^{2} \frac{\partial^2 u\left(\{x_i\}_{i=1}^2, t\right)}{\partial {x_\ell}^2},$$

to numerically solve diffusion of neurotransmitters. First we do the θ-rule discretization in (13)

$$\begin{aligned}\frac{u_{(i+1)\{j_k\}_{k=1}^2} - u_{i\{j_k\}_{k=1}^2}}{\Delta t} &= \sum_{\ell=1}^{2} \frac{\partial^2 u_{(i+\theta)\{j_k\}_{k=1}^2}}{\partial {x_\ell}^2} = \sum_{\ell=1}^{2} \frac{\partial^2 \left(\theta u_{(i+1)\{j_k\}_{k=1}^2} + (1-\theta)\, u_{i\{j_k\}_{k=1}^2}\right)}{\partial {x_\ell}^2} \\ &= \sum_{\ell=1}^{2} \left(\theta \frac{\partial u_{(i+1)\{j_k\}_{k=1}^2}}{\partial {x_\ell}^2} + (1-\theta) \frac{\partial u_{i\{j_k\}_{k=1}^2}}{\partial {x_\ell}^2}\right)\end{aligned}$$

where index i is stepping of t and j_k are stepping of x_k. Note also that the following notation expand accordingly $u_{i\{j_k\}_{k=1}^2} = u_{ij_1j_2}$, which becomes a more elegant notation for larger n in $u_{i\{j_k\}_{k=1}^n} = u_{ij_1j_2\ldots j_n}$. Now we implement the discretization of the second order in (15)

$$\begin{aligned}\frac{u_{(i+1)\{j_k\}_{k=1}^2} - u_{i\{j_k\}_{k=1}^2}}{\Delta t} = \sum_{\ell=1}^{2} \Bigg(&\frac{\theta}{\Delta {x_\ell}^2} \left(u_{(i+1)\{j_k+\delta_{k\ell}\}_{k=1}^2} - 2u_{(i+1)\{j_k\}_{k=1}^2} + u_{(i+1)\{j_k-\delta_{k\ell}\}_{k=1}^2}\right) \\ &+ \frac{1-\theta}{\Delta {x_\ell}^2} \left(u_{i\{j_k+\delta_{k\ell}\}_{k=1}^2} - 2u_{i\{j_k\}_{k=1}^2} + u_{i\{j_k-\delta_{k\ell}\}_{k=1}^2}\right)\Bigg),\end{aligned} \tag{16}$$

where $\delta_{k\ell}$ is the Kronecker delta, and we now clearly see the elegance of the notation $u_{i\{j_k\}_{k=1}^n}$.

The dimensionless initial condition from (4) gives us

$$u_{0\{j_k\}_{k=1}^2} = \begin{cases} 1 & : j_1 = 0 \text{ and } x_{2j_2} \in \mathbb{R}_{w_1}^{w_2} \\ 0 & : \text{elsewhere,} \end{cases}$$

where $x_{2j_2} = x_{20} + \frac{j_2}{\Delta x_2}$. For the explicit scheme $\theta = 0$ we get

$$u_{(i+1)\{j_k\}_{k=1}^2} = u_{i\{j_k\}_{k=1}^2} + \sum_{\ell=1}^{2} \alpha_\ell \left(u_{i\{j_k+\delta_{k\ell}\}_{k=1}^2} - 2u_{i\{j_k\}_{k=1}^2} + u_{i\{j_k-\delta_{k\ell}\}_{k=1}^2}\right) \tag{17}$$

where

$$\alpha_\ell = \frac{\Delta t}{\Delta {x_\ell}^2} \qquad \text{and} \qquad n_\ell = \frac{1}{\Delta x_\ell}.$$

2.4 Jacobi's iterative method

When we have $\theta \neq 0$ in (16) we have implicit schemes, and I rewrite (16) with unknowns on the left side and knowns on the right side;

$$\forall k \in \mathbb{N}_1^2 \forall j_k \in \mathbb{N}_1^{n_k-2} : \left(1 + 2\theta \sum_{\ell=1}^{2} \alpha_\ell\right) u_{(i+1)\{j_k\}_{k=1}^2} - \theta \sum_{\ell=1}^{2} \alpha_\ell \left(u_{(i+1)\{j_k+\delta_{k\ell}\}_{k=1}^2} + u_{(i+1)\{j_k-\delta_{k\ell}\}_{k=1}^2}\right)$$
$$= \left(1 - 2(1-\theta) \sum_{\ell=1}^{2} \alpha_\ell\right) u_{i\{j_k\}_{k=1}^2} + (1-\theta) \sum_{\ell=1}^{2} \alpha_\ell \left(u_{i\{j_k+\delta_{k\ell}\}_{k=1}^2} + u_{i\{j_k-\delta_{k\ell}\}_{k=1}^2}\right),$$

where the boundary are given at the indices $j_k = 0$ and $j_k = n_k - 1$ where n_k are the number of points i k direction. Now I will transform this into a linear algebra problem $\mathbf{A}_{i+1}\mathbf{v}_{i+1} = \mathbf{b}_i$, where elements of $\mathbf{v}_{i+1}$ are

$$v_{(i+1)j} = v_{(i+1)(j_1 n_2 + j_2)} = u_{(i+1)\{j_k\}_{k=1}^2} . \tag{18}$$

The fixed boundary values with $u_0 = 1$ at the presynaptic $x_1 = 0$ and $w_1 \leq x_2 \leq w_2$ are given by the indices j in the set

$$\mathbb{B}_1 = \mathbb{N}_{m_1}^{m_2} .$$

We have non-fixed values at the boundary on the side of the synaptic vesicles at the presynaptic $x_1 = 0$, $0 \leq x_2 < w_1$ and $w_2 < x_2 < w$, and are given by the indices j in the set

$$\mathbb{B}_2 = j \in \mathbb{N}_0^{n_2-1} \setminus \mathbb{B}_1 .$$

The boundary values at the postsynaptic $x_1 = d$ are given by the indices j in the set

$$\mathbb{B}_3 = \mathbb{N}_{n_1 n_2 - n_2}^{n_1 n_2 - 1} .$$

The left side boundary $x_2 = 0$ are given by the indices j in the set

$$\mathbb{B}_4 = \{\ell n_2\}_{\ell=1}^{n_1-2} .$$

The right side boundary $x_2 = w$ are given by the indices j in the set

$$\mathbb{B}_5 = \{\ell n_2 - 1\}_{\ell=2}^{n_1-1} .$$

And the boundary as a whole are given by the indices j in the set

$$\mathbb{B} = \bigcup_{i=1}^{5} \mathbb{B}_i .$$

The inner points are given by the indices j in the set

$$\mathbb{I} = \mathbb{N}_0^{n_1 n_2 - 1} \setminus \mathbb{B}\,.$$

We can now write the matrix elements of $\mathbf{A}_{(i+1)}$

$$a_{(i+1)jk} = a_{jk} = \begin{cases} 1 & : \forall j \in \mathbb{B} \setminus \mathbb{B}_2 : j = k \quad \text{(fixed boundary)} \\ 1 + 2\theta\alpha_2 & : \forall j \in \mathbb{B}_2 : j = k \quad \text{(side of vesicles)} \\ 1 + 2\theta \sum_{\ell=1}^{2} \alpha_\ell & : \forall j \in \mathbb{I} : j = k \quad \text{(the inner points } u_{(i+1)j_1 j_2}\text{)} \\ -\theta\alpha_2 & : \forall j \in \mathbb{I} \cup \mathbb{B}_2 : k \in \left\{ j-1, j+1 \right\} \quad \text{(diffusion in } x_2 \text{ direction)} \\ -\theta\alpha_1 & : \forall j \in \mathbb{I} : k \in \left\{ j-n_2, j+n_2 \right\} \quad \text{(diffusion in } x_1 \text{ direction)} \\ 0 & : \text{otherwise,} \end{cases} \tag{19}$$

and the elements of the vector $\mathbf{b}_i$ are

$$b_{ij} = b_{i(j_1 n_2 + j_2)} = \begin{cases} 1 & : j \in \mathbb{B}_1 \\ \left(1 - 2\left(1-\theta\right)\alpha_2\right) u_{i0j_2} + \left(1-\theta\right)\alpha_2 \left(u_{i0(j_2+1)} + u_{i0(j_2-1)} \right) & : j \in \mathbb{B}_2 \\ \left(1 - 2\left(1-\theta\right)\sum_{\ell=1}^{2} \alpha_\ell\right) u_{i\{j_k\}_{k=1}^{2}} + \left(1-\theta\right)\sum_{\ell=1}^{2} \alpha_\ell \left(u_{i\{j_k+\delta_{k\ell}\}_{k=1}^{2}} + u_{i\{j_k-\delta_{k\ell}\}_{k=1}^{2}} \right) & : j \in \mathbb{I} \\ 0 & : \text{otherwise.} \end{cases} \tag{20}$$

Unfortunately is the linear algebra problem $\mathbf{A}\mathbf{v}_{i+1} = \mathbf{b}_i$ is computational expensive to solve for the matrix in (19). To lower the computational time I therefore prepare the matrix $\mathbf{A}$ for Jacobi's iterative method by splitting it into a diagonal matrix $\mathbf{D}$ and a remainder matrix $\mathbf{R}$;

$$\mathbf{A} = \mathbf{D} + \mathbf{R}\,.$$

From (19) we see that the elements of the diagonal matrix are given by

$$d_{jk} = \begin{cases} 1 & : \forall j \in \mathbb{B} \setminus \mathbb{B}_2 : j = k \quad \text{(fixed boundary)} \\ 1 + 2\theta\alpha_2 & : \forall j \in \mathbb{B}_2 : j = k \quad \text{(side of vesicles)} \\ 1 + 2\theta \sum_{\ell=1}^{2} \alpha_\ell & : \forall j \in \mathbb{I} : j = k \quad \text{(the inner points } u_{(i+1)j_1 j_2}\text{)} \\ 0 & : \text{otherwise,} \end{cases} \tag{21}$$

and the elements of the remainder matrix are given by

$$r_{jk} = \begin{cases} -\theta\alpha_2 & : \forall j \in \mathbb{I} \cup \mathbb{B}_2 : k \in \left\{ j-1, j+1 \right\} \quad \text{(diffusion in } x_2 \text{ direction)} \\ -\theta\alpha_1 & : \forall j \in \mathbb{I} : k \in \left\{ j-n_2, j+n_2 \right\} \quad \text{(diffusion in } x_1 \text{ direction)} \\ 0 & : \text{otherwise.} \end{cases} \tag{22}$$

We can now introduce Jacobi's iterative method

$$\mathbf{v}_{i+1}^{(\ell)} = \begin{cases} \mathbf{v}_i & : \ell = 0 \\ \mathbf{D}^{-1}\left(\mathbf{b}_i - \mathbf{R}\mathbf{v}_{i+1}^{(\ell-1)} \right) & : \ell \in \mathbb{N}_1 \end{cases}$$

where ℓ is the number of iterations and $\ell = 0$ are the starting point. Rewriting it on element form yields

$$v^{(\ell)}_{(i+1)j} = \begin{cases} v_j & : \ell = 0 \\ \frac{1}{d_{jj}} \left(b_j - \sum_{k \neq j} r_{jk} v^{(\ell-1)}_{(i+1)k} \right) & : j \in \mathbb{N}_1 \, . \end{cases}$$

Using (18) and (22) we have the iterative solution to the heat equation in (6) with the boundaries in (2)

$$u^{(\ell)}_{(i+1)\{j_k\}_{k=1}^2} = \begin{cases} u_{i\{j_k\}_{k=1}^2} & : \ell = 0 \\ 1 & : j_1 = 0 \wedge j_2 \in \mathbb{N}^{m_2}_{m_1} \\ c_0 + c_1 \left(u^{(\ell-1)}_{(i+1)0(j_2+1)} + u^{(\ell-1)}_{(i+1)0(j_2-1)} \right) & : j_1 = 0 \wedge j_2 \in \mathbb{N}^{n_2-1}_{0} \setminus \mathbb{N}^{m_2}_{m_1} \\ c_2 + \sum_{o=1}^{2} c_{o+2} \left(u^{(\ell-1)}_{(i+1)\{j_k+\delta_{ko}\}_{k=1}^2} + u^{(\ell-1)}_{(i+1)\{j_k-\delta_{ko}\}_{k=1}^2} \right) & : j_1 \in \mathbb{N}^{n_1-2}_{1} \wedge j_2 \in \mathbb{N}^{n_2-2}_{1} \end{cases} \tag{23}$$

$$c_0 = c_5 u_{i0j_2} + c_6 \left(u_{i0(j_2+1)} + u_{i0(j_2-1)} \right) \tag{24}$$

$$c_1 = \frac{\theta\alpha_2}{1 + 2\theta\alpha_2} \tag{25}$$

$$c_2 = c_7 u_{i\{j_k\}_{k=1}^2} + \sum_{\ell=1}^{2} c_{\ell+7} \left(u_{i\{j_k+\delta_{k\ell}\}_{k=1}^2} + u_{i\{j_k-\delta_{k\ell}\}_{k=1}^2} \right) \tag{26}$$

$$c_3 = \frac{\theta\alpha_1}{1 + 2\theta \sum_{\ell=1}^{2} \alpha_\ell} \tag{27}$$

$$c_4 = \frac{\theta\alpha_2}{1 + 2\theta \sum_{\ell=1}^{2} \alpha_\ell} \tag{28}$$

$$c_5 = \frac{1 - 2(1-\theta)\alpha_2}{1 + 2\theta\alpha_2} \tag{29}$$

$$c_6 = \frac{(1-\theta)\alpha_2}{1 + 2\theta\alpha_2} \tag{30}$$

$$c_7 = \frac{1 - 2(1-\theta) \sum_{\ell=1}^{2} \alpha_\ell}{1 + 2\theta \sum_{\ell=1}^{2} \alpha_\ell} \tag{31}$$

$$c_8 = \frac{(1-\theta)\alpha_1}{1 + 2\theta \sum_{\ell=1}^{2} \alpha_\ell} \tag{32}$$

$$c_9 = \frac{(1-\theta)\alpha_2}{1 + 2\theta \sum_{\ell=1}^{2} \alpha_\ell} \tag{33}$$

The Jacobi iterative method converges when the matrix $\mathbf{A}$ is diagonally dominant

$$\left| a_{ii} \right| > \sum_{j \neq i} \left| a_{ij} \right| ,$$

which is the case for the matrix spanned out by (19) because $1 + 2\theta\alpha_2 > 2\theta\alpha_2$ and $1 + 2\theta \sum_{\ell=1}^{2} \alpha_\ell > 2\theta \sum_{\ell=1}^{2} \alpha_\ell$.

2.5 Markov chain and Metropolis-Hastings algorithm

I assume that diffusion is a memoryless physical process that approximates to a stochastic process, hence diffusion is assumed to satisfy the Markov property and therefore is a Markov process where the transition between states are described by Markov chains

$$w^{+}_{(i+1)\{j_k\}_{k=1}^{n}} = \sum_{\{\ell_k\}_{k=1}^{n}} W_{\{j_k\}_{k=1}^{n}\{\ell_k\}_{k=1}^{n}} w_{i\{\ell_k\}_{k=1}^{n}}, \tag{34}$$

where $w_{i\{j_k\}_{k=1}^{n}}$ is the probability distribution function "PDF" at time step i, and $W_{\{j_k\}_{k=1}^{n}\{\ell_k\}_{k=1}^{n}}$ is the transition probability from state $\{\ell_k\}_{k=1}^{n}$ to $\left\{j_k\right\}_{k=1}^{n}$. And the reverse Markov chain is given by

$$w^{-}_{(i+1)\{j_k\}_{k=1}^{n}} = \sum_{\{\ell_k\}_{k=1}^{n}} W_{\{\ell_k\}_{k=1}^{n}\{j_k\}_{k=1}^{n}} w_{i\{j_k\}_{k=1}^{n}}, \tag{35}$$

Using the assumption that diffusion can be modelled as Markov process then we can use a Master equation to describe the normalized concentration at time step $i+1$ by the difference of transition to and from the state $\left\{j_k\right\}_{k=1}^{n}$ plus the amount that stays in the state;

$$u_{(i+1)\{j_k\}_{k=1}^{n}} = W_{\{j_k\}_{k=1}^{n}\{j_k\}_{k=1}^{n}} w_{i\{j_k\}_{k=1}^{n}} + w^{+}_{(i+1)\{j_k\}_{k=1}^{n}} - w^{-}_{(i+1)\{j_k\}_{k=1}^{n}} \tag{36}$$

Note that the reverse Markov chain is all the transitions that a state $\left\{j_k\right\}_{k=1}$ takes, including transition it self. Therefore the reverse Markov chain in time step $i+1$ describes the transition of the entire concentration at time step i;

$$u_{i\{j_k\}_{k=1}^{n}} = w^{-}_{(i+1)\{j_k\}_{k=1}^{n}}.$$

We assume that one can't stay in the same state $W_{\{j_k\}_{k=1}^{n}\{j_k\}_{k=1}^{n}} = 0$, hence all of the concentration should transition to other states at any given time step, which yields

$$u_{(i+1)\{j_k\}_{k=1}^{n}} = w^{+}_{(i+1)\{j_k\}_{k=1}^{n}} - u_{i\{j_k\}_{k=1}^{n}}. \tag{37}$$

Further diffusion is assumed to be approximated as stochastic move in distance and direction, where distance and direction is stochastically independent. Which means that transition probability can be written as

$$W_{\{j_k\}_{k=1}^{n}\{\ell_k\}_{k=1}^{2}} = T_{\{j_k\}_{k=1}^{n}\{\ell_k\}_{k=1}^{2}} A_{\{j_k\}_{k=1}^{n}\{\ell_k\}_{k=1}^{2}}, \tag{38}$$

where $T_{\{j_k\}_{k=1}^{n}\{\ell_k\}_{k=1}^{2}}$ is the probability of moving the distance between the states $\left\{j_k\right\}_{k=1}^{n}$ and $\{\ell_k\}_{k=1}^{2}$, and $A_{\{j_k\}_{k=1}^{n}\{\ell_k\}_{k=1}^{2}}$ is the probability of moving in the direction between the states $\left\{j_k\right\}_{k=1}^{n}$ and $\{\ell_k\}_{k=1}^{2}$. T and A can be used as proposed and acceptance distribution accordingly in the Metropolis-Hastings algorithm.

Assume that we move a distance ℓ from state $\left\{j_k\right\}_{k=1}^{n}$, which means that the states $\left\{j_k + \delta_{km}\ell\right\}$ are the possible destinations, which puts the following constraint on the direction probability A;

$$\sum_{m=1}^{n}\left(A_{\{j_k+\delta_{km}\ell\}_{k=1}^n\{j_k\}_{k=1}^n}+A_{\{j_k-\delta_{km}\ell\}_{k=1}^n\{j_k\}_{k=1}^n}\right)=1\,.$$

Now assuming isotropic diffusion, which means that

$$\begin{aligned}\forall m,o\in\mathbb{N}_1^n:A_{\{j_k\pm\delta_{km}\ell\}_{k=1}^n\{j_k\}_{k=1}^n}&=A_{\{j_k\pm\delta_{ko}\ell\}_{k=1}^n\{j_k\}_{k=1}^n}\\ \forall m,o\in\mathbb{N}_1^n:T_{\{j_k\pm\delta_{km}\ell\}_{k=1}^n\{j_k\}_{k=1}^n}&=T_{\{j_k\pm\delta_{ko}\ell\}_{k=1}^n\{j_k\}_{k=1}^n}\end{aligned}$$

and the constraint on the direction probability then yields

$$A=A_{\{j_k\}_{k=1}^n\{\ell_k\}_{k=1}^n}=\frac{1}{2n}\,,\tag{39}$$

and the distance probability T is only dependent on the distance ℓ

$$p_\ell=T_{\{j_k\pm\delta_{km}\ell\}_{k=1}^n\{j_k\}_{k=1}^n}\,,$$

which has the constraint

$$\sum_\ell p_\ell=1\,.$$

Then the normalized concentration (37) is now given by

$$u_{(i+1)(j_k)_{k=1}^n}=\frac{1}{2n}\sum_\ell p_\ell\sum_{m=1}^{n}\left(u_{i\{j_k+\delta_{km}\ell\}_{k=1}^n}+u_{i\{j_k-\delta_{km}\ell\}_{k=1}^n}\right)-u_{i\{j_k\}_{k=1}^n}\,,$$

which can be rewritten to

$$\frac{u_{(i+1)\{j_k\}_{k=1}^n}-u_{i\{j_k\}_{k=1}^n}}{\Delta t}=\sum_\ell\frac{p_\ell\,(\ell\Delta x)^2}{2n\Delta t}\sum_{m=1}^{n}\frac{u_{i\{j_k+\delta_{km}\ell\}_{k=1}^n}-2u_{i\{j_k\}_{k=1}^n}+u_{i\{j_k-\delta_{km}\ell\}_{k=1}^n}}{(\ell\Delta x)^2}\,.\tag{40}$$

When we have the Kronecker delta distribution $p_\ell=\delta_{1\ell}$ this becomes the discretization of the heat equation in (1)

$$\frac{u_{(i+1)\{j_k\}_{k=1}^n}-u_{i\{j_k\}_{k=1}^n}}{\Delta t}=\frac{\Delta x^2}{2n\Delta t}\sum_{m=1}^{n}\frac{u_{i\{j_k+\delta_{km}\}_{k=1}^n}-2u_{i\{j_k\}_{k=1}^n}+u_{i\{j_k-\delta_{km}\}_{k=1}^n}}{\Delta x^2}\,,$$

with the relation to the diffusion constant D as follows

$$\Delta x=\sqrt{2nD\Delta t}\,.\tag{41}$$

For a general distribution p_ℓ we see from (40) that the step length now is given by

$$\Delta x_\ell = \ell \Delta x = \ell \sqrt{2nD\Delta t}\,. \tag{42}$$

The steps above with Markov chains are designed to give a approximated time evolution. However if we are only interested in the steady state solution, we can derive a faster algorithm to reach it, which is the Metropolis-Hastings algorithm. To derive the Metropolis-Hastings algorithm we start by looking back at the Master equation in (36) for equilibrium, where we realize that transition to and from the state $\left\{j_k\right\}_{k=1}^n$ must be equal

$$\lim_{i\to\infty}\left(w^+_{(i+1)\{j_k\}_{k=1}^n} - w^-_{(i+1)\{j_k\}_{k=1}^n}\right) = 0$$

and that concentration remains the same

$$\lim_{i\to\infty} u_{(i+1)\{j_k\}_{k=1}^n} = \lim_{i\to\infty} W_{\{j_k\}_{k=1}^n\{j_k\}_{k=1}^n} w_{i\{j_k\}_{k=1}^n} = \lim_{i\to\infty} u_{i\{j_k\}_{k=1}^n}\,.$$

Using the equality of transition to and from the state $\left\{j_k\right\}_{k=1}^n$ at equilibrium the Markov chains in (34) and (35) with independent probability distributions T and A in (38), we have the following equation

$$\sum_{\{\ell_k\}_{k=1}^n}\left(T_{\{j_k\}_{k=1}^n\{\ell_k\}_{k=1}^n} A_{\{j_k\}_{k=1}^n\{\ell_k\}_{k=1}^n} w_{i\{\ell_k\}_{k=1}^n} - T_{\{\ell_k\}_{k=1}^n\{j_k\}_{k=1}^n} A_{\{\ell_k\}_{k=1}^n\{j_k\}_{k=1}^n} w_{i\{j_k\}_{k=1}^n}\right) = 0$$

which is satisfied by the relation

$$\frac{A_{\{j_k\}_{k=1}^n\{\ell_k\}_{k=1}^n}}{A_{\{\ell_k\}_{k=1}^n\{j_k\}_{k=1}^n}} = \frac{T_{\{\ell_k\}_{k=1}^n\{j_k\}_{k=1}^n} w_{i\{j_k\}_{k=1}^n}}{T_{\{j_k\}_{k=1}^n\{\ell_k\}_{k=1}^n} w_{i\{\ell_k\}_{k=1}^n}}\,,$$

where T is a distribution that we propose and A is a probability to accept the proposition. We reach equilibrium fastest when the transition from states are as large as possible, which means $A_{\{\ell_k\}_{k=1}^n\{j_k\}_{k=1}^n}$ should be as close to 1 as possible and still satisfy that $A_{\{j_k\}_{k=1}^n\{\ell_k\}_{k=1}^n} \leq 1$, which is the Metropolis-Hastings algorithm;

$$A_{\{j_k\}_{k=1}^n\{\ell_k\}_{k=1}^n} = \min\left(1, \frac{T_{\{\ell_k\}_{k=1}^n\{j_k\}_{k=1}^n} w_{\{\ell_k\}_{k=1}^n}}{T_{\{j_k\}_{k=1}^n\{\ell_k\}_{k=1}^n} w_{\{j_k\}_{k=1}^n}}\right). \tag{43}$$

If we use a random generator to make a proposition to move at each time step, we get a deterministic running time with approximation in the result, which is known as Monte Carlo algorithm.

3 Attachments

The source files developed are

4 Resources

1. QT Creator 5.3.1 with C11
2. Eclipse Standard/SDK - Version: Luna Release (4.4.0) with PyDev for Python
3. Ubuntu 14.04.1 LTS
4. ThinkPad W540 P/N: 20BG0042MN with 32 GB RAM

References

[1] Morten Hjorth-Jensen, *FYS4150 - Project 5* - *Diffusion in two dimensions*, University of Oslo, 2014

[2] Morten Hjorth-Jensen, *Computational Physics - Lecture Notes Fall 2014*, University of Oslo, 2014

[3] Farnell and Gibson, *Monte Carlo simulation of diffusion in a spatially nonhomogeneous medium:* A biased random walk on an asymmetrical lattice, Journal of Computational Physics, volume 208, pages 253-265, 2005

[4] http://en.wikipedia.org/wiki/Diffusion_equation

[5] http://en.wikipedia.org/wiki/Heat_equation

[6] http://en.wikipedia.org/wiki/Dirichlet_boundary_condition

[7] http://en.wikipedia.org/wiki/Poisson%27s_equation

[8] http://en.wikipedia.org/wiki/Uniqueness_theorem_for_Poisson%27s_equation

[9] http://en.wikipedia.org/wiki/Ansatz

[10] http://en.wikipedia.org/wiki/Kronecker_delta

[11] http://en.wikipedia.org/wiki/Jacobi_method

[12] http://en.wikipedia.org/wiki/Stochastic_process

[13] http://en.wikipedia.org/wiki/Independence_%28probability_theory%29

[14] http://en.wikipedia.org/wiki/Markov_property

[15] http://en.wikipedia.org/wiki/Markov_process

[16] http://en.wikipedia.org/wiki/Markov_chain

[17] http://en.wikipedia.org/wiki/Master_equation

[18] http://en.wikipedia.org/wiki/Metropolis%E2%80%93Hastings_algorithm

[19] http://en.wikipedia.org/wiki/Monte_Carlo_algorithm

[20] http://en.wikipedia.org/wiki/Isotropy