Solving the modified diffusion equation using the Crank-Nicolson Algorithm

Rui Xu

June 29, 2015

The modified Diffusion Equation

We wish to solve a modified diffusion equation in Cartesian, cylindrical, and spherical coordinates using the Crank-Nicolson algorithm. We assume y and z symmetry in Cartesian coordinates, azimuthal and vertical symmetry in cylindrical coordinates, and azimuthal and polar symmetry in spherical coordinates. This makes for a one-dimensional computational box in the '$\mathbf{x}$' direction for the Cartesian coordinate system, and one-dimensional computational boxes in the radial '$\mathbf{r}$' direction for the cylindrical and spherical coordinate systems. For the purpose of this derivation, $\mathbf{x} = \mathbf{r}$, and we use $\mathbf{r}$ as our generalized coordinate.

We are solving the chain propagator for the continuous Gaussian chain representation of a polymer, where $q(\mathbf{r}, s)$ is an end-integrated chain propagator and 's' is an index that runs along the length of the chain. The modified diffusion equation has the form:

$$\frac{\partial q(\mathbf{r}; s)}{\partial s} = C\nabla^2 q(\mathbf{r}; s) - \omega(\mathbf{r})q(\mathbf{r}; s) \tag{1}$$

where C is the diffusion coefficient that represents R_g^2, the radius of gyration of the polymer squared, or $b^2/6$, where b is the persistence length of the polymer. The ω-field represents an auxiliary field coupled to polymer density.

We solve the modified diffusion equation in three coordinate systems, each with a different Laplacian. The Laplacian in Cartesian coordinates, with y and z symmetry and replacing $\mathbf{x}$ with $\mathbf{r}$ is:

$$\nabla^2 = \frac{\partial^2}{\partial r^2} \tag{2}$$

The Laplacian in cylindrical coordinates, assuming azimuthal and vertical symmetry, has the form:

$$\nabla^2 = \frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial}{\partial r}\right) = \frac{\partial^2}{\partial r^2} + \frac{1}{r}\frac{\partial}{\partial r} \tag{3}$$

The Laplacian in spherical coordinates, assuming azimuthal and polar symmetry, has the form:

$$\nabla^2 = \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial}{\partial r}\right) = \frac{\partial^2}{\partial r^2} + \frac{2}{r}\frac{\partial}{\partial r} \tag{4}$$

Instead of deriving the Crank-Nicolson algorithm separately for each coordinate system, we use a generalized Laplacian:

$$\nabla^2 = \frac{\partial^2}{\partial r^2} + \frac{(\mathbf{D}-1)}{r}\frac{\partial}{\partial r} \tag{5}$$

where $\mathbf{D} = 1$ for Cartesian coordinates, $\mathbf{D} = 2$ for cylindrical coordinates, and $\mathbf{D} = 3$ for spherical coordinates. Inserting Equation 5 into the modified diffusion equation gives:

$$\frac{\partial q(\mathbf{r};s)}{\partial s} = C\left(\frac{\partial^2}{\partial r^2} + \frac{(\mathbf{D}-1)}{r}\frac{\partial}{\partial r}\right) q(\mathbf{r};s) - \omega(\mathbf{r})q(\mathbf{r};s) \tag{6}$$

We define a computational grid of discrete points at which the continuous function $q(\mathbf{r};s)$ is sampled. We define the computational grid as having length L_r where $\mathbf{r} \in [0, L_r]$. We use N_r equally spaced points:

$$r_i = i\Delta r, \quad i = 0...N_r - 1 \tag{7}$$

for $\Delta r = L_r/(N_r - 1)$, the chosen grid spacing in the r-direction. We also discretize the continous Gaussian chain over the interval $s \in [0, N]$ using N_s equally spaced points:

$$s_n = n\Delta s, \quad n = 0...N_s - 1 \tag{8}$$

for $\Delta s = N/(N_s - 1)$, the contour step along $q(r, z; s)$. We now change our notation for the end-integrated chain propagator from its continuous form, $q(\mathbf{r};s)$, to its discretized form, q_i^n which we will use for the remainder of the derivation. As an initial condition, we set:

$$q_i^0 = 1 \quad \text{for} \quad i \in [0, N_r - 1] \tag{9}$$

The Crank-Nicolson algorithm is an implicit method that consists of forward Euler difference approximation in the $\mathbf{s}$ domain, and a combination of the forward Euler method at the n^{th} monomer, and the backward Euler method at the $n+1$ monomer. In the following section, we implement the Crank-Nicolson algorithm for our modified diffusion equation.

Crank-Nicolson

The next step is to implement a finite differencing method to the modified diffusion equation. For the s-derivative, we use a forward Euler difference approximation:

$$\frac{\partial q_i^n}{\partial s} \Rightarrow \frac{q_i^{n+1} - q_i^n}{\Delta s} \tag{10}$$

For the first order r-derivative, we use the following difference approximation:

$$\frac{\partial q_i^n}{\partial r} \Rightarrow \frac{1}{2}\left[\left(\frac{q_{i+1}^{n+1} - q_{i-1}^{n+1}}{2(\Delta r)}\right) + \left(\frac{q_{i+1}^n - q_{i-1}^n}{2(\Delta r)}\right)\right] \tag{11}$$

For the second order r-derivative, we use the following central difference approximation:

$$\frac{\partial^2 q_i^n}{\partial r^2} \Rightarrow \frac{1}{2(\Delta r)^2}\Big[\left(q_{i+1}^{n+1} - 2q_i^{n+1} + q_{i-1}^{n+1}\right) + \left(q_{i+1}^{n} - 2q_i^{n} + q_{i-1}^{n}\right)\Big] \tag{12}$$

After inserting the difference approximations into the modified diffusion equation, the modified diffusion equation has the form:

$$\frac{1}{\Delta s}\left(q_i^{n+1} - q_i^n\right) = \frac{C}{2(\Delta r)^2}\Big[\left(q_{i+1}^{n+1} - 2q_i^{n+1} + q_{i-1}^{n+1}\right) + \left(q_{i+1}^{n} - 2q_i^{n} + q_{i-1}^{n}\right)\Big] \tag{13}$$

$$+ \frac{C(\mathbf{D}-1)}{2r}\left[\left(\frac{q_{i+1}^{n+1} - q_{i-1}^{n+1}}{2(\Delta r)}\right) + \left(\frac{q_{i+1}^{n} - q_{i-1}^{n}}{2(\Delta r)}\right)\right] \tag{14}$$

$$- \frac{\omega_i}{2}\left(q_i^{n+1} + q_i^n\right) \tag{15}$$

The next step is to separate the $n+1$ terms from the n terms:

$$\frac{1}{\Delta s}q_i^{n+1} - \frac{C}{2(\Delta r)^2}\left(q_{i+1}^{n+1} - 2q_i^{n+1} + q_{i-1}^{n+1}\right) - \frac{C(\mathbf{D}-1)}{4r\Delta r}\left(q_{i+1}^{n+1} - q_{i-1}^{n+1}\right) + \frac{\omega_i}{2}q_i^{n+1} \tag{16}$$

$$= \frac{1}{\Delta s}q_i^{n} + \frac{C}{2(\Delta r)^2}\left(q_{i+1}^{n} - 2q_i^{n} + q_{i-1}^{n}\right) + \frac{C(\mathbf{D}-1)}{4r\Delta r}\left(q_{i+1}^{n} - q_{i-1}^{n}\right) - \frac{\omega_i}{2}q_i^{n} \tag{17}$$

The next step is to group the coefficients by their coordinates $(i+1, i, i-1)$. We multiply the entire expression by Δs, set C to 1, and use functions $\alpha_{+1,0,-1}$ and $\beta_{+1,0,-1}$ to simplify the expression:

$$\alpha_{+1}q_{i+1}^{n+1} + \alpha_0 q_i^{n+1} + \alpha_{-1}q_{i-1}^{n+1} = \beta_{+1}q_{i+1}^{n+1} + \beta_0 q_{i,j}^{n+1} + \beta_{-1}q_{i-1}^{n+1} \tag{18}$$

where:

$$\alpha_{+1} \equiv -\frac{(\Delta s)}{2(\Delta r)^2} - \frac{(\mathbf{D\text{-}1})(\Delta s)}{4r(\Delta r)} \tag{19}$$

$$\alpha_0 \equiv 1 + \frac{(\Delta s)}{(\Delta r)^2} + \frac{(\Delta s)}{2}\omega_i \tag{20}$$

$$\alpha_{-1} \equiv -\frac{(\Delta s)}{2(\Delta r)^2} + \frac{(\mathbf{D\text{-}1})(\Delta s)}{4r(\Delta r)} \tag{21}$$

$$\beta_{+1} \equiv \frac{(\Delta s)}{2(\Delta r)^2} + \frac{(\mathbf{D\text{-}1})(\Delta s)}{4r(\Delta r)} \tag{22}$$

$$\beta_0 \equiv 1 - \frac{(\Delta s)}{(\Delta z)^2} - \frac{(\Delta s)}{2}\omega_i \tag{23}$$

$$\beta_{-1} \equiv \frac{(\Delta s)}{2(\Delta r)^2} - \frac{(\mathbf{D\text{-}1})(\Delta s)}{4r(\Delta r)} \tag{24}$$

We implement a zero derivative boundary condition (Neumann boundary condition). The mathematical form of this boundary condition is the following:

$$\frac{\partial q_0}{\partial r} = \frac{\partial q_{N_r-1}}{\partial r} = 0 \tag{25}$$

This requires that $q_1 = q_{-1}$, and $q_{N_r-2} = q_{N_r,j}$. The matrix form of the modified diffusion equation is the following:

$$\begin{bmatrix} q_0^{n+1} \\ q_1^{n+1} \\ \vdots \\ q_{N_r-2}^{n+1} \\ q_{Nr-1}^{n+1} \end{bmatrix} = \begin{bmatrix} \alpha_0 & (\alpha_{+1}+\alpha_{-1}) & 0 & \cdots & 0 \\ \alpha_{-1} & \alpha_0 & \alpha_{+1} & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \alpha_{-1} & \alpha_0 & \alpha_1 \\ 0 & \cdots & 0 & (\alpha_1+\alpha_{-1}) & \alpha_0 \end{bmatrix}^{-1} \begin{bmatrix} (\beta_{+1}+\beta_{-1})q_1^n + \beta_0 q_0^n \\ \beta_{+1}q_2^n + \beta_0 q_1^n + \beta_{-1}q_0^n \\ \vdots \\ \beta_{+1}q_{Nr-1}^n + \beta_0 q_{Nr-2}^n + \beta_{-1}q_{Nr-3}^n \\ (\beta_{+1}+\beta_{-1})q_{Nr-2}^n + \beta_0 q_{Nr-1}^n \end{bmatrix}$$

This matrix is tridiagonal, which means that we can solve this matrix algebra problem with the Tridiagonal Matrix Algorithm (TDMA). This algorithm scales as $O(N)$, significantly better than Gaussian elimination. We solve the diffusion equation by propagating the solution from $n = 0$ to $n = N_s - 1$.