\documentclass[11pt]{article} % use larger type; default would be 10pt

\usepackage{mathrsfs,amsmath}

\usepackage{gensymb}

\usepackage{graphicx}

\usepackage{float}

\graphicspath{ {images/} }

\usepackage{fixltx2e} %for subscript

\usepackage[margin=0.8in]{geometry}

\setlength{\parskip}{\baselineskip}%

\setlength{\voffset}{0in}

%\pagenumbering{gobble}

\title{Solving the modified diffusion equation using the Crank-Nicolson Algorithm}

\author{Rui Xu}

%\date{} % Activate to display a given date or no date (if empty),

% otherwise the current date is printed

\begin{document}

\maketitle

**\section\*{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 `$\textbf{x}$' direction for the Cartesian coordinate system, and one-dimensional computational boxes in the radial `$\textbf{r}$' direction for the cylindrical and spherical coordinate systems. For the purpose of this derivation, $\textbf{x}=\textbf{r}$, and we use $\textbf{r}$ as our generalized coordinate.

We are solving the chain propagator for the continuous Gaussian chain representation of a polymer, where $q(\textbf{r}, s)$ 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:

\begin{equation}

\frac{\partial q(\textbf{r}; s)}{\partial s} = C \nabla^2 q(\textbf{r}; s) - \omega (\textbf{r}) q(\textbf{r}; s)

\end{equation}

\noindent

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 $\omega$-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 $\textbf{x}$ with $\textbf{r}$ is:

\begin{equation}

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

\end{equation}

\noindent

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

\begin{equation}

\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}

\end{equation}

\noindent

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

\begin{equation}

\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}

\end{equation}

\noindent

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

\begin{equation}

\nabla^2=\frac{\partial^2}{\partial r^2} + \frac{\left(\textbf{D}-1\right)}{r}\frac{\partial}{\partial r}

\end{equation}

\noindent

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

\begin{equation}

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

\end{equation}\\[12pt]

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

\begin{equation}

r\_i = i \Delta r, \quad i = 0 ... N\_r - 1

\end{equation}

\noindent

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

\begin{equation}

s\_n = n \Delta s, \quad n = 0 ... N\_s -1

\end{equation}

\noindent

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, $q(\textbf{r};s)$, to its discretized form, $q\_{i}^n$ which we will use for the remainder of the derivation. As an initial condition, we set:

\begin{equation}

q\_{i}^0 = 1 \quad \text{for} \quad i \in [0,N\_r-1]

\end{equation}

The Crank-Nicolson algorithm is an implicit method that consists of forward Euler difference approximation in the $\textbf{s}$ domain, and a combination of the forward Euler method at the $n^{\text{th}}$ 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.

**\section\*{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:

\begin{equation}

\frac{\partial q\_i^n}{\partial s} \Rightarrow \dfrac{q\_{i}^{n+1} - q\_{i}^n}{\Delta s}

\end{equation}

\noindent

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

\begin{equation}

\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]

\end{equation}

\noindent

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

\begin{equation}

\frac{\partial^2 q\_i^n}{\partial r^2} \Rightarrow \frac{1}{2(\Delta r)^2} \textbf{\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) \textbf{\Big]}

\end{equation}

\noindent

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

\begin{align}

\frac{1}{\Delta s} \left(q\_{i}^{n+1} - q\_{i}^n \right) &= \frac{C}{2(\Delta r)^2} \textbf{\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) \textbf{\Big]} \\

&+ \frac{C (\textbf{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] \\

&-\frac{\omega\_i}{2} \left(q\_i^{n+1}+q\_i^n \right)

\end{align}

\noindent

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

\begin{align}

&\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(\textbf{D}-1)}{2r\Delta r} \left( q\_{i+1}^{n+1} - q\_{i-1}^{n+1} \right) + \frac{\omega\_{i}}{2} q\_{i}^{n+1} \\

& = \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(\textbf{D}-1)}{2r\Delta r} \left( q\_{i+1}^{n} - q\_{i-1}^{n} \right) \frac{1}{2} - \frac{\omega\_{i}}{2} q\_{i}^{n}

\end{align}

\noindent

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

\begin{equation}

\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}

\end{equation}

\noindent

where:

\begin{align}

\alpha\_{+1} &\equiv - \dfrac{ (\Delta s)}{2(\Delta r)^2} - \dfrac{(\textbf{D-1}) (\Delta s)}{2r(\Delta r)} \\

\alpha\_0 &\equiv 1 + \dfrac{(\Delta s)}{(\Delta r)^2} + \dfrac{(\Delta s)}{2} \omega\_{i} \\

\alpha\_{-1} &\equiv - \dfrac{ (\Delta s)}{2(\Delta r)^2} + \dfrac{(\textbf{D-1}) (\Delta s)}{2r(\Delta r)} \\

\beta\_{+1} &\equiv \dfrac{ (\Delta s)}{2(\Delta r)^2} + \dfrac{(\textbf{D-1}) (\Delta s)}{2r(\Delta r)} \\

\beta\_0 &\equiv 1 - \dfrac{ (\Delta s)}{(\Delta z)^2} - \dfrac{(\Delta s)}{2} \omega\_{i} \\

\beta\_{-1} &\equiv \dfrac{ (\Delta s)}{2(\Delta r)^2} - \dfrac{(\textbf{D-1}) (\Delta s)}{2r(\Delta r)}

\end{align} \\[12pt]

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

\noindent

\begin{equation}

\dfrac{\partial q\_{0}}{\partial r} = \dfrac{\partial q\_{N\_r-1}}{\partial r} = 0

\end{equation}

\noindent

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} \\[0.5em]

q\_{1}^{n+1} \\[0.5em]

\vdots \\[0.5em]

q\_{N\_r-2}^{n+1} \\[0.5em]

q\_{Nr-1}^{n+1} \\[0.5em]

\end{bmatrix} =

%

\begin{bmatrix}

\alpha\_0 & (\alpha\_{+1}+\alpha\_{-1}) & 0 & \cdots & 0 \\[0.5em]

\alpha\_{-1} & \alpha\_0 & \alpha\_{+1} &\cdots &0 \\[0.5em]

\vdots & \ddots & \ddots & \ddots & \vdots \\[0.5em]

0 & \cdots & \alpha\_{-1} & \alpha\_0 & \alpha\_1 \\[0.5em]

0 & \cdots & 0 & (\alpha\_1+\alpha\_{-1}) & \alpha\_0 \\[0.5em]

\end{bmatrix}^{-1}

%

\begin{bmatrix}

(\beta\_{+1}+\beta\_{-1})q\_{1}^{n} + \beta\_0q\_{0}^{n} \\[0.5em]

\beta\_{+1}q\_2^n +\beta\_0 q\_{1}^{n} + \beta\_{-1}q\_0^n \\[0.5em]

\vdots \\[0.5em]

\beta\_{+1}q\_{Nr-1}^n +\beta\_0 q\_{Nr-2}^{n} + \beta\_{-1}q\_{Nr-3}^n \\[0.5em]

(\beta\_{+1}+\beta\_{-1})q\_{Nr-2}^{n} + \beta\_0q\_{Nr-1}^{n} \\[0.5em]

\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 $n=N\_s-1$.

\end{document}