Efficient Order-Adaptive Methods for Polymer Self-Consistent Field Theory

Hector D. Ceniceros

Department of Mathematics, University of California, Santa Barbara, CA 93110, U.S.A.

Abstract

A highly accurate and memory-efficient approach for the solution of polymer self-consistent field theory (SCFT) is proposed. The central idea is to combine spectral integration in the polymer chain contour variable with a spectral deferred correction technique to solve the SCFT modified diffusion equations with arbitrarily high order of accuracy. The result is a robust method that achieves high accuracy with a minimal number of discrete contour nodes, which translates into vastly reduced memory requirements and increased computational efficiency. In particular, this spectral deferred correction method enables the computation of strongly segregated systems with unprecedented accuracy. Moreover, the framework of deferred corrections allows us to adaptively increase the order of accuracy during the outer saddle point iteration to drastically reduce the cost of a SCFT computation.

Keywords: spectral deferred correction, Clenshaw-Curtis, Chebyshev nodes, spectral integration, mean field approximation

1. Introduction

Self consistent field theory (SCFT) or mean field theory approximation has been a powerful tool to investigate and discover polymer phases (see for example [11]). Computationally, polymer SCFT amounts to three problems: 1) the solution of one or several Fokker-Planck or modified diffusion equations (MDE's), 2) the computation of nonlocal, volume fraction operators, and 3) finding saddle points for the effective Hamiltonian. The third problem is solved through an iterative method, typically gradient descent-ascent or a combination of this and the conjugate gradient method [16], and each iteration requires the solution of problems 1) and 2). The latter are $d+1$ dimensional problems (d being the spatial dimension) as conformational information along the polymer chains is needed in addition to the spatial variables. This makes polymer SCFT computationally expensive and memory demanding.

In this work, we propose a numerical approach that significantly reduces the cost of polymer SCFT computations and cuts down the memory requirements by an order of magnitude with respect to existing methods. The central idea is to use spectral integration along

Email address: ceniceros@ucsb.edu (Hector D. Ceniceros)

the contour polymer chain variable s both in the solution of the MDE's and the computation of the volume fraction operators to drastically reduce the number of nodes in s for a given high accuracy. This is achieved with the use of Chebyshev (Gauss-Lobatto) nodes, Clenshaw-Curtis type quadratures and spectral deferred corrections.

While we focus here on problems 1) and 2), we also propose a strategy to further reduce the cost of the saddle point iterations, problem 3). By adaptively varying the order of accuracy in s to solve 1), without changing the resolution (i.e. without increasing memory), we produce a hierarchy of increasingly more accurate initial guesses for the saddle point iteration.

The rest of the paper is organized as follows. The diblock copolymer model, which is used as a test bed problem for the proposed methodology is summarized in Section 2. This is followed by a brief Section 3 on the idea of spectral contour chain integration. Section 4 is devoted to the numerical solution of the MDE's and in particular to high and arbitrary order methods in s. Important observations on the asymptotic behavior at small scales (high wave numbers) of some methods for the MDE's, including some commonly used schemes, are also made. The contour spectral approach is integrated in the SCFT framework in Section 5 while Section 6 is devoted to the faster adaptive order SCFT iterations. Concluding remarks are made in Section 7 and a detailed derivation and formulas for the spectral integration is provided in Appendix A.

2. The Diblock Copolymer Model

We take an incompressible melt of flexible AB diblock copolymers as our prototype SCFT model to discuss and test the proposed new numerical approach.

We assume for simplicity the same statistical segment length of the two blocks in the diblock chain, $b_A = b_B = b$, and employ a Flory parameter χ to describe the strength binary contacts between A and B. The free energy can be written as [11]

$$H[\mu_A, \mu_B] = \int d\mathbf{r}\ [-f\mu_A - (1-f)\mu_B + (\mu_A - \mu_B)^2/(4\chi N)] - V \ln Q_c[\mu_A, \mu_B], \qquad (1)$$

where V is the system volume, N is the copolymer degree of polymerization, f is the average volume fraction of type A blocks. $Q_c[\mu_A, \mu_B]$ is the partition function for a single copolymer experiencing chemical potentials μ_A and μ_B that exert forces, respectively, on the A and B blocks. This single chain partition function is given by

$$Q_c[\mu_A, \mu_B] = \frac{1}{V} \int d\mathbf{r}\ q(\mathbf{r}, 1; [\mu_A, \mu_B]), \qquad (2)$$

where the copolymer propagator $q[\mathbf{r}, s; \mu_a, \mu_B]$ satisfies the Fokker-Planck or modified diffusion equation (MDE)

$$\frac{\partial q}{\partial s} = \nabla^2 q - \psi q, \quad q(\mathbf{r}, 0; [\mu_A, \mu_B]) = 1. \qquad (3)$$

Here ψ is the potential acting on each block:

$$\psi(\mathbf{r}, s) = \begin{cases} \mu_A(\mathbf{r}), & 0 \le s \le f, \\ \mu_B(\mathbf{r}), & f < s \le 1. \end{cases} \tag{4}$$

The SCFT problem for this model is to find saddle points in which $H[\mu_A, \mu_B]$ is a minimum with respect to the exchange potential

$$\mu_-(\mathbf{r}) \equiv \frac{1}{2}[\mu_B(\mathbf{r}) - \mu_A(\mathbf{r})] \tag{5}$$

and a maximum with respect to the pressure

$$\mu_+(\mathbf{r}) \equiv \frac{1}{2}[\mu_A(\mathbf{r}) + \mu_B(\mathbf{r})]. \tag{6}$$

The first variation of H with respect to these fields can be written in terms of the local volume fractions ϕ_A and ϕ_B

$$\frac{\delta H[\mu_+, \mu_-]}{\delta \mu_+(\mathbf{r})} = \phi_A(\mathbf{r}; [\mu_+, \mu_-]) + \phi_B(\mathbf{r}; [\mu_+, \mu_-]) - 1, \tag{7}$$

$$\frac{\delta H[\mu_+, \mu_-]}{\delta \mu_-(\mathbf{r})} = (2f - 1) + \frac{2}{\chi N}\mu_-(\mathbf{r}) + \phi_B(\mathbf{r}; [\mu_+, \mu_-]) - \phi_A(\mathbf{r}; [\mu_+, \mu_-]). \tag{8}$$

The local volume fraction operators ϕ_A and ϕ_B can be computed from the Feynman-Kac formulas

$$\phi_A(\mathbf{r}; [\mu_+, \mu_-]) = \frac{1}{Q_c[\mu_+, \mu_-]} \int_0^f ds\; q(\mathbf{r}, s; [\mu_+, \mu_-]) q^\dagger(\mathbf{r}, 1 - s; [\mu_+, \mu_-]), \tag{9}$$

$$\phi_B(\mathbf{r}; [\mu_+, \mu_-]) = \frac{1}{Q_c[\mu_+, \mu_-]} \int_f^1 ds\; q(\mathbf{r}, s; [\mu_+, \mu_-]) q^\dagger(\mathbf{r}, 1 - s; [\mu_+, \mu_-]). \tag{10}$$

The new propagator $q^\dagger$ expresses the lack of head-to-tail symmetry of a diblock copolymer, and satisfies the following MDE:

$$\frac{\partial q^\dagger}{\partial s} = \nabla^2 q^\dagger - \psi^\dagger q^\dagger, \quad q^\dagger(\mathbf{r}, 0; [\mu_+, \mu_-]) = 1, \tag{11}$$

with

$$\psi^\dagger(\mathbf{r}, s) = \begin{cases} \mu_B(\mathbf{r}), & 0 \le s \le 1 - f, \\ \mu_A(\mathbf{r}), & 1 - f < s \le 1. \end{cases} \tag{12}$$

3. High Order Contour Chain Integration

At the core of the iteration to find a saddle point is the evaluation of the local volume fractions ϕ_A and ϕ_B, given by the integrals (9) and (10). To date, a popular quadrature

to obtain approximations for these operators has been Simpson's rule using equally-spaced points, Δs apart, along the chain contour variable s. This yields a fourth order approximation in Δs, assuming the propagators q and $q^\dagger$ are computed with at least that accuracy. Note that the integrals (9) and (10) have to be computed at every node $\mathbf{r}$ of the spatial grid. Thus, these operations are as costly as solving the MDE's and require considerable memory because values of q and $q^\dagger$ are needed at every point of $d+1$ grid (d being here the spatial dimension).

A spectral quadrature, such as a Gaussian or a Chebychev-node interpolatory quadrature, gives a desired high accuracy with a largely reduced number of nodes, relative to a fixed order quadrature, when the integrand is smooth. This would immediately reduce the memory requirements substantially and could potentially lower also the computational cost of a SCFT simulation. This is the central idea of this work.

The link of the Chebychev-node based Clenshaw-Curtis quadrature to the discrete cosine transform (DCT) [12, 13] makes this quadrature computationally very efficient and competitive with the Gaussian quadrature as pointed out in [20]. Moreover, the Chebychev nodes, unlike the Gaussian nodes, include the end points of the interval of integration and this is of relevance in SCFT because of the initial value problems (the MDE's) that have to be solved to generate the integrands. But to take advantage of this quadrature's spectral accuracy and consequently to achieve high accuracy with a minimal number of contour points, we need highly accurate and stable methods for the MDE's. Furthermore, these methods have also to be stable to the outer saddle point iteration. This is a subtle but crucial point in the design of an effective SCFT method as we discuss in detail below.

We consider next the problem of solving the MDE's with the goal of constructing robust and efficient high order methods for the SCFT saddle point iteration.

4. Solving the MDE's

In this section we take a closer look at the problem of solving the MDE's of SCFT. Due to potential discontinuities at the block junctions and at $s = 0$, the MDE's should be solved block by block. Thus, it is sufficient to consider the problem

$$\begin{aligned} \frac{\partial q}{\partial s} &= \nabla^2 q - wq, \quad 0 < s \le f, \\ q(\mathbf{r}, 0) &= 1, \end{aligned} \tag{13}$$

where w is a given field. For concreteness we take $f = 1/2$ and restrict ourselves to the one-dimensional problem ($d = 1$). Periodic boundary conditions are used as it is common in SCFT computations. The Laplacian is approximated spectrally with the discrete Fourier transform (DFT) using the FFT. We solve (13) on an interval of length $L = 10$.

4.1. Second Order Methods

Rasmussen and Kalosakas [18] proposed a Strang splitting [19] method that has become popular in polymer SCFT computations. This second order scheme, which we will denote

as SS^0, can be written as

$$q_{j+1}(\mathbf{r}) = \exp\left[-\frac{\Delta s}{2}w(\mathbf{r})\right]\exp\left[\Delta s\nabla^2\right]\exp\left[-\frac{\Delta s}{2}w(\mathbf{r})\right]q_j(\mathbf{r}), \tag{14}$$

for all nodes $\mathbf{r}$ of a spatial, uniform grid. This method requires only one pair of FFT's per step and has apparent unconditional stability. As a one-step method, it also allows for variable step size although, to our knowledge, this feature has not been exploited. For smooth fields w, this method is hard to beat, cost and stability-wise, among second order schemes. It has however one significant drawback for SCFT computations, particularly for highly segregated systems and for stochastic (complex Langevin) simulations; it has poor high-modal damping. Indeed, to first order in Δs

$$\exp\left[-\frac{\Delta s}{2}w(\mathbf{r})\right] \approx 1 - \frac{\Delta s}{2}w(\mathbf{r}) \tag{15}$$

and consequently the Fourier modes of w, and hence of q, get decreased approximately by a factor of $\Delta s/2$. Figure 1 shows the spectrum of the approximation of $q(s = 1/2, \mathbf{r})$ obtained with scheme (14) given a random, uniformly distributed w field of amplitude 10^{-4}, with a spatial resolution of $N_r = 256$ nodes, and uniform $\Delta s = 0.5/N_s$ for $N_s = 32$ and $N_s = 256$ (N_s is the number of contour nodes). This numerical experiment confirms that indeed the Fourier modes of q are decreased by approximately a factor of $\Delta s/2$, for $N_s \lessapprox N_r$. Moreover, we observe that as N_s increases (for fixed N_r) the attenuation factor asymptotically approaches Δs. This poor damping is independent of N_r, which is a particularly serious limitation in the stiff limit, $N_r \to \infty$, relevant for highly segregated systems.

Implicit-Explicit (IMEX) Runge-Kutta (RK) methods [2] offer a wide class of schemes suitable for problem (13). While in general more expensive than their multistep counterparts [3], the IMEX RK methods have superior stability properties and allow for easy variable step size and step size control. Out of this wide class, we select a second order IMEX RK scheme with strongest high modal attenuation. This method corresponds to the $(2, 2, 2)$ scheme derived by Ascher *et al.*, which we will denote as RK_{222}, and for the MDE (13) can be written as

$$\begin{aligned}
\left[1 - \gamma\Delta s\nabla^2\right] q^{(1)}(\mathbf{r}) &= \left[1 - \gamma\Delta s\, w(\mathbf{r})\right] q_j(\mathbf{r}), \\
\left[1 - \gamma\Delta s\nabla^2\right] q_{j+1}(\mathbf{r}) &= \left[1 - \beta\Delta s\, w(\mathbf{r})\right] q_j(\mathbf{r}) + \Delta s\left[(1-\gamma)\nabla^2 - (1-\beta)w(\mathbf{r})\right] q^{(1)}(\mathbf{r}),
\end{aligned} \tag{16}$$

where $\gamma = (2 - \sqrt{2})/2$ and $\beta = 1 - 1/(2\gamma)$. This is a two-stage, diagonally implicit RK (DIRK) method which can be implemented with 4 FFT's per step . It is *L stable* (the amplification factor is zero at the stiffness limit [14]) and *stiffly accurate* (it gives the exact solution to $y' = \lambda y$ as $\lambda\Delta s \to \infty$ [14]).

Figure 2 compares the spectrum of $q(s = 1/2, \mathbf{r})$ obtained with the RK_{222} (16) and with SS^0 for the same previous test with $\Delta s = 0.5/32$ and $N_r = 1024$. As remarked above, the attenuation of SS^0 is flat ($\Delta s/2$ across modes) and remains the same for $N_r = 1024$ as it was for $N_r = 256$. In marked contrast, high modal damping of the RK_{222} becomes even stronger as N_r increases because the method is L-stable.

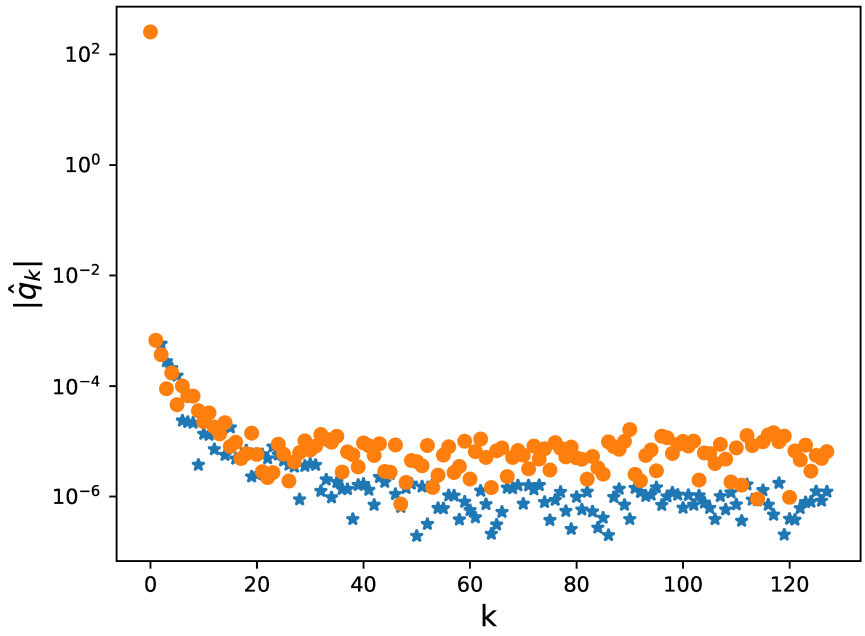

Figure 1: Spectrum of the numerical approximation to $q(s = 1/2, \mathbf{r})$ obtained with SS^0 scheme (14) when w is a random field and $\Delta s = 0.5/32$ (circles) and $\Delta s = 0.5/256$ (stars).

Before proceeding with the construction of high order schemes, we note that the forward-backward Euler

$$\left[1 - \Delta s \nabla^2\right] q_{j+1}(\mathbf{r}) = \left[1 - \Delta s\, w(\mathbf{r})\right] q_j(\mathbf{r}), \tag{17}$$

which is a first order IMEX RK method has, like RK_{222}, strong high modal damping and despite its low accuracy it may be useful for non-smooth fields as is the case in Complex Langevin computations.

4.2. Fourth Order and Beyond

The need for higher than second order methods, particularly for large χN, has been well documented [9, 21, 1]. In [9], a fourth order IMEX multistep method [3] was employed for large χN SCFT computations. Another fourth order method, which results by applying Richardson's extrapolation to SS^0, has been more extensively used [17, 1]. The IMEX multistep method is cost efficient per step, requiring only one FFT pair, but as pointed out in [4] it has limited stability properties.

The fourth order, extrapolated Strang splitting method, which we will denote as SS^1, is given by

$$q_{j+1}(\mathbf{r}) = \frac{4S^0_{\Delta s/2}[w]q_j(\mathbf{r}) - S^0_{\Delta s}[w]q_j(\mathbf{r})}{3}, \tag{18}$$

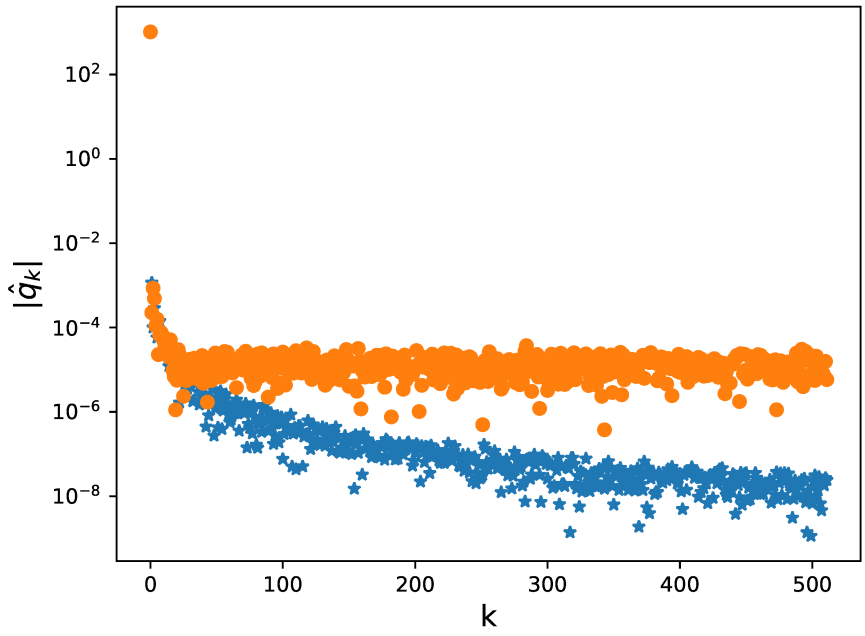

Figure 2: Spectrum of the numerical approximation to $q(s = 1/2, \mathbf{r})$, when w is a random field and $\Delta s = 0.5/32$, obtained with SS^0 (circles) with RK_{222} (stars). $N_r = 1024$.

where $S^0_{\Delta s}[w]q_j(\mathbf{r})$ stands for the right hand side of (14). This method requires 6 FFT's per step. It is possible to save one w-exponentiation by combining adjacent half-steps in the computation of $S^0_{\Delta s/2}[w]q_j$, though the main cost is really that of the FFT's. In principle, one can obtain a method of order $2p$ by applying extrapolation $p-1$ times to the original SS scheme. For example, applying extrapolation twice we get the 6th order scheme (SS^2)

$$q_{j+1}(\mathbf{r}) = \frac{16S^1_{\Delta s/2}[w]q_j(\mathbf{r}) - S^1_{\Delta s}[w]q_j(\mathbf{r})}{15}, \tag{19}$$

where $S^1_{\Delta s}[w]q_j(\mathbf{r})$ stands for the right hand side of (18). However, this repeated extrapolation quickly becomes prohibitively expensive. The $6th$ order method (19) has a cost of 18 FFT's per step.

Not surprisingly, the extrapolated methods (18) and (19) inherit the poor damping of the SS^0 scheme. For example, SS^1 has an flat attenuation factor of $\Delta s/6$ and that of SS^2 is $(7/90)\Delta s$ for $N_s \lessapprox N_r$ and both factors approach Δs as $N_s \to \infty$.

4.3. Spectral Deferred Corrections

An alternative approach we propose here to construct robust methods of arbitrarily high order in Δs for the SCFT MDE's is *spectral deferred correction* (SDC) [10]. In the classical deferred correction approach, one solves the differential equation system in question with

a given method, then solves a differential equation system for the error (derived from the original system) with the same method, add the resulting approximation of the error to the original approximation, and repeat the process as desired. Unfortunately, due to repeated numerical differentiation and interpolation at equally-spaced nodes (assuming a uniform step size) this process is numerically unstable and in practice only a very small number of nodes can be used. Dutt, Greengard, and Rokhlin [10] proposed a way to overcome these difficulties and to achieve robust methods of arbitrarily high order. Their SDC methods are based on the integral form of the differential equation system as it is done in Picard's iteration, and on the use of Legendre nodes for interpolation and the corresponding Gaussian quadrature for integration.

For concreteness, we describe now the SDC approach for the particular case of the MDE (13) and point out the variations we make to the original method of Dutt *et al.*[10]. We start by rewriting (13) as

$$q(s, \mathbf{r}) = q(0,\, \mathbf{r}) + \int_0^s \left[\nabla^2 q(\tau, \mathbf{r}) - w(\mathbf{r}) q(\tau, \mathbf{r})\right] d\tau. \tag{20}$$

Suppose we find an approximation $q^{[0]}$ to the solution of (20) with a given method. Define the residual of this approximation as

$$\epsilon^{[0]}(s, \mathbf{r}) = q(0,\, \mathbf{r}) + \int_0^s \left[\nabla^2 q^{[0]}(\tau, \mathbf{r}) - w(\mathbf{r}) q^{[0]}(\tau, \mathbf{r})\right] d\tau - q^{[0]}(s, \mathbf{r}) \tag{21}$$

and the error

$$\delta^{[0]}(s, \mathbf{r}) = q(s, \mathbf{r}) - q^{[0]}(s, \mathbf{r}). \tag{22}$$

Then, the error satisfies the integral equation

$$\delta^{[0]}(s, \mathbf{r}) = \int_0^s \left[\nabla^2 \delta^{[0]}(\tau, \mathbf{r}) - w(\mathbf{r}) \delta^{[0]}(\tau, \mathbf{r})\right] d\tau + \epsilon^{[0]}(s, \mathbf{r}). \tag{23}$$

The same method employed to solve (20) can now be used to solve (23) to find an approximation of the error, $\delta^{[0]}$. We then define a new, corrected approximation by

$$q^{[1]}(s, \mathbf{r}) = q^{[0]}(s, \mathbf{r}) + \delta^{[0]}(s, \mathbf{r}) \tag{24}$$

and the process can be repeated to generate $q^{[2]}, \ldots, q^{[J]}$, for some pre-determined number of deferred corrections J. We will denote this SDC method with J corrections and N_s (Chebyshev) nodes as $\mathrm{SDC}_{N_s}^{J}$.

If the method to solve (20) and (23) is order p and the quadrature to compute each residual is $O(\Delta s)^m$ accurate then the order of accuracy obtained by doing J deferred corrections is [5]

$$O(\Delta s)^\alpha, \quad \alpha = \min\{(J+1)p, m\}. \tag{25}$$

Thus, the deferred correction process can only be repeated as long as the integral in the residual is evaluated with sufficient accuracy. To this effect, Dutt *et al.* [10] use the Gaussian (Legendre nodes) quadrature and hence the adjective *spectral* in their method. Here, we propose to employ the interpolatory quadrature based on the Chebyshev (Gauss-Lobatto) points because of its implementation efficiency via the DCT and to use the end points of integration, relevant for the SCFT MDE problem. This yields also a spectral quadrature with a convergence rate about half that of the optimal Gaussian quadrature. But for smooth integrands, this difference is irrelevant as both quadratures achieve machine precision with just a few nodes [20]. The standard Clenshaw-Curtis quadrature to evaluate the volume fraction operators (9) and (10) and a related Chebyshev-node based quadrature to evaluate the integral operator in (21) are derived in Appendix A.

We propose to use the second order RK_{222} (16) to solve (20) and (23). In principle, one can use any non-stiff method that allows for variable step-size, including the SS^0 method (14), with a modification to solve the non-homogeneous equation (23), or the first order forward-backward Euler scheme (17). Dutt *et al.* [10] considered only first order schemes but for the SCFT MDE's the forward-backward Euler method requires a much larger number of nodes than the second order scheme (16), and consequently increased memory and ultimately higher computational cost. We have already argued about the desirability of strong high modal damping when solving the SCFT MDE's. We conducted a numerical study and found that L stability or at least very strong damping appears to be necessary for solving the error equation (23) during the deferred correction iteration.

The RK_{222} (16) can be applied directly to solve (20) using a variable step size:

$$\begin{aligned}
\left[1-\gamma\Delta s_j\nabla^2\right] q^{(1)}(\mathbf{r}) &= \left[1-\gamma\Delta s_j\, w(\mathbf{r})\right] q_j(\mathbf{r}),\\
\left[1-\gamma\Delta s_j\nabla^2\right] q_{j+1}(\mathbf{r}) &= \left[1-\beta\Delta s_j\, w(\mathbf{r})\right] q_j(\mathbf{r})\\
&\quad + \Delta s_j\left[(1-\gamma)\nabla^2-(1-\beta)w(\mathbf{r})\right] q^{(1)}(\mathbf{r}),
\end{aligned} \tag{26}$$

for $j=0,1,\ldots N_s$, where now $\Delta s_j = s_{j+1}-s_j$ and

$$s_j = \frac{f}{2}-\frac{f}{2}\cos\left(\frac{j\pi}{N_s}\right), \quad j=0,1,\ldots N_s, \tag{27}$$

are the N_s+1 Chebyshev nodes in $[0,f]$. For the error equation, the RK_{222} becomes

$$\begin{aligned}
\left[1-\gamma\Delta s_j\nabla^2\right] \delta^{(1)}(\mathbf{r}) &= \left[1-\gamma\Delta s_j\, w(\mathbf{r})\right] \delta_j(\mathbf{r}) + \gamma(\epsilon_{j+1}(\mathbf{r})-\epsilon_j(\mathbf{r})),\\
\left[1-\gamma\Delta s_j\nabla^2\right] \delta_{j+1}(\mathbf{r}) &= \left[1-\beta\Delta s_j\, w(\mathbf{r})\right] \delta_j(\mathbf{r}) + \Delta s_j\left[(1-\gamma)\nabla^2-(1-\beta)w(\mathbf{r})\right] \delta^{(1)}(\mathbf{r})\\
&\quad + \epsilon_{j+1}(\mathbf{r})-\epsilon_j(\mathbf{r})
\end{aligned} \tag{28}$$

for $j=0,1,\ldots,N_s-1$.

The cost of SDC^J using the RK_{222} is approximately $4(J+1)$ FFT's whereas that of the J-times extrapolated Strang splitting scheme (SS^J) is about $2\cdot 3^J$ FFT's. SS^1 is approximately 30% cheaper than $\mathrm{SDC}^{\,1}$ but for $J>1$ the cost of SDC^J is a fraction of that of SS^J.

We now compare SS^1 and SDC for the MDE (13) for a fixed given field w, i.e. isolated from the SCFT saddle point iteration. For this test we take

$$w(r) = 9\cos(6\pi r/L), \quad 0\le r\le L \tag{29}$$

with $L = 10$. This field is qualitatively similar to that in a SCFT computation for $\chi N \approx 40$. We fix the spatial resolution to $N_r = 128$. To estimate the error of the approximations produced by SS^1 and SDC at $s = f = 1/2$, we compute a reference solution obtained with SDC^3_{1024} (a resolution study was performed to determine that the SDC^3_{1024} approximation converges within about 14 digits of accuracy. This reference solution was also compared with a high resolution SS^2 approximation). Figure 3 displays this reference solution (at $s = f = 1/2$) and its spectrum. The spectrum of the approximation produced with SS^1_{1024} is also included for comparison. Note that the SS^1_{1024} produces a significant amplification of the round-off error for low wave numbers which prevents this scheme from reaching more than about 11 digits of accuracy for this example. This problem becomes exacerbated in a SCFT computation for moderate to high χN.

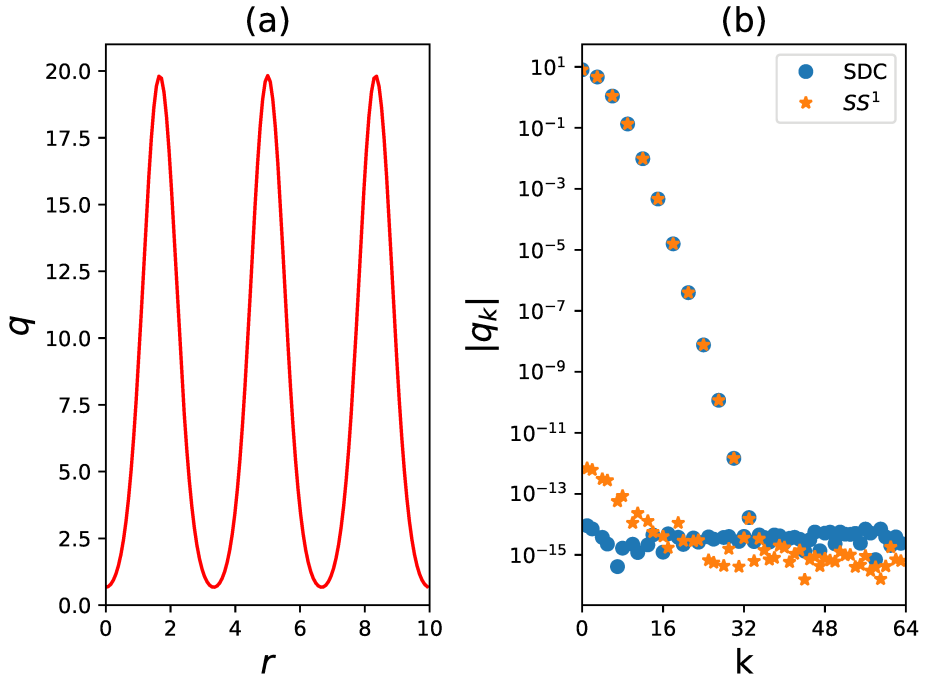

Figure 3: (a) $q(1/2, r)$ and (b) spectrum of the approximation to $q(1/2, r)$ obtained with SDC^3_{1024} (circles) and SS^1_{1024} (stars). $N_r = 128$.

Using the reference solution we evaluate the error (in the maximum norm) of the approximations obtained by employing SS^1 and SDC at difference accuracy levels. Table 1 presents these data along with a normalized execution time (relative to the corresponding SS^1 execution time). The SDC method produces more accurate and faster approximations than SS^1 with a fraction of the nodes required by SS^1. For example, with only $N_s = 12$ the SDC can get to an $O(10^{-7})$ error, second row in Table 1, whereas the SS^1 requires $N_s = 128$ for that accuracy and is about twice more expensive. The superior performance of SDC becomes even more striking at higher levels of accuracy.

Method	Error	time/(SS^1 time)
SS^1_{64}	1.61×10^{-6}	1.0
SDC^4_{10}	5.67×10^{-7}	0.65
SS^1_{128}	1.032×10^{-7}	1.0
SDC^4_{12}	8.11×10^{-8}	0.50
SS^1_{512}	4.11×10^{-10}	1.0
SDC^5_{16}	2.46×10^{-10}	0.18
SS^1_{1024}	2.75×10^{-11}	1.0
SDC^6_{20}	2.63×10^{-12}	0.13

Table 1: Comparison of the extrapolated SS (SS^1) and the SDC for different levels of accuracy. The subscript in the methods is the number of nodes in s and the superscript in SDC is the number of deferred corrections. $N_r = 256$ and the error is computed using the maximum norm.

5. Contour Spectral SCFT

We now look at the SCFT problem for a diblock copolymer melt. The saddle point iteration we employ is the semi-implicit Siedel (SIS) scheme [6]:

$$\begin{aligned}\frac{\mu_+^{j+1} - \mu_+^j}{\Delta t} &= -(g_{AA} + 2g_{AB} + g_{BB}) * \mu_+^{j+1} + \frac{\delta H[\mu_+^j, \mu_-^j]}{\delta \mu_+} \\ &\quad + (g_{AA} + 2g_{AB} + g_{BB}) * \mu_+^j, \end{aligned} \tag{30}$$

$$\frac{\mu_-^{j+1} - \mu_-^j}{\Delta t} = -(2/\chi N)\mu_-^{j+1} - \frac{\delta H[\mu_+^{j+1}, \mu_-^j]}{\delta \mu_-} + (2/\chi N)\mu_-^j, \tag{31}$$

where $*$ denotes convolution and the Fourier symbols of the kernels are

$$\hat{g}_{AA}(k) = \frac{2}{k^4}[fk^2 + \exp(-k^2 f) - 1], \tag{32}$$

$$\hat{g}_{AB}(k) = \frac{1}{k^4}[1 - \exp(-k^2 f)][1 - \exp(-k^2(1-f))], \tag{33}$$

$$\hat{g}_{BB}(k) = \frac{2}{k^4}[(1-f)k^2 + \exp(-k^2(1-f)) - 1]. \tag{34}$$

Each update fields is followed by a step in which the zeroth mode of μ_+^{j+1} and μ_-^{j+1} is set to zero.

We consider next two illustrative cases corresponding to a low-moderate $\chi N = 16$ and a high $\chi N = 80$ for a symmetric diblock $f = 1/2$.

The size of the first variation of H, (7)-(8), in any norm might not be an accurate stopping criterium for the saddle point iteration [1]. In the numerical experiments to follow, a highly accurate reference solution is first computed with a high resolution, and many-level SDC

to obtain, up to as many digits as possible, the free energy H corresponding to the saddle point. We call this value $H_{\rm ref}$ and set the stopping criterium to be

$$|H_{\rm ref} - H^{j+1}| < \epsilon_H \, H_{\rm ref}, \tag{35}$$

where H^{j+1} is the free energy evaluated at μ_+^{j+1} and μ_-^{j+1} and ϵ_H is the desired accuracy in the energy.

In our first example we take $\chi N = 16$ and the size of the domain is $L = 10$. The step size for the SIS iteration is $\Delta t = 500$ and the spatial resolution is fixed to $N_r = 256$. We use as initial guess for the SIS iteration the fields:

$$\mu_+(r) = -0.1\cos(2\pi r/L), \quad \mu_-(r) = 0.1\cos(2\pi r/L). \tag{36}$$

The saddle points fields, μ_+ and μ_- are plotted in Fig. 4. Table 2 compares SS[1] and SDC

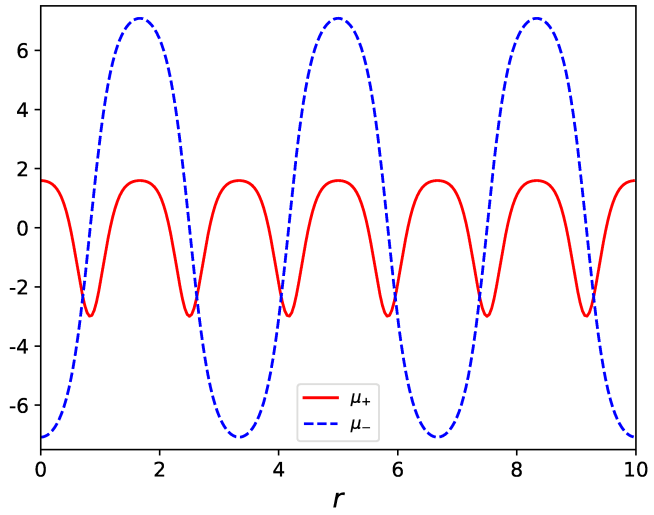

Figure 4: The saddle points fields, μ_+ and μ_- for $\chi N = 16$ and $L = 10$.

for this χN at different levels of accuracy in the relative error of the energy. The number of contour points per block was selected to be approximately the minimal number required for each method to achieve the desired accuracy. However, in the case of the SDC scheme it is possible to use an even smaller number of contour points at the expense of increasing the number of iterations. At low accuracies ($\epsilon_H \leq 10^{-6}$) both methods have a similar cost, except that SDC can use a fraction of the contour nodes that SS[1] requires and hence a much smaller memory. At higher accuracies, SDC easily outperforms SS[1] and with an order of magnitude fewer contour points.

It is important to note that the SCFT iteration is a method for inverting the smoothing operators ϕ_A and ϕ_B and consequently the iteration produces amplification of high wave-number modes, including those of the round-off error. This is immediate to see from the

ϵ_H	Method	Iterations	time/(SS^1 time)
10^{-6}	SS^1_{32}	44	1.0
	SDC^5_8	43	0.84
10^{-8}	SS^1_{80}	47	1.0
	SDC^5_{10}	47	0.41
10^{-10}	SS^1_{200}	53	1.0
	SDC^5_{20}	70	0.40
10^{-12}	SS^1_{600}	78	1.0
	SDC^5_{26}	98	0.17

Table 2: Comparison of SS^1 and SDC for $\chi N = 16$ at different levels of accuracy for the energy. The subindex in each method indicates the number of contour points per block and the superindex in SDC is the number of deferred corrections.

asymptotic expansion of these operators at high k, which yields the following expansion for the first variation of H:

$$\frac{\delta H[\mu_+,\mu_-]}{\delta \mu_+} = -(g_{AA}+2g_{AB}+g_{BB})*\mu_+ + (g_{AA}-g_{BB})*\mu_- + \dots . \tag{37}$$

$$\frac{\delta H[\mu_+,\mu_-]}{\delta \mu_-} = \frac{2}{\chi N}\mu_- - (g_{AA}-2g_{AB}+g_{BB})*\mu_- + (g_{AA}-g_{BB})*\mu_+ + \dots \tag{38}$$

The spectrum of μ_+, corresponding to high accuracy computations obtained with SDC^6_{128} and SS^1_{600} after 400 iterations is displayed in Fig. 5. There is a clear amplification of the round-off error, which is now is $O(10^{-13})$. While it is practically flat across k for SS^1_{600}, it is smaller and k-dependent for the more accurate SDC^6_{128}, consistent with the inversion of the leading order term in (37), $(\hat{g}_{AA}(k)+2\hat{g}_{AB}(k)+\hat{g}_{BB}(k))^{-1} \approx k^2$. The round-off error application becomes more pronounced as χN increases because the smoothing effect of the term $-\frac{2}{\chi N}\mu_-$ diminishes. This phenomenon is inherent to the ill-posedness of the inverse problem of finding a saddle point for H and not of the particular numerical method employed to solve the MDE's, as Fig. 5 demonstrates. If unattended, it could lead to a significant loss of accuracy and eventually cause instability of the iteration, particularly for large χN.

One approach to control the growth of the round-off error in some ill-posed problems is to employ a Fourier filter [15, 7] consisting of setting to zero all Fourier modes below a threshold ϵ_F near machine precision. That is, to filter a periodic array we compute its DFT, set to zero all of the Fourier coefficients whose modulus is less than ϵ_F, compute the inverse DFT.

We now consider $\chi N = 80$ and $L = 5$. Now the spatial resolution is set to $N_r = 512$ and the SIS step size is $\Delta t = 40$. We also apply Fourier filtering to μ_- and μ_+ at every iteration

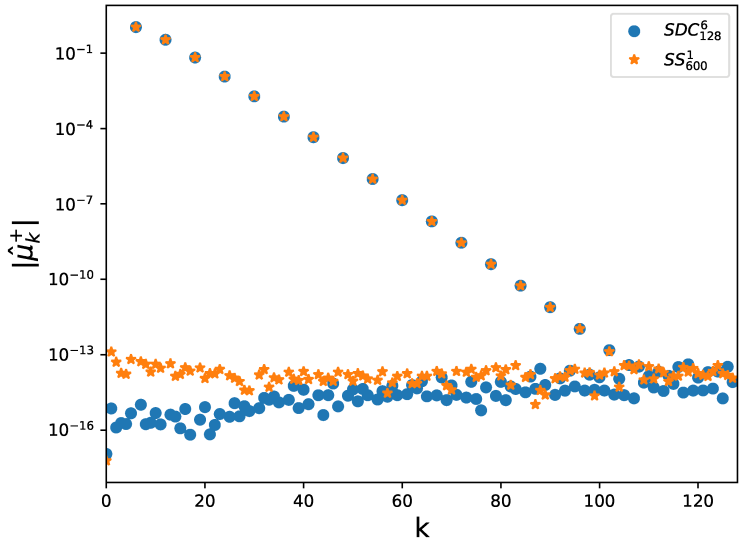

Figure 5: The spectrum of μ_+ for $\chi N = 16$ and $L = 10$ obtained with SDC^6_{128} (circles) and SS^1_{600} (stars).

with $\epsilon_F = 10^{-12}$. The initial guess for the SIS iteration is

$$\mu_+(r) = -0.1\cos(4\pi r/L), \quad \mu_-(r) = 0.1\cos(4\pi r/L). \tag{39}$$

We compute a reference energy H_{ref} using SDC^5_{512} and cross checked this with a computation using SDC^{10}_{128}. Their relative difference is $O(10^{-9})$. The saddle point fields are displayed in Fig.6.

Table 3 offers a comparison of SDC and SS^1 schemes at different accuracies in the energy as expressed in (35). Again, the number of contour points per block was selected to be approximately the minimal number required for each method to achieve the desired accuracy although no attempt was made to fine-tune the combination of contour points and levels of deferred corrections for the SDC scheme. The superiority of SDC over SS^1 is even more marked for this large χN case.

6. Adaptive Order SCFT Iterations

We propose now a strategy to accelerate a SCFT saddle point computation by adaptively varying the order of the MDE SDC scheme during the iteration. This strategy is inspired by the multilevel embedding [6], which uses initial guesses constructed through hierarchically finer resolutions, but it is more effective as the resolution, both in r and s, is keep fixed (avoiding interpolation) and only the order of the SDC changes.

The strategy is the following: select the number of contour points for each block (e.g. based on χN) and start the SCFT saddle point iteration with only one level of deferred correction (fourth order method), iterate until the relative change in the first variation of H in two consecutive iterations is less than a threshold value ϵ_T. Then, increase the number

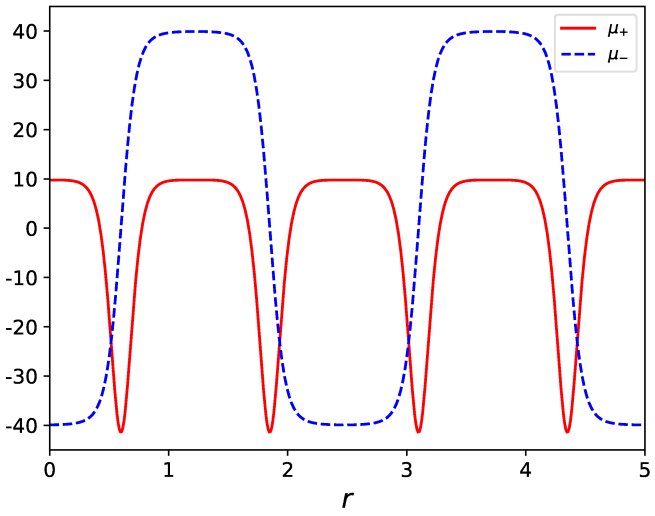

Figure 6: The saddle points fields, μ_+ and μ_- for $\chi N = 80$ and $L = 5$.

of deferred corrections by one and repeat until convergence to the desired level of accuracy or until the maximum number of allowed deferred corrections has been reached. In more detail, define

$$\left\| \frac{\delta H^j}{\delta \mu} \right\| = \left\| \frac{\delta H[\mu_+^j, \mu_+^j]}{\delta \mu_-} \right\| + \left\| \frac{\delta H[\mu_+^j, \mu_-^j]}{\delta \mu_-} \right\|. \tag{40}$$

Then, we increase the level of spectral deferred corrections by one during the SCFT iteration when

$$\left\| \frac{\delta H^{j+1}}{\delta \mu} \right\| - \left\| \frac{\delta H^j}{\delta \mu} \right\| < \left\| \frac{\delta H^{j+1}}{\delta \mu} \right\| \epsilon_T. \tag{41}$$

The threshold value ϵ_T depends on the accuracy sought. We use the $\|\cdot\|_\infty$ norm in our implementation.

To illustrate the efficacy of this strategy we reconsider the highly segregated case, $\chi N = 80$. Figure 7 shows a plot of the maximum norm of the first variation of H against the SIS iterations for both the adaptive order SDC strategy and the fixed SDC_{32}^7 up to reaching a relative error in the energy $\epsilon_H = 10^{-6}$. For this particular case we took $\epsilon_T = 0.01$. For most of the iterations the error is larger for the adaptive order SDC because it is using fewer than 7 levels of correction; only in the last 4 iterations the method uses 7 levels to reach quickly the desired accuracy in about one third of time required by the fixed order SDC and about 12 times faster than SS^1.

Low contour resolution (as few 8 or so points per block and zero levels of deferred correction) SCFT iterations can be useful for obtaining good initial fields for the higher resolution

ϵ_H	Method	Iterations	time/(SS^1 time)
10^{-5}	SS^1_{256}	53	1.0
	SDC^6_{32}	53	0.42
10^{-6}	SS^1_{400}	85	1.0
	SDC^7_{32}	86	0.26
10^{-7}	SS^1_{800}	115	1.0
	SDC^6_{64}	108	0.19
10^{-8}	SS^1_{1200}	183	1.0
	SDC^6_{64}	159	0.12

Table 3: Comparison of SS^1 and SDC for $\chi N = 80$, $L = 5$ at different levels of accuracy for the energy. The subindex in each method indicates the number of contour points per block and the superindex in SDC is the number of deferred corrections.

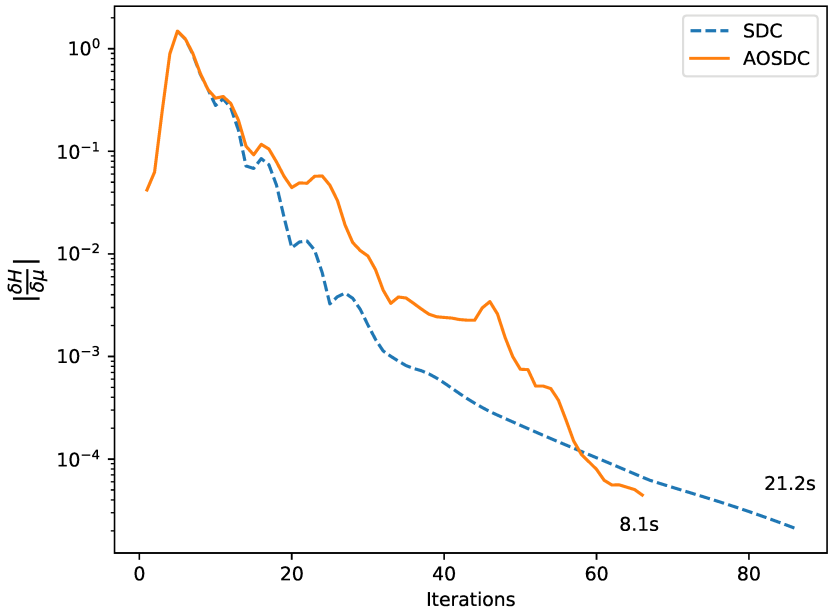

Figure 7: Maximum norm of the first variation of H versus number of iterations to reach a relative energy level of 10^{-6} for $\chi N = 80$ for the adaptive order SDC (continuous curve) and the fixed order SDC (dashed curve).

SCFT iterations in a negligible cpu time to further speed-up the convergence to the saddle point. This technique can also be employed when using random initial fields. After a few hundred iterations, the random noise level is low enough and can be Fourier filtered. The resulting fields, consisting of just the first modes provide a smooth, good initial guess for the

highly accurate SDC SCFT iterations.

7. Conclusions

We propose a cost and memory efficient, highly accurate method for the solution of polymer SCFT. The method is built from spectral integration using Chebyshev (Gauss-Lobatto) nodes in the chain contour variable and an arbitrary order spectral deferred correction (SDC) method for the modified diffusion (Fokker-Planck) equations. Special attention is paid to the selection of the core implicit-explicit scheme and its behavior in the stiff limit. The resulting method is robust and achieves high accuracy with a minimal number of contour nodes. This translates into an order of magnitude savings in memory, relative to existing approaches, and superior computational efficiency. The savings in memory are particularly relevant for GPU implementations as GPU memory is notoriously limited.

We also propose an adaptive approach to significantly accelerate the computation of the saddle points by systematically adapting the order of the SDC scheme during the iteration, without the use of interpolation and/or memory increase. The idea is to use initial guess produced with increasingly high order of accuracy. This approach can also be employed to obtain good initial fields for higher resolution SCFT iterations in a negligible cpu time.

Acknowledgements

The author would like to thank Glenn Fredrickson and Kris Delaney for stimulating discussions about this work. The author gratefully acknowledges partial support by the National Science Foundation under grant DMS 1818821.

Appendix A. Spectral Integration with Chebyshev nodes

We provide here the details of the spectral integration using the (second kind) Chebyshev or Gauss-Lobatto nodes to compute

$$\int_a^b f(s)ds \quad \text{and} \quad \int_a^{s_j} f(t)dt, \tag{A.1}$$

where s_j, $j = 0, \ldots, n$ are the Chebyshev nodes in $[a, b]$. The interpolatory quadrature using the Chebyshev nodes for the first integral is known as the Clenshaw-Curtis quadrature [8]. To obtain it we take the interval $[-1, 1]$ and for a general interval $[a, b]$ we use the change of variables

$$x = \frac{a+b}{2} + \frac{b-a}{2}t, \quad t \in [-1, 1]. \tag{A.2}$$

The Chebyshev nodes in $[-1, 1]$ are

$$s_j = -\cos\left(\frac{j\pi}{n}\right), \quad j = 0, 1, \ldots, n \tag{A.3}$$

The interpolating polynomial of f at these nodes can be written as

$$p_n(s) = \frac{a_0}{2} + \sum_{k=1}^{n-1} a_k T_k(s) + \frac{a_n}{2} T_n(s), \tag{A.4}$$

where $T_k(s)$ stands for the Chebyshev polynomial of degree k. Setting $s = -\cos\theta$, for $\theta \in [0, \pi]$ we get

$$p_n(-\cos\theta) = \frac{a_0}{2} + \sum_{k=1}^{n-1} a_k \cos k\theta + \frac{1}{2} a_n \cos n\theta. \tag{A.5}$$

Then $\Pi_n(\theta) = p_n(-\cos\theta)$ interpolates $F(\theta) = f(-\cos\theta)$ at the uniform nodes $\theta_j = j\pi/n$. Therefore,

$$a_k = \frac{2}{n} \sum_{j=0}^{n}{}'' F(\theta_j) \cos k\theta_j, \qquad k = 0, 1, .., n, \tag{A.6}$$

where the double prime in the sum means that the first and last coefficient have to be multiplied by a factor of $1/2$. That is, the coefficients $a_0, a_1, \ldots, a_n$ are the (Type I) Discrete Cosine Transform (DCT) coefficients of F [12, 13] and we can compute them efficiently in $O(n \log_2 n)$ operations with the FFT. With the change of variable $s = -\cos\theta$ we get

$$\int_{-1}^{1} f(s)ds = \int_0^{\pi} F(\theta) \sin\theta d\theta, \approx \int_0^{\pi} \Pi_n(\theta) \sin\theta d\theta. \tag{A.7}$$

But

$$\int_0^{\pi} \Pi_n(\theta) \sin\theta d\theta = \frac{a_0}{2} \int_0^{\pi} \sin\theta d\theta + \sum_{k=1}^{n-1} a_k \int_0^{\pi} \cos k\theta \sin\theta d\theta + \frac{a_n}{2} \int_0^{\pi} \cos n\theta \sin\theta d\theta. \tag{A.8}$$

Using $\cos k\theta \sin\theta = \frac{1}{2}[\sin(1+k)\theta + \sin(1-k)\theta]$ and assuming n is even we get the Clenshaw-Curtis Quadrature

$$\int_{-1}^{1} f(s)ds \approx a_0 + \sum_{\substack{k=2 \\ k \text{ even}}}^{n-2} \frac{2a_k}{1-k^2} + \frac{a_n}{1-n^2}. \tag{A.9}$$

For a general interval $[a, b]$, we get an extra factor of $(b-a)/2$ from the change of variables (A.2)

We adapt the Clenshaw-Curtis idea to evaluate

$$\int_{-1}^{s_j} f(t)dt = \int_0^{\theta_j} F(\theta) \sin\theta d\theta \approx \int_0^{\theta_j} \Pi_n(\theta) \sin\theta d\theta, \tag{A.10}$$

at the Chebyshev points (A.3). Since

$$\int_0^{\theta_j} \cos k\theta \sin\theta d\theta = \begin{cases} \frac{1}{4} - \frac{1}{4}\cos 2\theta_j \quad \text{for } k = 1 \\ \dfrac{1}{1-k^2} - \dfrac{\cos(k+1)\theta_j}{2(k+1)} + \dfrac{\cos(k-1)\theta_j}{2(k-1)}, \quad \text{for } k \neq 1, \end{cases} \tag{A.11}$$

we get

$$\int_{-1}^{s_j} f(t)dt \approx \frac{A_0}{2} + \sum_{k=1}^{n-1} A_k \cos k\theta_j + \frac{1}{2}A_n \cos n\theta_j - \frac{a_n}{4(n+1)} \cos(n+1)\theta_j, \tag{A.12}$$

where

$$A_0 = a_0 + \frac{1}{2}a_1 + \sum_{k=2}^{n-1} \frac{2a_k}{1-k^2} + \frac{a_n}{1-n^2}, \tag{A.13}$$

$$A_k = \frac{1}{2k}(a_{k+1} - a_{k-1}), \quad k = 1, \ldots, n-2, \tag{A.14}$$

$$A_{n-1} = \frac{1}{2(n-1)} \left(\frac{a_n}{2} - a_{n-2} \right), \tag{A.15}$$

$$A_n = -\frac{1}{2n} a_{n-1}. \tag{A.16}$$

The first three terms in the right hand size of (A.12) can be evaluated fast with the DCT so the overall cost is again $O(n \log_2 n)$.

References

[1] A. Arora, J. Qin, D. C. Morse, K. T. Delaney, G. H. Fredrickson, F. S. Bates, and K. D. Dorfman. Broadly accessible self-consistent field theory for block polymer materials discovery. *Macromolecules*, 49(13):4675–4690, 2016.

[2] U. M. Ascher, S. J. Ruuth, and R. J. Spiteri. Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations. *Appl. Numer. Math.*, 25:151, 1997.

[3] U. M. Ascher, S. J. Ruuth, and B. Wetton. Implicit-Explicit Methods for Partial Differential Equations. *SIAM J. Numer. Anal.*, 32(3):797–823, 1995.

[4] D. J. Audus, K. T. Delaney, H. D. Ceniceros, and G. H. Fredrickson. Comparison of pseudospectral algorithms for field-theoretic simulations of polymers. *Macromolecules*, 46(20):8383–8391, 2013.

[5] K. Böhmer and H. J. Stetter, editors. *Deferred correction methods. Theory and Applications.* Springer-Verlag, Wien, 1984.

[6] H. D. Ceniceros and G. H. Fredrickson. Numerical solution of polymer self-consistent field theory. *Multiscale Modeling & Simulation*, 2(3):452–474, 2004.

[7] H. D. Ceniceros and T. Y. Hou. The singular perturbation of surface tension in hele-shaw flows. *Journal of Fluid Mechanics*, 409:251–272, 2000.

[8] C. W. Clenshaw and A. R. Curtis. A method for numerical integration on an automatic computer. *Numerische Mathematik*, 2(1):197–205, Dec 1960.

[9] E. W. Cochran, C. J. Garcia-Cervera, and G. H. Fredrickson. Stability of the gyroid phase in diblock copolymers at strong segregation. *Macromolecules*, 39:2449–2451, 2006.

[10] A. Dutt, L. Greengard, and V. Rokhlin. Spectral deferred correction methods for ordinary differential equations. *BIT Numerical Mathematics*, 40(2):241–266, Jun 2000.

[11] G. H. Fredrickson. *The Equilibrium Theory of Inhomogeneous Polymers*. Oxford University Press, New York, 2006.

[12] W. M. Gentleman. Implementing Clenshaw-Curtis quadrature, I methodology and experience. *Comm. ACM*, 15(5):337–342, 1972.

[13] W. M. Gentleman. Implementing Clenshaw-Curtis quadrature, II computing the cosine transformation. *Comm. ACM*, 15(5):343–346, 1972.

[14] E. Hairer and G. Wanner. *Solving ordinary differential equations II. Stiff and differential-algebraic problems*. Springer-Verlag, Berlin Heidelberg, 1996.

[15] R. Krasny. A study of singularity formation in a vortex sheet by the point vortex approximation. *J. Fluid. Mechanics*, 167:65–93, 1986.

[16] Q. Liang, K. Jiang, and P. Zhang. Efficient numerical schemes for solving the self-consistent field theory equations of flexible-semiflexible diblock copolymers. *Mathematical Methods in the Applied Sciences*, 38(2):4553–4563, 2015.

[17] A. Ranjan, J. Qin, and D. C. Morse. Linear response and stability of ordered phases of block copolymer melts. *Macromolecules*, 41(3):942–954, 2008.

[18] K. O. Rasmussen and G. Kalosakas. Improved numerical algorithm for exploring block copolymer mesophases. *Journal of Polymer Science Part B-Polymer Physics*, 40(16):1777–1783, 2002.

[19] G. Strang. On the construction and comparison of difference schemes. *SIAM J. Numer. Anal.*, 5(3):506–516, 1968.

[20] L. N. Trefethen. Is Gauss quadrature better than Clenshaw-Curtis? *SIAM Review*, 50(1):67–87, 2008.

[21] W. Xu, K. Jiang, P. Zhang, and A.-C. Shi. A strategy to explore stable and metastable ordered phases of block copolymers. *The Journal of Physical Chemistry B*, 117(17):5296–5305, 2013. PMID: 23551204.