Numerical solution to differential equations with different schemes

Pedro de la Torre Luque

July 2021

1 2D diffusion equation

Here, we present the diffusion equation assuming cylindrical symmetry of the system.

$$\frac{\partial u}{\partial t}(r,z,t,E_e) = \frac{1}{r}\frac{\partial}{\partial r}\left(rD_r\frac{\partial u}{\partial r}(r,z,t,E_e)\right) + \frac{\partial}{\partial z}\left(D_z\frac{\partial u}{\partial z}(r,z,t,E_e)\right) + \\ +\mathcal{S}(r,z,t,E_e) - \frac{\partial}{\partial E_e}\left(\frac{\partial E_e}{\partial t}u(r,z,t,E_e)\right) \tag{1}$$

Notice that:

$$\frac{1}{r}\frac{\partial}{\partial r}\left(rD_r\frac{\partial u}{\partial r}(r,z,t,E_e)\right) = D_r\left(r^{-1}\frac{\partial u}{\partial r}(r,z,t,E_e) + \frac{\partial^2 u}{\partial r^2}(r,z,t,E_e)\right) \tag{2}$$

And that:

$$\frac{\partial}{\partial z}\left(D_z\frac{\partial u}{\partial z}(r,z,t,E_e)\right) = D_z\frac{\partial^2 u}{\partial z^2}(r,z,t,E_e). \tag{3}$$

This simplification is possible because the diffusion coefficients are homogeneous (same in every point of the grid, but not in every direction: Anisotropic diffusion). See, for example, eq.1 of Ref. [2] for more details.

Then, we have the next terms:

Injection spectrum:

$$\mathcal{S}(E,t) = Q(E)S(t)\delta(r)\delta(z)$$

Energy losses due to synchrotron and IC:

$$\frac{\partial E_e}{\partial t} = -\frac{4}{3}\,c\,\sigma_T\left(U_B + \mathcal{F}_{KN}(E)U_{ph}\right)\left(\frac{E}{m_e c^2}\right)^2$$

With: $U_{ph} = U_{CMB} + U_{IR} + U_{opt} + U_{UV}$ and $U_B = \frac{B_0^2}{4\pi}$.

Then, $D_r = D_\perp$ and $D_z = D_\parallel$, so that we could write the diffusion matrix as:

$$D(r,z) = \begin{pmatrix} D_\perp & 0 & 0 \\ 0 & D_\perp & 0 \\ 0 & 0 & D_\parallel \end{pmatrix}, \qquad D_\perp = D_\parallel M_A^4.$$

Given that there should not be any crossed derivatives, we neglect the other terms of the diffusion coefficient.

1.1 Discretization of 2D diffusion equation

- Forward Euler (explicit)

$$\begin{aligned}\frac{u_{i,k,\epsilon}^{\tau+1} - u_{i,k,\epsilon}^{\tau}}{\Delta t} &= \frac{D_\perp}{\Delta r^2}(u_{i+1,k,\epsilon}^{\tau} - 2u_{i,k,\epsilon}^{\tau} + u_{i-1,k,\epsilon}^{\tau}) + \frac{D_\perp}{r_i}\frac{(u_{i+1,k,\epsilon}^{\tau} - u_{i-1,k,\epsilon}^{\tau})}{2\Delta r} + \\ &+ \frac{D_\parallel}{\Delta z^2}(u_{i,k+1,\epsilon}^{\tau} - 2u_{i,k,\epsilon}^{\tau} + u_{i,k-1,\epsilon}^{\tau}) - b(E_\epsilon) \times \frac{(u_{i,k,\epsilon+1}^{\tau} - u_{i,k,\epsilon}^{\tau})}{\Delta E} - u_{i,k,\epsilon}^{\tau}\frac{-4}{3}\frac{c\,\sigma_T}{(m_e c^2)^2}U_B 2E_\epsilon + \\ &- u_{i,k,\epsilon}^{\tau}\left(\frac{-4}{3}\frac{c\,\sigma_T}{(m_e c^2)^2}U_{ph}\right)\left[2E_\epsilon \mathcal{F}_{KN}(E_\epsilon) + \left(E_e^2 \frac{\mathcal{F}_{KN}(E_{\epsilon+1}) - \mathcal{F}_{KN}(E_\epsilon)}{\Delta E}\right)\right] + \\ &+ S(r_i, z_k, E_\epsilon, t^{\tau+1/2}) \ . \end{aligned} \tag{4}$$

Notice that this is only order $\mathcal{O}(\Delta E)$ in energy but $\mathcal{O}(\Delta r^2)$ and $\mathcal{O}(\Delta z^2)$. Rearranging this equation gives us an easy formula to implement as our solver:

$$\begin{aligned} u_{i,k,\epsilon}^{\tau+1} &= u_{i,k,\epsilon}^{\tau}\left(1 - 2\alpha_1 - 2\alpha_3 + b(E_\epsilon)\alpha_4 - \alpha_5\left(U_B 2E_\epsilon + U_{ph}\Delta[bE^2]\right)\right) + \\ &+ u_{i+1,k,\epsilon}^{\tau}\left(\alpha_1 + \frac{\alpha_2}{r_i}\right) + u_{i-1,k,\epsilon}^{\tau}\left(\alpha_1 - \frac{\alpha_2}{r_i}\right) + u_{i,k+1,\epsilon}^{\tau}\alpha_3 + u_{i,k-1,\epsilon}^{\tau}\alpha_3 + \\ &- u_{i,k,\epsilon+1}^{\tau} b(E_\epsilon)\alpha_4 + \Delta t S(r_i, z_k, E_\epsilon, t^{\tau+1/2}) \ , \end{aligned} \tag{5}$$

where we have abbreviated

$$\left[2E_\epsilon \mathcal{F}_{KN}(E_\epsilon) + E_\epsilon^2 \frac{\mathcal{F}_{KN}(E_{\epsilon+1}) - \mathcal{F}_{KN}(E_\epsilon)}{\Delta E}\right] \equiv \Delta[bE^2],$$

and where we defined the factors:

$$\alpha_1 = \frac{D_\perp}{\Delta r^2}\Delta t, \ \alpha_2 = \frac{D_\perp}{2\Delta r}\Delta t, \ \alpha_3 = \frac{D_\parallel}{\Delta z^2}\Delta t, \ \alpha_4 = \frac{\Delta t}{\Delta E}, \ \alpha_5 = -\frac{4}{3}\frac{c\,\sigma_T}{(m_e c^2)^2}\Delta t.$$

Easily solved with the suitable boundary condition (**Von Neumann boundary conditions** at $r = 0$ and **Dirichlet boundary conditions** for the edges). Specifically, this should be solved with a for loop in the energy dimension, starting from $\epsilon = \epsilon_{max}$ to $\epsilon = 1$ (i.e., backwards in energy), as explained in the supplemental material of Ref. [2].

- Crank-Nicholson (implicit)

$$
\begin{aligned}
\frac{u^{\tau+1}_{i,k,\epsilon} - u^{\tau}_{i,k,\epsilon}}{\Delta t} = \frac{D_\perp}{\Delta r^2} \times \frac{1}{2}\left[(u^{\tau}_{i+1,k,\epsilon} - 2u^{\tau}_{i,k,\epsilon} + u^{\tau}_{i-1,k,\epsilon}) + (u^{\tau+1}_{i+1,k,\epsilon} - 2u^{\tau+1}_{i,k,\epsilon} + u^{\tau+1}_{i-1,k,\epsilon}) \right] + \\
+ \frac{D_\perp}{2r_i} \frac{(u^{\tau}_{i+1,k,\epsilon} - u^{\tau}_{i-1,k,\epsilon}) + (u^{\tau+1}_{i+1,k,\epsilon} - u^{\tau+1}_{i-1,k,\epsilon})}{2\Delta r} + \\
+ \frac{D_\parallel}{\Delta z^2} \times \frac{1}{2}\left[(u^{\tau}_{i,k+1,\epsilon} - 2u^{\tau}_{i,k,\epsilon} + u^{\tau}_{i,k-1,\epsilon}) + (u^{\tau+1}_{i,k+1,\epsilon} - 2u^{\tau+1}_{i,k,\epsilon} + u^{\tau+1}_{i,k-1,\epsilon}) \right] + \\
- \frac{b(E_\epsilon)}{2} \times \frac{(u^{\tau}_{i,k,\epsilon+1} - u^{\tau}_{i,k,\epsilon-1}) + (u^{\tau+1}_{i,k,\epsilon+1} - u^{\tau+1}_{i,k,\epsilon-1})}{2\Delta E} + \frac{1}{2}\left(u^{\tau}_{i,k,\epsilon} + u^{\tau+1}_{i,k,\epsilon} \right) \frac{4}{3} \frac{c\,\sigma_T\, U_B}{(m_e c^2)^2} 2E_\epsilon + \\
+ \frac{1}{2}\left(u^{\tau}_{i,k,\epsilon} + u^{\tau+1}_{i,k,\epsilon} \right) \frac{4}{3} \frac{c\,\sigma_T U_{ph}}{(m_e c^2)^2} 2E_\epsilon \mathcal{F}_{KN}(E_\epsilon) + \\
+ \frac{1}{2}\left(u^{\tau}_{i,k,\epsilon} + u^{\tau+1}_{i,k,\epsilon} \right) \frac{4}{3} \frac{c\,\sigma_T U_{ph}}{(m_e c^2)^2} \left(E^2_\epsilon \frac{\mathcal{F}_{KN}(E_{\epsilon+1}) - \mathcal{F}_{KN}(E_{\epsilon-1})}{2\Delta E} \right) + \\
+ S(r_i, z_k, E_\epsilon, t^{\tau+1/2}) \ .
\end{aligned} \tag{6}
$$

With $\Delta E = 1/2 \times (E_{\epsilon+1} - E_{\epsilon-1})$.

Please, notice that this is $\mathcal{O}(\Delta E^2)$ also and that $\frac{\partial b}{\partial E} \neq \frac{b(E_{\epsilon+1}) - b(E_\epsilon)}{\Delta E}$, where $b(E) = \frac{\partial E_e}{\partial t}$, but, actually, you have:

$$
\begin{aligned}
\frac{\partial}{\partial E_e}\left(-\frac{4}{3} c\,\sigma_T \left(U_B + \mathcal{F}_{KN}(E_e) U_{ph}\right) \left(\frac{E_e}{m_e c^2} \right)^2 \right) \longrightarrow \\
-\frac{4}{3} c\,\sigma_T \left[\left(\frac{U_B}{(m_e c^2)^2} \frac{\partial}{\partial E_e} E^2_e \right) + \left(\frac{U_{ph}}{(m_e c^2)^2} \frac{\partial}{\partial E_e} E^2_e \mathcal{F}_{KN}(E_e) \right) \right] = \\
-\frac{4}{3} c\,\sigma_T \left[\frac{2E_e U_B}{(m_e c^2)^2} + \frac{U_{ph}}{(m_e c^2)^2} \left(\mathcal{F}_{KN}(E_e) 2E_e + E^2_e \frac{\partial \mathcal{F}_{KN}(E_e)}{\partial E_e} \right) \right]
\end{aligned} \tag{7}
$$

Equation 6 can be reorganized as:

$$
\begin{aligned}
u_{i,k,\epsilon}^{\tau+1} - u_{i,k,\epsilon}^{\tau} = \alpha_1 \left(u_{i+1,k,\epsilon}^{\tau} - 2u_{i,k,\epsilon}^{\tau} + u_{i-1,k,\epsilon}^{\tau} + u_{i+1,k,\epsilon}^{\tau+1} - 2u_{i,k,\epsilon}^{\tau+1} + u_{i-1,k,\epsilon}^{\tau+1} \right) + \\
+ \frac{\alpha_2}{r_i} \left(u_{i+1,k,\epsilon}^{\tau} - u_{i-1,k,\epsilon}^{\tau} + u_{i+1,k,\epsilon}^{\tau+1} - u_{i-1,k,\epsilon}^{\tau+1} \right) + \\
+ \alpha_3 \left(u_{i,k+1,\epsilon}^{\tau} - 2u_{i,k,\epsilon}^{\tau} + u_{i,k-1,\epsilon}^{\tau} + u_{i,k+1,\epsilon}^{\tau+1} - 2u_{i,k,\epsilon}^{\tau+1} + u_{i,k-1,\epsilon}^{\tau+1} \right) + \\
+ \alpha_4 b(E_\epsilon) \left(u_{i,k,\epsilon+1}^{\tau} - u_{i,k,\epsilon-1}^{\tau} + u_{i,k,\epsilon+1}^{\tau+1} - u_{i,k,\epsilon-1}^{\tau+1} \right) + \alpha_5 U_B 2E_\epsilon \left(u_{i,k,\epsilon}^{\tau} + u_{i,k,\epsilon}^{\tau+1} \right) + \\
+ \alpha_5 U_{ph} \left(u_{i,k,\epsilon}^{\tau} + u_{i,k,\epsilon}^{\tau+1} \right) \left[2E_\epsilon \mathcal{F}_{KN}(E_\epsilon) + E_\epsilon^2 \mathcal{F'}_{KN}(E_\epsilon \right] + \\
+ 2\Delta t S(r_i, z_k, E_\epsilon, t^{\tau+1/2}) ,
\end{aligned} \tag{8}
$$

where these pre-CN coefficients are defined as:

$$
\alpha_1 = \frac{D_\perp}{2\Delta r^2}\Delta t,\ \ \alpha_2 = \frac{D_\perp}{4r\Delta r}\Delta t,\ \ \alpha_3 = \frac{D_\parallel}{2\Delta z^2}\Delta t,\ \ \alpha_4 = \frac{-\Delta t}{4\Delta E}^{1},\ \ \alpha_5 = \frac{\Delta t}{2} \times \frac{4}{3} \frac{c\,\sigma_T}{(m_e c^2)^2}
$$

.

Unfortunately, this equation is computationally too hard to solve because of the several dimensions involved. Thus, we need to make use of the alternating-direction implicit (ADI) method, which is mantein the same accuracy. This requires converting equation 8 into two equations (actually, we follow the Peaceman and Rachford scheme, which does not work for 3D equations unless you divide the time-steps in $\Delta/3$ steps), which differentiate in steps of $\Delta t/2$ instead of in Δt (which implies that $\Delta t \rightarrow \Delta t/2$):

$$
\begin{aligned}
u_{i,k,\epsilon}^{\tau+1/2} - u_{i,k,\epsilon}^{\tau} = \alpha_1 \left(u_{i+1,k,\epsilon}^{\tau} - 2u_{i,k,\epsilon}^{\tau} + u_{i-1,k,\epsilon}^{\tau} \right) + \frac{\alpha_2}{r_i} \left(u_{i+1,k,\epsilon}^{\tau} - u_{i-1,k,\epsilon}^{\tau} \right) + \\
+ \alpha_3 \left(u_{i,k+1,\epsilon}^{\tau+1/2} - 2u_{i,k,\epsilon}^{\tau+1/2} + u_{i,k-1,\epsilon}^{\tau+1/2} \right) + \alpha_4 b(E_\epsilon) \left(u_{i,k,\epsilon+1}^{\tau} - u_{i,k,\epsilon}^{\tau} \right) + \alpha_5 U_B 2E_\epsilon u_{i,k,\epsilon}^{\tau} + \\
+ \alpha_5 U_{ph} u_{i,k,\epsilon}^{\tau} \left[2E_\epsilon \mathcal{F}_{KN}(E_\epsilon) + E_\epsilon^2 \mathcal{F'}_{KN}(E_\epsilon) \right] + \frac{\Delta t}{2} S(r_i, z_k, E_\epsilon, t^{\tau+1/4}) .
\end{aligned} \tag{9}
$$

Which is, at the end, expanded forward only in the k dimension (physically, the z axis), and,in order to reduce the difficulty of the computation, reduced the accuracy in the ϵ dimension to $\mathcal{O}(\Delta E)$ (the derivative of the density with energy - the term involving α_4 - have been simplified so that α_4 is a factor 2 greater than defined above.

[1] From equation 9, this factor is treated implicitly ($\mathcal{O}(\Delta E)$), so we have α_4 divided by 2 and not by 4.

The companion equation is:

$$u_{i,k,\epsilon}^{\tau+1} - u_{i,k,\epsilon}^{\tau+1/2} = \alpha_1 \left(u_{i+1,k,\epsilon}^{\tau+1} - 2u_{i,k,\epsilon}^{\tau+1} + u_{i-1,k,\epsilon}^{\tau+1} \right) + \frac{\alpha_2}{r_i} \left(u_{i+1,k,\epsilon}^{\tau+1} - u_{i-1,k,\epsilon}^{\tau+1} \right) +$$
$$+\alpha_3 \left(u_{i,k+1,\epsilon}^{\tau+1/2} - 2u_{i,k,\epsilon}^{\tau+1/2} + u_{i,k-1,\epsilon}^{\tau+1/2} \right) + \alpha_4 b(E_\epsilon) \left(u_{i,k,\epsilon+1}^{\tau+1/2} - u_{i,k,\epsilon}^{\tau+1/2} \right) + \alpha_5 U_B 2E_\epsilon u_{i,k,\epsilon}^{\tau+1/2} +$$
$$+\alpha_5 U_{ph} u_{i,k,\epsilon}^{\tau+1/2} \left[2E_\epsilon \mathcal{F}_{KN}(E_\epsilon) + E_\epsilon^2 \mathcal{F}'_{KN}(E_\epsilon) \right] + \frac{\Delta t}{2} S(r_i, z_k, E_\epsilon, t^{\tau+3/4}) . \tag{10}$$

Notice here two facts: First, we could evaluate the source term at $\tau+1/2$ (actually, we should take the average of the source term between τ and $\tau+1$) and the accuracy of the error would always be the same ($\mathcal{O}(\Delta t^2)$). Second, that in both equations (eqs. 9 and 10) the energy dimension can be treated implicitly (it will be calculated as the most internal loop for).

The only step needed here, is to form the matrices for calculating these terms. First, let's reorganize eq. 9:

$$u_{i,k,\epsilon}^{\tau+1/2}(1+2\alpha_3) - u_{i,k+1,\epsilon}^{\tau+1/2}\alpha_3 - u_{i,k-1,\epsilon}^{\tau+1/2}\alpha_3 =$$
$$u_{i,k,\epsilon}^{\tau} \left(1 - 2\alpha_1 - \alpha_4 b(E_\epsilon) + \alpha_5 U_B 2E_\epsilon + \alpha_5 U_{ph} \Delta[bE^2]\right) +$$
$$+u_{i+1,k,\epsilon}^{\tau} \left(\alpha_1 + \frac{\alpha_2}{r_i}\right) + u_{i-1,k,\epsilon}^{\tau} \left(\alpha_1 - \frac{\alpha_2}{r_i}\right) + u_{i,k,\epsilon+1}^{\tau}\alpha_4 b(E_\epsilon) + \frac{\Delta t}{2} S(r_i, z_k, E_\epsilon, t^{\tau+1/4}) \tag{11}$$

And for eq. 10:

$$\begin{aligned} u_{i,k,\epsilon}^{\tau+1}(1+2\alpha_1) - u_{i+1,k,\epsilon}^{\tau+1}\left(\alpha_2 + \frac{\alpha_2}{r_i}\right) - u_{i-1,k,\epsilon}^{\tau+1}\left(\alpha_1 - \frac{\alpha_2}{r_i}\right) &= \\ +u_{i,k,\epsilon}^{\tau+1/2}\left(1 - 2\alpha_3 - \alpha_4 b(E_\epsilon) + \alpha_5 U_B 2E_\epsilon + \alpha_5 U_{ph}\Delta[bE^2]\right) &+ \\ +u_{i,k+1,\epsilon}^{\tau+1/2}\alpha_3 + u_{i,k-1,\epsilon}^{\tau+1/2}\alpha_3 + u_{i,k,\epsilon+1}^{\tau+1/2}\alpha_4 b(E_\epsilon) + \frac{\Delta t}{2} S(r_i, z_k, E_\epsilon, t^{\tau+3/4}) & \end{aligned} \tag{12}$$

Here, we have abbreviated

$$\left[2E_\epsilon \mathcal{F}_{KN}(E_\epsilon) + E_\epsilon^2 \mathcal{F}'_{KN}(E_\epsilon)\right] \equiv \Delta[bE^2]$$

These two equations can be rewritten as:

$$\begin{aligned} A u_{i,k,\epsilon}^{\tau+1/2} + B u_{i,k+1,\epsilon}^{\tau+1/2} + C u_{i,k-1,\epsilon}^{\tau+1/2} &= D u_{i,k,\epsilon}^{\tau} + E u_{i+1,k,\epsilon}^{\tau} + F u_{i-1,k,\epsilon}^{\tau} + G u_{i,k,\epsilon+1}^{\tau} + \Delta t S \\ A' u_{i,k,\epsilon}^{\tau+1} + B' u_{i+1,k,\epsilon}^{\tau+1} + C' u_{i-1,k,\epsilon}^{\tau+1} &= D' u_{i,k,\epsilon}^{\tau+1/2} + E' u_{i,k+1,\epsilon}^{\tau+1/2} + F' u_{i,k-1,\epsilon}^{\tau+1/2} + G' u_{i,k,\epsilon+1}^{\tau+1/2} + \Delta t S \end{aligned} \tag{13}$$

The easiest way to solve for each of the points in the grid is to evaluate these linear equations as matrices, evaluating these matrices in every point in energy. In this way, for every point in the i and k grid, we will have a linear equation to solve. As a result, we have two tridiagonal matrices for equation 11 and other two for eq. 12, one for the left-hand side and another for the right hand side:

$$
\begin{aligned}
\mathcal{M}_{xx'} &= \begin{bmatrix} A & B & 0 & 0 & 0 & ... & 0 \\ C & A & B & 0 & 0 & ... & 0 \\ 0 & C & A & B & 0 & ... & 0 \\ 0 & 0 & C & A & B & ... & 0 \\ & & & ... & & & \\ 0 & 0 & ... & 0 & 0 & C & A \end{bmatrix} \quad \text{for the left-hand side} \\
\mathcal{V}_{x_2x_2'} &= \begin{bmatrix} D & E & 0 & 0 & 0 & ... & 0 \\ F & D & E & 0 & 0 & ... & 0 \\ 0 & F & D & E & 0 & ... & 0 \\ 0 & 0 & F & D & E & ... & 0 \\ & & & ... & & & \\ 0 & 0 & ... & 0 & 0 & F & D \end{bmatrix} \quad \text{for the right-hand side}
\end{aligned}
\tag{14}
$$

With a row for every point in the grid that we are evaluating. Here, x means the index along the corresponding dimension, for the matrix in the left hand side of the equation, it will correspond to one dimension, for the matrix in the right hand side of the equation, it will correspond to the other spatial dimension.

Then, we also have the vector designating the density at each point. For the case of eq. 12, we would have the matrices $\mathcal{M}_{kk'}$ and $\mathcal{V}_{ii'}$ (this one depends also on the value of ϵ, which is evaluated in a for loop in energy space) and the density vectors are:

$$
\begin{gathered}
\mathcal{U}^{\tau+1/2}_{i,\cdot,\epsilon} = \begin{bmatrix} U^{\tau+1/2}_{i,1,\epsilon} \\ U^{\tau+1/2}_{i,2,\epsilon} \\ U^{\tau+1/2}_{i,3,\epsilon} \\ U^{\tau+1/2}_{i,4,\epsilon} \\ ... \\ U^{\tau+1/2}_{i,N_\parallel-1,\epsilon} \end{bmatrix} \quad \mathcal{U}^{\tau}_{\cdot,k,\epsilon} = \begin{bmatrix} U^{\tau}_{1,k,\epsilon} \\ U^{\tau}_{2,k,\epsilon} \\ U^{\tau}_{3,k,\epsilon} \\ U^{\tau}_{4,k,\epsilon} \\ ... \\ U^{\tau+1/2}_{N_\perp-1,k,\epsilon} \end{bmatrix} \\
\text{summed to the vector: } \mathcal{C}_{i,\cdot,\epsilon} = \begin{bmatrix} U^{\tau}_{i,1,\epsilon+1} + \Delta t S(r_i, z_1, E_\epsilon, t^{\tau+1/4}) \\ U^{\tau}_{i,2,\epsilon+1} + \Delta t S(r_i, z_2, E_\epsilon, t^{\tau+1/4}) \\ U^{\tau}_{i,3,\epsilon+1} + \Delta t S(r_i, z_3, E_\epsilon, t^{\tau+1/4}) \\ U^{\tau}_{i,4,\epsilon+1} + \Delta t S(r_i, z_4, E_\epsilon, t^{\tau+1/4}) \\ ... \\ U^{\tau}_{i,N_\parallel-1,\epsilon} + \Delta t S(r_i, z_{N_\parallel-1}, E_\epsilon, t^{\tau+1/4}) \end{bmatrix}
\end{gathered}
\tag{15}
$$

And analogously for eq. 12. We end up with:

$$\begin{aligned} \mathcal{U}_{i,.,\epsilon}^{\tau+1/2} &= \mathcal{M}_{kk'}^{-1} \left[\left(\mathcal{V}_{ii'} \times \mathcal{U}_{.,k,\epsilon}^{\tau} \right) + \mathcal{C}_{i,.,\epsilon}^{\tau} \right] \\ \mathcal{U}_{.,k,\epsilon}^{\tau+1} &= \mathcal{M}'^{-1}_{ii'} \left[\left(\mathcal{V}'_{kk'} \times \mathcal{U}_{i,.,\epsilon}^{\tau+1/2} \right) + \mathcal{C}_{.,k,\epsilon}^{\tau+1/2} \right] \end{aligned} \tag{16}$$

Here, it is crucial to consider the boundary conditions in order to solve this equation. Specifically, it is important to set the boundary condition for energy as $u_{i,k,\epsilon_{max}}^{\tau+1/2} = 0$, because we be inside a loop in the energy dimension, starting from $\epsilon = \epsilon_{max}$ to $\epsilon = 1$ (i.e., backwards in energy), as explained in the supplemental material of Ref. [2]. The rest of conditions are set similarly, such that at boundaries ($r = L_r$ and $z = L_z$) we allow to have free escape of particles (**Dirichlet boundary conditions**) and at $r = 0$ we make use of the mirror condition (**Von Neumann boundary condition**), for which $u_{-1,k,\epsilon}^{\tau} = u_{1,k,\epsilon}^{\tau}$.

It is important to realise that the matrix of density is defined as U[r, z, E], which, in python, means that the first dimension (U[i]) corresponds to a slice along the jth (perpendicular, z, dimension) and ϵth (energy dimension) of the ith dimension (r in this case). This means that you will have a fixed r position in the z, E space. This is important for matrix multiplication, and essentially, means that the definition of the density matrix is:

$$\mathcal{U}^{\tau}[:,:,ie] = \begin{bmatrix} U_{0,0,ie}^{\tau} & U_{0,1,ie}^{\tau} & U_{0,2,ie}^{\tau} & \cdots & U_{0,dimz,ie}^{\tau} \\ U_{1,0,ie}^{\tau} & U_{1,1,ie}^{\tau} & U_{1,2,ie}^{\tau} & \cdots & U_{1,dimz,ie}^{\tau} \\ U_{2,0,ie}^{\tau} & U_{2,1,ie}^{\tau} & U_{2,2,ie}^{\tau} & \cdots & U_{2,dimz,ie}^{\tau} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ U_{dimr,0,ie}^{\tau} & U_{0,1,ie}^{\tau} & U_{dimr,2,ie}^{\tau} & \cdots & U_{dimr,dimz,ie}^{\tau} \end{bmatrix} \tag{17}$$

Therefore, when we multiply with a tridiagonal matrix over the r space ($\mathcal{U}_{i-1}^{\tau}, \mathcal{U}_{i}^{\tau}, \mathcal{U}_{i+1}^{\tau}$) ($ith$ coordinate), we need simply use np.dot(Alpha, U), but when we multiply along the z space (jth coordinate), the multiplication must be to the transposed matrix, np.dot(Alpha, U.T), and then, before summing to the density matrix at the last time-step (which has size [Nr, Nz]), make the transpose again $\longrightarrow$ np.dot(Alpha, U.T).T.

2 3D diffusion equation

Here, we present the diffusion equation assuming no symmetry of the system and treated in Cartesian coordinates. This is the diffusion equation defining the density of CR particles for every point in energy and space:

$$\frac{\partial u}{\partial t}(x,y,z,E_e,t) = \frac{\partial}{\partial x}\left(D(x,y,z,E_e)\frac{\partial u}{\partial x}(x,y,z,E_e,t)\right) + \frac{\partial}{\partial y}\left(D(x,y,z,E_e)\frac{\partial u}{\partial y}(x,y,z,E_e,t)\right) + \\ + \frac{\partial}{\partial z}\left(D(x,y,z,E_e)\frac{\partial u}{\partial z}(x,y,E_e,t)\right) + \mathcal{S}(x,y,z,E_e,t) + \frac{\partial}{\partial E_e}\left(\frac{\partial E_e}{\partial t}u(x,y,z,E_e,t)\right) \tag{18}$$

Which can be discretized and numerically solved using:

- Explicit forward Euler

$$\frac{u^{\tau+1}_{i,k,\epsilon} - u^{\tau}_{i,k,\epsilon}}{\Delta t} = \frac{D_x}{\Delta x^2}\left(u^{\tau}_{i+1,j,k,\epsilon} - 2u^{\tau}_{i,j,k,\epsilon} + u^{\tau}_{i-1,j,k,\epsilon}\right) + \\ + \frac{D_y}{\Delta y^2}\left(u^{\tau}_{i,j+1,k,\epsilon} - 2u^{\tau}_{i,j,k,\epsilon} + u^{\tau}_{i,j-1,k,\epsilon}\right) + \frac{D_z}{\Delta z^2}\left(u^{\tau}_{i,j,k+1,\epsilon} - 2u^{\tau}_{i,j,k,\epsilon} + u^{\tau}_{i,j,k-1,\epsilon}\right) + \\ - b(E_\epsilon) \times \frac{(u^{\tau}_{i,j,k,\epsilon+1} - u^{\tau}_{i,j,k,\epsilon})}{\Delta E} + u^{\tau}_{i,j,k,\epsilon}\left(\frac{4}{3}\frac{c\,\sigma_T}{(m_e c^2)^2}\right) 2E_\epsilon\left(U_B + U_{ph}\mathcal{F}_{KN}(E_\epsilon)\right) + \\ + u^{\tau}_{i,j,k,\epsilon}\left(\frac{4}{3}\frac{c\,\sigma_T}{(m_e c^2)^2}U_{ph}\right)\left(E_e^2\mathcal{F}'_{KN}(E_\epsilon)\right) + S(x_i,y_j,z_k,E_\epsilon,t^{\tau+1/2})\ . \tag{19}$$

Rearranging this equation gives us an easy formula to implement as our solver:

$$u^{\tau+1}_{i,j,k,\epsilon} = u^{\tau}_{i,j,k,\epsilon}\left(1 - 2\alpha_1 - 2\alpha_2 - 2\alpha_3 - b(E_\epsilon)\alpha_4 + \alpha_5\left(U_B 2E_\epsilon + U_{ph}\Delta[bE^2]\right)\right) + \\ + u^{\tau}_{i+1,j,k,\epsilon}\alpha_1 + u^{\tau}_{i-1,j,k,\epsilon}\alpha_1 + u^{\tau}_{i,j+1,k,\epsilon}\alpha_2 + u^{\tau}_{i,j-1,k,\epsilon}\alpha_2 + u^{\tau}_{i,j,k+1,\epsilon}\alpha_3 + u^{\tau}_{i,j,k-1,\epsilon}\alpha_3 + \\ + u^{\tau}_{i,j,k,\epsilon+1}b(E_\epsilon)\alpha_4 + \Delta t S(x_i,y_j,z_k,E_\epsilon,t^{\tau+1/2})\ . \tag{20}$$

This time, the factors are:

$$\alpha_1 = \frac{D_x}{\Delta x^2}\Delta t, \quad \alpha_2 = \frac{D_y}{\Delta y^2}\Delta t, \quad \alpha_3 = \frac{D_z}{\Delta z^2}\Delta t\,, \tag{21}$$

and the terms α_4 and α_5 are the same as defined above.

Finally, this can be reduced to:

$$u^{\tau+1}_{i,j,k,\epsilon} = A u^{\tau}_{i,j,k,\epsilon} + \alpha_1\left(u^{\tau}_{i+1,j,k,\epsilon} + u^{\tau}_{i-1,j,k,\epsilon}\right) + \alpha_2\left(u^{\tau}_{i,j+1,k,\epsilon} + u^{\tau}_{i,j-1,k,\epsilon}\right) + \\ + \alpha_3\left(u^{\tau}_{i,j,k+1,\epsilon} + u^{\tau}_{i,j,k-1,\epsilon}\right) + \alpha_4 b(E_\epsilon)u^{\tau}_{i,j,k,\epsilon+1} + \Delta t S(x_i,y_j,z_k,E_\epsilon,t^{\tau+1/2})\ . \tag{22}$$

- Crank-Nicholson scheme

From the discretization of eq. 18 following the Crank-Nicholson scheme, we obtain:

$$
\begin{aligned}
u_{i,j,k,\epsilon}^{\tau+1} - u_{i,j,k,\epsilon}^{\tau} = \alpha_1 \left(u_{i+1,j,k,\epsilon}^{\tau} - 2u_{i,j,k,\epsilon}^{\tau} + u_{i-1,j,k,\epsilon}^{\tau} + u_{i+1,j,k,\epsilon}^{\tau+1} - 2u_{i,j,k,\epsilon}^{\tau+1} + u_{i-1,j,k,\epsilon}^{\tau+1} \right) + \\
+\alpha_2 \left(u_{i,j+1,k,\epsilon}^{\tau} - 2u_{i,j,k,\epsilon}^{\tau} + u_{i,j-1,k,\epsilon}^{\tau} + u_{i,j+1,k,\epsilon}^{\tau+1} - 2u_{i,j,k,\epsilon}^{\tau+1} + u_{i,j-1,k,\epsilon}^{\tau+1} \right) + \\
+\alpha_3 \left(u_{i,j,k+1,\epsilon}^{\tau} - 2u_{i,j,k,\epsilon}^{\tau} + u_{i,j,k-1,\epsilon}^{\tau} + u_{i,j,k+1,\epsilon}^{\tau+1} - 2u_{i,j,k,\epsilon}^{\tau+1} + u_{i,j,k-1,\epsilon}^{\tau+1} \right) + \\
+\alpha_4 b(E_\epsilon) \times \left(u_{i,j,k,\epsilon+1}^{\tau} - u_{i,j,k,\epsilon}^{\tau} + u_{i,j,k,\epsilon+1}^{\tau+1} - u_{i,j,k,\epsilon}^{\tau+1} \right) + \\
+\alpha_5 \left(u_{i,j,k,\epsilon}^{\tau} + u_{i,j,k,\epsilon}^{\tau+1} \right) 2E_\epsilon \left(U_B + U_{ph} \mathcal{F}_{KN}(E_\epsilon) \right) + \\
+\alpha_5 \left(u_{i,j,k,\epsilon}^{\tau} + u_{i,j,k,\epsilon}^{\tau+1} \right) U_{ph} \left(E_e^2 \mathcal{F}'_{KN}(E_\epsilon) + \Delta t S(x_i, y_j, z_k, E_\epsilon, t^{\tau+1/2} \right) .
\end{aligned}
\tag{23}
$$

It is important to notice that the pre-CN (α) coefficients are the same as in eq. 21 but divided by two. Rearranging terms, we get:

$$
\begin{aligned}
u_{i,j,k,\epsilon}^{\tau+1} \left(1 + 2\alpha_1 + 2\alpha_2 + 2\alpha_3 + \alpha_4 b(E_\epsilon) + \alpha_5 \left(U_B 2E_\epsilon + U_{ph} \Delta[bE^2] \right) \right) + \\
-\alpha_1 \left(u_{i-1,j,k,\epsilon}^{\tau+1} + u_{i+1,j,k,\epsilon}^{\tau+1} \right) - \alpha_2 \left(u_{i,j-1,k,\epsilon}^{\tau+1} + u_{i,j+1,k,\epsilon}^{\tau+1} \right) - \\
-\alpha_3 \left(u_{i,j,k-1,\epsilon}^{\tau+1} + u_{i,j,k+1,\epsilon}^{\tau+1} \right) - \alpha_4 b(E_\epsilon) u_{i,j,k,\epsilon+1}^{\tau+1} = \\
u_{i,j,k,\epsilon}^{\tau} \left(1 - 2\alpha_1 - 2\alpha_2 - 2\alpha_3 - b(E_\epsilon)\alpha_4 + \alpha_5 \left(U_B 2E_\epsilon + U_{ph} \Delta[bE^2] \right) \right) + \\
+\alpha_1 \left(u_{i+1,j,k,\epsilon}^{\tau} + u_{i-1,j,k,\epsilon}^{\tau} \right) + \alpha_2 \left(u_{i,j+1,k,\epsilon}^{\tau} + u_{i,j-1,k,\epsilon}^{\tau} \right) + \alpha_3 \left(u_{i,j,k+1,\epsilon}^{\tau} + u_{i,j,k-1,\epsilon}^{\tau} \right) + \\
+u_{i,j,k,\epsilon+1}^{\tau} b(E_\epsilon)\alpha_4 + \Delta t S(x_i, y_j, z_k, E_\epsilon, t^{\tau+1/2}) .
\end{aligned}
\tag{24}
$$

This, as in the case for the 2D solution of above, is impractical, so we need to solve it following the ADI scheme, which will leave us the next three equations:

$$
\begin{aligned}
u_{i,j,k,\epsilon}^{\tau+1/3} (1 + 2\alpha_3) - \alpha_3 \left(u_{i,j,k-1,\epsilon}^{\tau+1/3} + u_{i,j,k+1,\epsilon}^{\tau+1/3} \right) = \\
u_{i,j,k,\epsilon}^{\tau} \left(1 - 2\alpha_1 - 2\alpha_2 - b(E_\epsilon)\alpha_4 + \alpha_5 \left(U_B 2E_\epsilon + U_{ph} \Delta[bE^2] \right) \right) + \\
+\alpha_1 \left(u_{i+1,j,k,\epsilon}^{\tau} + u_{i-1,j,k,\epsilon}^{\tau} \right) + \alpha_2 \left(u_{i,j+1,k,\epsilon}^{\tau} + u_{i,j-1,k,\epsilon}^{\tau} \right) + \\
+u_{i,j,k,\epsilon+1}^{\tau} b(E_\epsilon)\alpha_4 + \Delta t S(x_i, y_j, z_k, E_\epsilon, t^{\tau+1/6}) .
\end{aligned}
\tag{25}
$$

$$
\begin{aligned}
u^{\tau+2/3}_{i,j,k,\epsilon}\left(1+2\alpha_2\right)-\alpha_2\left(u^{\tau+2/3}_{i,j+1,k,\epsilon}+u^{\tau+2/3}_{i,j-1,k,\epsilon}\right) = \\
u^{\tau+1/3}_{i,j,k,\epsilon}\left(1-2\alpha_1-2\alpha_3-b(E_\epsilon)\alpha_4+\alpha_5\left(U_B 2E_\epsilon+U_{ph}\Delta[bE^2]\right)\right)+ \\
+\alpha_1\left(u^{\tau+1/3}_{i+1,j,k,\epsilon}+u^{\tau+1/3}_{i-1,j,k,\epsilon}\right)+\alpha_3\left(u^{\tau+1/3}_{i,j,k-1,\epsilon}+u^{\tau+1/3}_{i,j,k+1,\epsilon}\right)+ \\
+u^{\tau+1/3}_{i,j,k,\epsilon+1}b(E_\epsilon)\alpha_4+\Delta t S(x_i,y_j,z_k,E_\epsilon,t^{\tau+1/2})\ .
\end{aligned}
\tag{26}
$$

$$
\begin{aligned}
u^{\tau+1}_{i,j,k,\epsilon}\left(1+2\alpha_1\right)-\alpha_1\left(u^{\tau+1}_{i+1,j,k,\epsilon}+u^{\tau+1}_{i-1,j,k,\epsilon}\right) = \\
u^{\tau+2/3}_{i,j,k,\epsilon}\left(1-2\alpha_1-2\alpha_3-b(E_\epsilon)\alpha_4+\alpha_5\left(U_B 2E_\epsilon+U_{ph}\Delta[bE^2]\right)\right)+ \\
+\alpha_2\left(u^{\tau+2/3}_{i,j+1,k,\epsilon}+u^{\tau+2/3}_{i,j-1,k,\epsilon}\right)+\alpha_3\left(u^{\tau+2/3}_{i,j,k-1,\epsilon}+u^{\tau+2/3}_{i,j,k+1,\epsilon}\right)+ \\
+u^{\tau+2/3}_{i,j,k,\epsilon+1}b(E_\epsilon)\alpha_4+\Delta t S(x_i,y_j,z_k,E_\epsilon,t^{\tau+1/2})\ .
\end{aligned}
\tag{27}
$$

At the end, these equations are solved in a matritial way, similar to eq. 16, but in a 3-step procedure. **Note that here we are using steps of $\Delta t/3$**, meaning that the coefficients $\alpha_1, \alpha_2, \alpha_3$ and α_5 are divided by 3, as well as Δt.

Bonus: The modified Peaceman-Rachford method

A similar ADI method is the Peaceman-Rachford method presented in [1]. Detailed information about the method can be found in the next thesis: `https://mountainscholar.org/bitstream/handle/11124/170299/Wray_mines_0052N_11045.pdf?sequence=1&isAllowed=y`. Basically, it consists of a procedure similar to the standard ADI, but with a different way of factorizing. For example, let's start from a simple a 2D diffusion equation, such as

$$
\frac{du}{dt}=\nabla^2_{r,z}u+S \longrightarrow \frac{u^{t+1}_{i,j}-u^t_{i,j}}{\Delta t}=\frac{1}{2}\left(\delta^2_r u^{t+1}_{i,j}+\delta^2_r u^t_{i,j}\right)+\left(\delta^2_z u^{t+1}_{i,j}+\delta^2_z u^t_{i,j}\right)+S^{t+1/2},
$$

where $\delta^2_z u^t_{i,j}=\frac{u^t_{i,j-1}-2u^t_{i,j}+u^t_{i,j+1}}{\Delta z^2}$ and $\delta^2_r u^t_{i,j}=\frac{u^t_{i-1,j}-2u^t_{i,j}+u^t_{i+1,j}}{\Delta r^2}$.

The solution of this equation, following the Peaceman-Rachford numerical method, must first be factorized and splitted into two fractional timesteps:

$$
\begin{aligned}
\left(1-\frac{\Delta t}{2}\delta^2_r\right)u^{t+1/2}_{i,j} &= \left(1+\frac{\Delta t}{2}\delta^2_z\right)u^t_{i,j}+\frac{\Delta t}{2}S^{t+1/2} \\
\left(1-\frac{\Delta t}{2}\delta^2_z\right)u^{t+1}_{i,j} &= \left(1+\frac{\Delta t}{2}\delta^2_r\right)u^{t+1/2}_{i,j}+\frac{\Delta t}{2}S^{t+1/2}
\end{aligned}
\tag{28}
$$

The simplest way to solve it is to use a three-steps scheme:

$$\begin{aligned}
V_{i,j}^{t} &= \left(1+\frac{\Delta t}{2}\delta_z^2\right)u_{i,j}^{t} \\
\left(1-\frac{\Delta t}{2}\delta_r^2\right)u_{i,j}^{t+1/2} &= V_{i,j}^{t}+\frac{\Delta t}{2}S^{t+1/2} \\
\left(1-\frac{\Delta t}{2}\delta_z^2\right)u_{i,j}^{t+1} &= 2u_{i,j}^{t+1/2}-V_{i,j}^{t}
\end{aligned} \tag{29}$$

Applied to the 2D case (eq. 8, excluding energy losses), the analogous to eq. 28 is:

$$\begin{aligned}
\left(1-\frac{\Delta t}{2}\delta_r^2-\frac{\Delta t}{2}\frac{1}{2r}\delta_r\right)u_{i,j}^{t+1/2} &= \left(1+\frac{\Delta t}{2}\delta_z^2\right)u_{i,j}^{t}+\frac{\Delta t}{2}S^{t+1/2} \\
\left(1-\frac{\Delta t}{2}\delta_z^2\right)u_{i,j}^{t+1} &= \left(1+\frac{\Delta t}{2}\delta_r^2+\frac{\Delta t}{2}\frac{1}{2r}\delta_r\right)u_{i,j}^{t+1/2}+\frac{\Delta t}{2}S^{t+1/2}
\end{aligned} \tag{30}$$

having that $\delta_r u_{i,j}^t = \frac{u_{i+1,j}^t+u_{i-1,j}^t}{\Delta r}$

This can be solved as:

$$\begin{aligned}
V_{i,j}^{t} &= \left(1+\frac{\Delta t}{2}\delta_z^2\right)u_{i,j}^{t} \\
\left(1-\frac{\Delta t}{2}\delta_r^2-\frac{\Delta t}{2}\frac{1}{2r}\delta_r\right)u_{i,j}^{t+1/2} &= V_{i,j}^{t}+\frac{\Delta t}{2}S^{t+1/2} \\
\left(1-\frac{\Delta t}{2}\delta_z^2\right)u_{i,j}^{t+1} &= 2u_{i,j}^{t+1/2}-V_{i,j}^{t}
\end{aligned} \tag{31}$$

Appendix

Discretization of first and second derivatives at order Δ_x^2

Here we explain two of the most common discretizated terms, but not so simple to see, which are employed in the discretizations that we need, using the finite difference method.

First, in the right-hand side of equation 4, the third term can not be so simple to get at a first look. The derivation is the next:

$$\frac{1}{r}D_r\frac{du}{dr} \longrightarrow \frac{D_r}{r}\times\left[\left(\frac{\Delta u}{\Delta r}\right)_{i+1/2} - \left(\frac{\Delta u}{\Delta r}\right)_{i-1/2}\right] = \frac{D_r}{r}\times\frac{1}{2}\frac{[(u_{i+1,k,\epsilon}-u_{i,k,\epsilon})+(u_{i,k,\epsilon}-u_{i-1,k,\epsilon})]}{\Delta r}$$

Taking the mid-grid points is needed to have the second order precision in the radial space (Δr^2).

Then, for the first and third terms in equation 4, the derivation is the next: From a Taylor expansion to second order in space, we have $u(x+\Delta x,t) = u(x,t) + \Delta x\frac{du}{dx} + \frac{1}{2}\Delta x^2\frac{d^2u}{dx^2} + \mathcal{O}(\Delta x^4)$ and $u(x-\Delta x,t) = u(x,t) - \Delta x\frac{du}{dx} + \frac{1}{2}\Delta x^2\frac{d^2u}{dx^2} + \mathcal{O}(\Delta x^4)$. Their sum gives that $u(x+\Delta x,t)+u(x-\Delta x,t) = 2u(x,t) + \Delta x^2\frac{d^2u}{dx^2}$. Rearranging this, we obtain:

$$\frac{d^2u}{dx^2} \approx \frac{u(x+\Delta x,t)+u(x-\Delta x,t)-2u(x,t)}{\Delta x^2} + \mathcal{O}(\Delta x^2).$$

As one can check, this result is the same as taking the derivative of the first derived equation.

Implicit formulas from matrix equations

Here we show the basics of the matrix multiplication operations for a formula just involving spatial diffusion. The starting point is Equations 11 and 12, which have the following form without considering energy losses:

$$\begin{aligned} u_{i,k,\epsilon}^{\tau+1/2}(1+2\alpha_3) - u_{i,k+1,\epsilon}^{\tau+1/2}\alpha_3 - u_{i,k-1,\epsilon}^{\tau+1/2}\alpha_3 = u_{i,k,\epsilon}^{\tau}(1-2\alpha_1) + \\ + u_{i+1,k,\epsilon}^{\tau}\left(\alpha_1+\frac{\alpha_2}{r_i}\right) + u_{i-1,k,\epsilon}^{\tau}\left(\alpha_1-\frac{\alpha_2}{r_i}\right) + \frac{\Delta t}{2}S(r_i,z_k,E_\epsilon,t^{\tau+1/4}) \end{aligned} \tag{32}$$

$$\begin{aligned} u_{i,k,\epsilon}^{\tau+1}(1+2\alpha_1) - u_{i+1,k,\epsilon}^{\tau+1}\left(\alpha_1+\frac{\alpha_2}{r_i}\right) - u_{i-1,k,\epsilon}^{\tau+1}\left(\alpha_1-\frac{\alpha_2}{r_i}\right) = \\ + u_{i,k,\epsilon}^{\tau+1/2}(1-2\alpha_3) + u_{i,k+1,\epsilon}^{\tau+1/2}\alpha_3 + u_{i,k-1,\epsilon}^{\tau+1/2}\alpha_3 + \frac{\Delta t}{2}S(r_i,z_k,E_\epsilon,t^{\tau+3/4}) \end{aligned} \tag{33}$$

This can be seen as an equation where we go on half time step applying a reshuffling implicitly in the k direction first, and then another half time step by reshuffling implicitly in the i direction. Each equation includes an explicit reshuffling in the other direction too, which means that each equation involves two matrix operations. The matrices involving operations in the k space are totally symmetric (which means that the rate of diffusion to the $k-1$ bin is the same as for the $k+1$ bin), while this does not happen for the i space, for which the rate of transport of particles is not the same going from a smaller ring (radius) to a bigger one, simply because the availability of more phase space where to go (cylindrical coordinates).

3 Conclusion

We have revised the main ways of solving numerically differential equations for physical systems of 2 and more dimensions. We tested the hypothesis of anisotropic diffusion where the diffusion in a direction is related to the diffusion in the other direction in an energy-independent way, just depending on one parameter: M_A. We have used the Peaceman-Rachford scheme, although for many dimensions it is easier the use of the Operator Splitting method. Another improvement of these equations would be to use a Range-Kutta method, for the discretization of the diffusion equation.

References

[1] Bernard Bialecki and Ryan I. Fernandes. Orthogonal spline collocation laplace-modified and alternating-direction methods for parabolic problems on rectangles. *Mathematics of Computation*, 60(202):545–573, 1993.

[2] Ruo-Yu Liu, Huirong Yan, and Heshou Zhang. Understanding the multiwavelength observation of geminga's tev halo: The role of anisotropic diffusion of particles. *Phys. Rev. Lett.*, 123:221103, Nov 2019.