Discrete Fourier Transform

E.H.F.

May 16, 2005

1 Introduction

We want to perform a one dimensional self-consistent field theory calculation for a diblock copolymer. We'll produce all the code from scratch. These notes will assume that you know how to program in a language that can multiply complex numbers.

2 Mean-Field Equations

For a diblock copolymer, we want to find fields $E(x)$ and $P(x)$ that satisfy the mean-field equations

$$\phi_A(x;[E,P]) + \phi_B(x;[E,P]) - 1 = 0 \tag{1}$$

$$\frac{2}{\chi N}E(x) - \phi_A(x;[E,P]) + \phi_B(x;[E,P]) = 0 \tag{2}$$

where the volume fractions

$$\phi_A(x;[E,P]) = \frac{1}{Q[E,P]}\int_0^f q(x,t;[E,P])q^\dagger(x,1-t;[E,P])dt \tag{3}$$

$$\phi_B(x;[E,P]) = \frac{1}{Q[E,P]}\int_f^1 q(x,t;[E,P])q^\dagger(x,1-t;[E,P])dt \tag{4}$$

depend on a propagator $q(x,t;[E,P])$ that satisfies

$$\frac{\partial}{\partial t}q(x,t) = \frac{\partial^2}{\partial x^2}q(x,t) - \psi(x,t)q(x,t) \tag{5}$$

with the initial condition $q(x,0) = 1$. The contour dependent field is $\psi(x,t) = P(x) - E(x)$ for $t \in [0, f)$ and $\psi(x,t) = P(x) + E(x)$ for $t \in [f, 1]$. $q^\dagger(x,t)$ also satisfies the diffusion equation but initial condition $q^\dagger(x,1) = 1$. And also

$$Q[E,P] = L^{-1} \int q(x,1)dx \tag{6}$$

where L is the length of our periodic domain in units of R_g. I'm not going to derive these equations; we're just going to solve them numerically.

Formally, we treat the diffusion equation as a first order linear equation and write the solution

$$q(x,t) = e^{\mathcal{L}t} q(x,0) \tag{7}$$

where the operator is

$$\mathcal{L} \equiv \frac{\partial^2}{\partial x^2} - \psi(x,t) \tag{8}$$

To solve the diffusion equation numerically, we pick a small time interval Δt and iterate:

$$q(x, \Delta t) = e^{\mathcal{L}\Delta t} q(x,0) \tag{9}$$

If the operator $\mathcal{L}$ only consisted of the field $\psi(x,t)$, then we simply multiply $q(x,0) = 1$ by $e^{\psi(x,t)\Delta t}$. If we represent the spatial domain $x \in [0, L]$ as a collection of n points, then this requires n multiplications. Not too bad. It turns out that it's not as easy to apply the operation $e^{(\partial^2/\partial x^2)\Delta t}$, meaning that we can't do it in n multiplications in real space. But we can do it in n multiplications in Fourier space. That's why we're going to spend some time talking about discrete Fourier series.

3 Discrete Fourier Series

The spatial domain we want to solve this diffusion equation is $[0, L]$. Numbers in a computer are *always* dimensionless. To represent this spatial domain in a computer, we discretize it.

$$x_j = \frac{jL}{n} \tag{10}$$

for $j = 0, 1, \ldots, n-1$. Consider a function $f(x)$ on this interval; on the discrete points, we define $f_j \equiv f(x_j)$. We're interested in periodic function, so $f_j = f_{j+n}$.

The main point of this section is that we can represent any discrete periodic function as a sum over basis functions:

$$f_j = n^{-1/2} \sum_{k=0}^{n-1} \hat{f}_k e^{i((2\pi/n)k)j} \tag{11}$$

where $i = \sqrt{-1}$. If $\alpha_k = 2\pi k/n$, then the basis functions are $n^{-1/2}e^{i\alpha_k j}$ for $k = 0, 1, \ldots, n-1$ where $n^{-1/2}$ is a normalization factor. In this basis function, j denotes the position in space, and k indexes the basis functions. These basis functions are orthonormal. To show this, the inner product of two basis functions is

$$n^{-1} \sum_{j=0}^{n-1} e^{i\alpha_k j} e^{-i\alpha_l j} = n^{-1} \sum_{j=0}^{n-1} e^{i(2\pi/n)(k-l)j} \tag{12}$$

For $k = l$, each exponential is 1, so then the sum is 1. For $k \neq l$, the sum over the complex exponentials amounts to

$$J_m \equiv \sum_{j=0}^{n-1} e^{i\alpha_m j} \tag{13}$$

for $m \neq 0$. (think $m = k - l$) Orthonormality follows from $J_m = 0$ for $m \neq 0$. The trick to showing this is multiplying by $e^{i(2\pi/n)m}$:

$$J_m e^{i(2\pi/n)m} = \sum_{j=0}^{n-1} e^{i(2\pi/n)mj} e^{i(2\pi/n)m} = \sum_{l=1}^{n} e^{i(2\pi/n)ml} = \sum_{l=0}^{n-1} e^{i(2\pi/n)ml} = J_m \tag{14}$$

To move from the second to third expression, make the substitution $l = j+1$. To move from the third to fourth expression, note that $e^{i\alpha_m n} = 1$ using the periodicity of the basis functions. For $m \neq 0$, $e^{i(2\pi/n)m} \neq 0$ from which it follows that $J_m = 0$.

Neat, eh? Now we can claim that

$$n^{-1} \sum_{j=0}^{n-1} e^{i\alpha_k j} e^{-i\alpha_l j} = \delta_{kl} \tag{15}$$

where δ_{kl} is the Kronecker delta function. Now we can use this orthogonality and eq. 11 to claim that

$$\hat{f}_k = n^{-1/2} \sum_{j=0}^{n-1} f_j e^{-i\alpha_k j} \tag{16}$$

This is a discrete Fourier transform. You can think of eq. 11 as the discrete inverse Fourier transform. Your homework is to implement these in software. Write a program that accepts an array of complex numbers and an integer that represents the length of the array and outputs the discrete fourier transform of the array. It will probably be easiest to forget about the factor of $n^{-1/2}$.

To develop the algorithm for solving the diffusion equation in the next section, we'll need the following notation

$$F_{kj} = n^{-1/2} e^{i(2\pi/n)kj} \tag{17}$$

so then one can write the fourier transform of a discrete function as

$$\hat{f}_k = \sum_{j=0}^{n-1} F_{kj} f_j. \tag{18}$$

It is also useful to define the conjugate transpose of the previous matrix.

$$F^\dagger_{kj} = n^{-1/2} e^{-i(2\pi/n)kj} \tag{19}$$

Since F_{kj} is symmetric, $F^\dagger_{kj}$ results from simply taking the complex conjugate of each entry of F_{kj}. The orthonormality of the basis function gives that this matrix is the inverse of F_{kj}:

$$\sum_{k=0}^{n-1} F^\dagger_{lk} F_{kj} = \delta_{lj} \tag{20}$$

so then F_{kj} is a unitary matrix. This is useful since it provides a resolution of the identity operator δ_{lj}.