A Parallel Routine for Liquid Crystal Simulation

Yixiang Luo

May 21, 2017

1 Introduction

Liquid crystal is an important material widely used in areas such as display, lacer and smart glass. In order to apply liquid crystal, we need to know its phases under certain conditions. This package is an implementation of minimizing the energy functional of liquic crystal described in [1]. The specific phenomenons studied are the phases of rodlike liquid crystal in a unit ball, with enforcing LC molecules pointing radially on the shpere.

A simple form of free energy for this case

$$F = \int_\Omega B : Q - \log Z - \frac{1}{2}\alpha_1 |Q|^2 + \frac{1}{2}\alpha_2 |\bigtriangledown Q|^2 \, dV + F_{pena} \tag{1}$$

Where B is a 3×3 matrix function in the unit ball Ω, $Z = Z(B)$, and we denote Q as

$$Q = \begin{pmatrix} q_1 & q_2 & q_3 \\ q_2 & q_4 & q_5 \\ q_3 & q_5 & -q_1 - q_4 \end{pmatrix}$$

Penalty term

$$F_{pena} = \int_{\partial\Omega} [q_1 xy - q_2(x^2 - \frac{1}{3})]^2 + [q_2 z - q_3 y]^2 + [q_4 xy - q_2(y^2 - \frac{1}{3})]^2 + [q_2 z - q_5 x]^2 \, dS$$

enforces the molecules pointing radially on the shpere. This satisfies the so-called relax radial anchoring boundary condition

$$(q_1, q_2, q_3, q_4, q_5) // (x^2 - \frac{1}{3}, xy, xz, y^2 - \frac{1}{3}, yz)$$

Function $B^*((x))$ which minimizes the energy functional (1) describes a stable phase of liquid crystal which achieves local minimal free energy. And our purpose is to find such functions.

2 Serial Algorithm

2.1 General Idea

The general idea of the algorithm is to express the energy functional (1) approximately from finite number of variables, then use optimization method (L-BFGS method[2] here) to minimize it.

2.2 Notes and Definitions

Enforce the elements of matrix function

$$Q = \begin{pmatrix} q_1 & q_2 & q_3 \\ q_2 & q_4 & q_5 \\ q_3 & q_5 & -q_1 - q_4 \end{pmatrix}$$

to be polynomials of r, θ, ϕ,

$$q_i = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} \sum_{l=0}^{L-1} A_{nml}^{(i)} r^n \theta^m \phi^l$$

where r, θ, ϕ are variables in spherical coordinate system, in other word,

$$r = \sqrt{x^2 + y^2 + z^2}$$

$$\theta = arccos \frac{z}{\sqrt{x^2 + y^2 + z^2}}$$

$$\phi = arctan\left(\frac{y}{x}\right)$$

Moreover, we denote $B = P\tilde{B}P^T$, where

$$\tilde{B} = diag(b_1, b_2, -b_1 - b_2)$$

is a diagonal matrix and P is a rotation matrix expressed by Euler angle α, β, γ

$$P = \begin{pmatrix} \cos(\alpha)\cos(\gamma) - \cos(\beta)\sin(\alpha)\sin(\gamma) & \cos(\gamma)\sin(\alpha) + \cos(\alpha)\cos(\beta)\sin(\gamma) & \sin(\beta)\sin(\gamma) \\ -\cos(\beta)\cos(\gamma)\sin(\alpha) - \cos(\alpha)\sin(\gamma) & \cos(\alpha)\cos(\beta)\cos(\gamma) - \sin(\alpha)\sin(\gamma) & \cos(\gamma)\sin(\beta) \\ \sin(\alpha)\sin(\beta) & -\cos(\alpha)\sin(\beta) & \cos(\beta) \end{pmatrix}$$

We can calculate Q and Z from these five free variables $b_1, b_2, \alpha, \beta, \gamma$ at any point in Ω. That is, employ *the routine for Bingham moment*, we get Z and the eigenvalue diagonal matrix

$$\tilde{Q} = diag(\tilde{q_1}, \tilde{q_2}, -\tilde{q_1} - \tilde{q_2})$$

form b_1, b_2. Then we get

$$Q = P\tilde{Q}P^T$$

Notice that q_n $(n = 1, 2, \cdots, 5)$ are $(N-1)(M-1)(L-1)$ order polynomial. We denote N Gaussian Nodes in [0, 1] as $\{r_i\}$, M Gaussian Nodes in [0, π] as $\{\theta_j\}$, and L Gaussian Nodes in [0, 2π] as $\{\phi_k\}$. Then $\{(r_i, \theta_j, \phi_k)\}$ can determine Q at $N \times M \times L$ points, which further determine the whole Q in Ω. Therefore, we set the values of $b_1, b_2, \alpha, \beta, \gamma$ on $\{(r_i, \theta_j, \phi_k)\}$ are the $5NML$ free variables of the total energy (1). For simplicity, we denote

$$var^{i,j,k} = var\big|_{(r_i, \theta_j, \phi_k)}$$

where var stands for $b_1, b_2, \alpha, \beta, \gamma, q_n, \tilde{q_i}, Q, Z$. And denote

$$D^{i,j,k} = D_0^{i,j,k} \cdot r_i^2 sin\theta_j$$

where $D_0^{i,j,k}$ stands for the numerical integration coefficient at (r_i, θ_j, ϕ_k) for integration over Ω.

2.3 Pre-calculation

Denote

$$Z_{n,m} = \int_{|x|=1} x_1^n x_2^m \, exp(b_1 x_1^2 + b_2 x_2^2 + (-b_1 - b_2)x_3^2) \, dx$$

and

$$Q_{n,m} = Z_{n,m}/Z_{0,0}$$

Clearly, $Z_{n,m}$ and $Q_{n,m}$ can be calculated directly by *the routine for Bingham moment*. And for Z, $\tilde{q_n}$ and q_n, we have

$$Z = Z_{0,0}$$

$$\tilde{q_1} = Q_{2,0}, \;\; \tilde{q_2} = Q_{0,2}$$

And we can get q_n by

$$Q = P\tilde{Q}P^T$$

Moreover, the gradient of the total energy (1) is needed for optimization method. we need to calculate the following terms in preparation of computing the gradient.

$$\frac{\partial}{\partial b_1} Z = -Z_{0,0} + 2Z_{2,0} + Z_{0,2}$$

$$\frac{\partial}{\partial b_2} Z = -Z_{0,0} + Z_{2,0} + 2Z_{0,2}$$

$$\frac{\partial}{\partial b_1} \tilde{q_1} = 2Q_{4,0} + Q_{2,2} - (2Q_{2,0} + Q_{0,2})Q_{2,0}$$

$$\frac{\partial}{\partial b_2} \tilde{q_1} = Q_{4,0} + 2Q_{2,2} - (Q_{2,0} + 2Q_{0,2})Q_{2,0}$$

$$\frac{\partial}{\partial b_1} \tilde{q_2} = 2Q_{2,2} + Q_{0,4} - (2Q_{2,0} + Q_{0,2})Q_{0,2}$$

$$\frac{\partial}{\partial b_2} \tilde{q_2} = Q_{2,2} + 2Q_{0,4} - (Q_{2,0} + 2Q_{0,2})Q_{0,2}$$

and

$$\frac{\partial}{\partial b_i} Q = P \cdot \frac{\partial}{\partial b_i} \tilde{Q} \cdot P^T$$

$$\frac{\partial}{\partial \alpha} Q = \frac{\partial}{\partial \alpha} P \cdot \tilde{Q}\, P^T + \left(\frac{\partial}{\partial \alpha} P \cdot \tilde{Q}\, P^T \right)^T$$

$$\frac{\partial}{\partial \beta} Q = \frac{\partial}{\partial \beta} P \cdot \tilde{Q}\, P^T + \left(\frac{\partial}{\partial \beta} P \cdot \tilde{Q}\, P^T \right)^T$$

$$\frac{\partial}{\partial \gamma} Q = \frac{\partial}{\partial \gamma} P \cdot \tilde{Q}\, P^T + \left(\frac{\partial}{\partial \gamma} P \cdot \tilde{Q}\, P^T \right)^T$$

2.4 Bulk Energy

For $\int_\Omega B : Q - \log Z \, dV$ term, We calculate it by numerical integration

$$\begin{aligned}\int_\Omega B : Q - \log Z \, dV &= \int_0^1 \int_0^\pi \int_0^{2\pi} [b_1\tilde{q_1} + b_2\tilde{q_2} + (b_1+b_2)(\tilde{q_1}+\tilde{q_2}) - \log Z] \cdot r^2 sin\theta \, d\phi d\theta dr \\ &\approx \sum_{i=0}^{N-1}\sum_{j=0}^{M-1}\sum_{k=0}^{L-1} D^{i,j,k} \cdot [b_1^{i,j,k}\tilde{q_1}^{i,j,k} + b_2^{i,j,k}\tilde{q_2}^{i,j,k} \\ &\quad + (b_1^{i,j,k} + b_2^{i,j,k})(\tilde{q_1}^{i,j,k} + \tilde{q_2}^{i,j,k}) - \log Z^{i,j,k}]\end{aligned}$$

And for its gradient, stil denote $var^{i,j,k}$ as var for simplicity, then we get

$$\frac{\partial}{\partial b_1}\int_\Omega B : Q - \log Z \, dV = D^{i,j,k}\left(2\tilde{q_1} + \tilde{q_2} + (2b_1+b_2)\frac{\partial \tilde{q_1}}{\partial b_1} + (b_1+2b_2)\frac{\partial \tilde{q_2}}{\partial b_1} - \frac{1}{Z}\frac{\partial Z}{\partial b_1}\right)$$

$$\frac{\partial}{\partial b_2}\int_\Omega B : Q - \log Z \, dV = D^{i,j,k}\left(\tilde{q_1} + 2\tilde{q_2} + (2b_1+b_2)\frac{\partial \tilde{q_1}}{\partial b_2} + (b_1+2b_2)\frac{\partial \tilde{q_2}}{\partial b_2} - \frac{1}{Z}\frac{\partial Z}{\partial b_2}\right)$$

Similarly, we have

$$\begin{aligned}\int_\Omega |Q|^2 \, dV &= \int_0^1 \int_0^\pi \int_0^{2\pi} [\tilde{q_1}^2 + \tilde{q_2}^2 + (\tilde{q_1}+\tilde{q_2})^2] \cdot r^2 sin\theta \, d\phi d\theta dr \\ &\approx \sum_{i=0}^{N-1}\sum_{j=0}^{M-1}\sum_{k=0}^{L-1} D^{i,j,k} \cdot [(\tilde{q_1}^{i,j,k})^2 + (\tilde{q_2}^{i,j,k})^2 + (\tilde{q_1}^{i,j,k} + \tilde{q_2}^{i,j,k})^2]\end{aligned}$$

And denote $var^{i,j,k}$ as var for simplicity,

$$\frac{\partial}{\partial b_1}\int_\Omega |Q|^2 \, dV = 2D^{i,j,k} \cdot \left[\tilde{q_1}(2\frac{\partial \tilde{q_1}}{\partial b_1} + \frac{\partial \tilde{q_2}}{\partial b_1}) + \tilde{q_2}(\frac{\partial \tilde{q_1}}{\partial b_1} + 2\frac{\partial \tilde{q_2}}{\partial b_1})\right]$$

$$\frac{\partial}{\partial b_1}\int_\Omega |Q|^2 \, dV = 2D^{i,j,k} \cdot \left[\tilde{q_1}(2\frac{\partial \tilde{q_1}}{\partial b_2} + \frac{\partial \tilde{q_2}}{\partial b_2}) + \tilde{q_2}(\frac{\partial \tilde{q_1}}{\partial b_2} + 2\frac{\partial \tilde{q_2}}{\partial b_2})\right]$$

2.5 Elastic Energy

Calculating elastic energy term is a bit more complex. To get $\int_\Omega |\nabla Q|^2 \, dV$, we need to calculate

$$\frac{\partial q_n}{\partial r}, \ \frac{\partial q_n}{\partial \theta}, \ \frac{\partial q_n}{\partial \phi}$$

first. Here, we take $\frac{\partial q_n}{\partial r}$ for example.

For any give j_0 and k_0, q_n is a (N-1)th order polynomial of r. Therefore, denoting N-dimensional column vector $\mathbf{p} = (\frac{\partial q_n}{\partial r}|_{(r_i,j_0,k_0)})_i$ and N-dimensional

column vector $\mathbf{q} = (q_n^{i,j_0,k_0})_i$, we get

$$\mathbf{p} = T^{(r)}\mathbf{q} \tag{2}$$

where $T^{(r)}$ is a $N \times N$ matrix only related to $\{r_i\}$. Its elements $T_{ij}^{(r)}$ are

$$T_{ij}^{(r)} = \left(\sum_{n=0,n\neq j}^{N} \prod_{k=0,k\neq j,k\neq n}^{N} (r_i - r_k) \right) / \prod_{k=0,k\neq j}^{N} (r_j - r_k) \tag{3}$$

Similarly, we can define matrix $T^{(\theta)}$ and $T^{(\phi)}$.

Using (2), we can get all the

$$\frac{\partial q_n}{\partial r}|_{(r_i,\theta_j,\phi_k)}$$

from $\{q_n^{i,j,k}\}$. Similarly, $\frac{\partial q_n}{\partial \theta}$ and $\frac{\partial q_n}{\partial \phi}$ can also be calculated like this.

Then, by the chain rule

$$\begin{aligned}
\frac{\partial q_n}{\partial x} &= \frac{\partial q_n}{\partial r}\frac{\partial r}{\partial x} + \frac{\partial q_n}{\partial \theta}\frac{\partial \theta}{\partial x} + \frac{\partial q_n}{\partial \phi}\frac{\partial \phi}{\partial x} \\
\frac{\partial q_n}{\partial y} &= \frac{\partial q_n}{\partial r}\frac{\partial r}{\partial y} + \frac{\partial q_n}{\partial \theta}\frac{\partial \theta}{\partial y} + \frac{\partial q_n}{\partial \phi}\frac{\partial \phi}{\partial y} \\
\frac{\partial q_n}{\partial z} &= \frac{\partial q_n}{\partial r}\frac{\partial r}{\partial z} + \frac{\partial q_n}{\partial \theta}\frac{\partial \theta}{\partial z} + \frac{\partial q_n}{\partial \phi}\frac{\partial \phi}{\partial z}
\end{aligned}$$

and

$$\frac{\partial r}{\partial x} = \frac{x}{\sqrt{x^2+y^2+z^2}} \quad \frac{\partial r}{\partial y} = \frac{y}{\sqrt{x^2+y^2+z^2}} \quad \frac{\partial r}{\partial z} = \frac{z}{\sqrt{x^2+y^2+z^2}}$$

$$\frac{\partial \theta}{\partial x} = \frac{xz}{\sqrt{x^2+y^2}\,(x^2+y^2+z^2)} \quad \frac{\partial \theta}{\partial y} = \frac{yz}{\sqrt{x^2+y^2}\,(x^2+y^2+z^2)}$$

$$\frac{\partial \theta}{\partial z} = -\frac{\sqrt{x^2+y^2}}{x^2+y^2+z^2}$$

$$\frac{\partial \phi}{\partial x} = -\frac{y}{x^2+y^2} \quad \frac{\partial \phi}{\partial y} = \frac{x}{x^2+y^2} \quad \frac{\partial \phi}{\partial z} = 0$$

we get

$$\nabla q_n = (\frac{\partial q_n}{\partial x}, \frac{\partial q_n}{\partial y}, \frac{\partial q_n}{\partial z})$$

Finally, we can calculate

$$\int_\Omega |\nabla Q|^2 \, dV = \int_0^1 \int_0^\pi \int_0^{2\pi} 2(\sum_{i=1}^{5} |\nabla q_i|^2 + \nabla q_1 \cdot \nabla q_4) r^2 sin\theta \, d\phi d\theta dr$$

by numerical integration.

Following we compute the gradient of $\int_\Omega |\nabla Q|^2 \, dV$ on our free variables. For simplicity, denote

$$\ddot{a} = \frac{\partial}{\partial a} \int_\Omega |\nabla Q|^2 \, dV$$

For simplicity, denote $q_{n,a} = \frac{\partial q_n}{\partial a}$ and denote $var^{i,j,k}$ as var. Then we have

$$\ddot{q}_{n,x} = D^{i,j,k} \cdot 4q_{n,x}, \quad n = 2, 3, 5$$

$$\ddot{q}_{1,x} = D^{i,j,k} \left(4q_{1,x} + 2q_{4,x}\right)$$

$$\ddot{q}_{4,x} = D^{i,j,k} \left(4q_{4,x} + 2q_{1,x}\right)$$

Similarly, we can calculte $\ddot{q}_{n,y}$ and $\ddot{q}_{n,z}$. And further, by applying the chain rule, $\ddot{q}_{n,r}$, $\ddot{q}_{n,\theta}$ and $\ddot{q}_{n,\phi}$ can be obtained.

Then we calculate $\ddot{q}_n\big|_{i_0,j_0,k_0}$ for any i_0, j_0, k_0. Denote N-dimensional, M-dimensional and L-dimensional column vector

$$\ddot{\mathbf{p}}_r = (\ddot{q}_{n,r}\big|_{i,j_0,k_0})_i, \;\; \ddot{\mathbf{p}}_\theta = (\ddot{q}_{n,\theta}\big|_{i_0,j,k_0})_j, \;\; \ddot{\mathbf{p}}_\phi = (\ddot{q}_{n,\phi}\big|_{i_0,j_0,k})_k$$

and denote the nth column of matrix T as T_n, then we have

$$\ddot{q}_n\big|_{i_0,j_0,k_0} = T^{(r)}_{i_0} \cdot \ddot{\mathbf{p}}_r + T^{(\theta)}_{j_0} \cdot \ddot{\mathbf{p}}_\theta + T^{(\phi)}_{k_0} \cdot \ddot{\mathbf{p}}_\phi \tag{4}$$

where $T^{(r)}$ is a matrix defined in (3), with $T^{(\theta)}$ and $T^{(\phi)}$ similar defined. Further, employing the chain rule and results in 2.3, we can obtain $\ddot{b}_1$, $\ddot{b}_2$, $\ddot{\alpha}$, $\ddot{\beta}$ and $\ddot{\gamma}$, which consist the gradient of $\int_\Omega |\nabla Q|^2 \, dV$.

2.6 Penalty Term

In order to applying numerical integration to calculate the penalty term, we need to compute $q_n\big|_{1,\theta_i,\phi_k}$ first. For simplicity, we denote $q_n\big|_{1,\theta_i,\phi_k}$ as $\hat{q_n}^{j,k}$.

For any give j_0 and k_0, q_n is a (N-1)th order polynomial of r. Therefore, denoting N-dimensional column vector $\mathbf{q} = (q_n^{i,j_0,k_0})_i$, we get

$$\hat{q_n}^{j,k} = \mathbf{v} \cdot \mathbf{q} \tag{5}$$

where $\mathbf{v}$ is a N-dimensional vector only related to $\{r_i\}$. Its elements v_i are

$$v_i = \prod_{k=0,k\neq j}^{N} \frac{1-r_k}{r_j - r_k} \tag{6}$$

Further, denote

$$D_s^{j,k} = D_{s0}^{j,k} \cdot sin\theta_j$$

where $D_{s0}^{j,k}$ stands for the numerical integration coefficient at $(1, \theta_j, \phi_k)$ for integration over $\partial\Omega$, or say, the unit sphere. And denote $sin\theta_j cos\phi_k$ as $x^{j,k}$, $sin\theta_j sin\phi_k$ as $y^{j,k}$ and $cos\theta_j$ as z^j for simplicity. Then we can calculate the penalty term by

$$\begin{aligned} F_{pena} &= \int_0^{\pi}\int_0^{2\pi} \Big\{[q_1 xy - q_2(x^2 - \frac{1}{3})]^2 + (q_2 z - q_3 y)^2 + [q_4 xy - q_2(y^2 - \frac{1}{3})]^2 \\ &\quad + (q_2 z - q_5 x)^2\Big\} sin\theta \, d\phi d\theta dr \\ &\approx \sum_{j=0}^{M-1}\sum_{k=0}^{L-1} D_s^{j,k} \cdot \Big\{[\hat{q_1}^{j,k} \cdot x^{j,k} y^{j,k} - \hat{q_2}^{j,k}((x^{j,k})^2 - \frac{1}{3})]^2 \\ &\quad + (\hat{q_2}^{j,k} z^j - \hat{q_3}^{j,k} y^{j,k})^2 + [\hat{q_4}^{j,k} \cdot x^{j,k} y^{j,k} - \hat{q_2}^{j,k}((y^{j,k})^2 - \frac{1}{3})]^2 \\ &\quad + (\hat{q_2}^{j,k} z^j - \hat{q_5}^{j,k} x^{j,k})^2\Big\} \end{aligned}$$

In the following we compute the gradient of F_{pena} on our free variables. For simplicity denote $var^{j,k}$ as var. Then we have

$$\frac{\partial}{\partial \hat{q_1}} F_{pena} = D^{j,k} \cdot 2xy[\hat{q_1} xy - \hat{q_2}(x^2 - \frac{1}{3})]$$

$$\begin{aligned} \frac{\partial}{\partial \hat{q_2}} F_{pena} = D^{j,k} \cdot \Big\{ &- 2(x^2 - \frac{1}{3})[\hat{q_1} xy - \hat{q_2}(x^2 - \frac{1}{3})] + 2z(\hat{q_2} z - \hat{q_3} y) \\ &- 2(y^2 - \frac{1}{3})[\hat{q_4} xy - \hat{q_2}(y^2 - \frac{1}{3})] + 2z(\hat{q_2} z - \hat{q_5} x)\Big\} \end{aligned}$$

$$\frac{\partial}{\partial \hat{q_3}} F_{pena} = -D^{j,k} \cdot 2y(\hat{q_2} z - \hat{q_3} y)$$

$$\frac{\partial}{\partial \hat{q_4}} F_{pena} = D^{j,k} \cdot 2xy[\hat{q_4} xy - \hat{q_2}(y^2 - \frac{1}{3})]$$

$$\frac{\partial}{\partial \hat{q_5}} F_{pena} = -D^{j,k} \cdot 2x(\hat{q_2} z - \hat{q_5} x)$$

Further, we have

$$\frac{\partial}{\partial q_n^{i,j,k}} F_{pena} = v_i \cdot \frac{\partial}{\partial \hat{q}_n^{j,k}} F_{pena}$$

Finally, employing the chain rule and results in 2.3, we can obtain

$$\frac{\partial}{\partial b_1^{i,j,k}} F_{pena}, \quad \frac{\partial}{\partial b_2^{i,j,k}} F_{pena}, \quad \frac{\partial}{\partial \alpha^{i,j,k}} F_{pena}, \quad \frac{\partial}{\partial \beta^{i,j,k}} F_{pena}, \quad \frac{\partial}{\partial \gamma^{i,j,k}} F_{pena},$$

which consist the gradient of F_{pena}.

3 Parallel Algorithm

The leading expenses in serial algorithm is vector inner product, making the routine run for hours or even a day for fine grid. Thus it is feasible and necessary to parallelize it. MPI is used for this part.

3.1 Partition

Each MPI task undertake a segment in axis r, θ and ϕ. Denote there are in total I tasks along r, J tasks along θ and K tasks along ϕ. Then the total tasks number is

$$N = I \cdot J \cdot K.$$

and each task should undertake

$$L = \frac{R}{I} \cdot \frac{T}{J} \cdot \frac{P}{K}$$

local points.

A diagram of the partiton is shown in Figure 1.

3.2 Parallel Function Evaluation

3.2.1 Bulk Energy

For $\int_\Omega B : Q - \log Z \, dV$ term, Things are quite easy. These numerical intergration can be reduced to a vector inner product finally. Thus each MPI task can calculate its own local product and need no communication.

The time complexity for calculation is $O(M \cdot N^{-1})$.

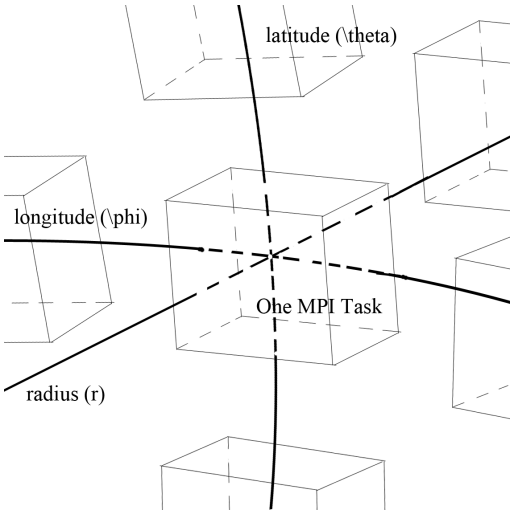


Figure 1: There are in total I task blocks along r axis, J tasks along θ axis and K tasks along ϕ axis.

3.2.2 Elastic Energy and Penalty Term

The elastic energy term $\int_\Omega |\nabla Q|^2 \, dV$ involves a spatial gradient ∇Q. Interpolations on all the points along axis r, θ or ϕ are employed to calculate the partial gradient, as demonstrated above. Thus for this part, each MPI task need to communicate q_i with all others lie on the same axis r, θ or ϕ and calculate inner products on them.

The time complexity for this communication is

$$O(\log I + \log J + \log K) = O(\log N).$$

And for calculation is

$$O((I + J + K)M \cdot N^{-1}).$$

The calculation of penalty term are similar.

3.3 Parallel Optimization

I simply parallelize the L-BFGS method[2] by paralizing the red marked inner products in the following demonstration of L-BFGS:

Denote the independent variables as $\boldsymbol{x}$.

- Evaluation:

$$F_k = F(\boldsymbol{x}_k), \quad \boldsymbol{g}_k = \nabla F(\boldsymbol{x}_k)$$

- Generating Search Direction $\boldsymbol{d}$ (the L-BFGS two loop recursion):

Denote $\boldsymbol{s}_k = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k, \quad \boldsymbol{y}_k = \boldsymbol{g}_{k+1} - \boldsymbol{g}_k$ and $\rho_k = 1/\boldsymbol{y}_k^T \boldsymbol{s}_k$.

$$\begin{aligned}
&\boldsymbol{d} = \boldsymbol{g}_k \\
&\text{for } i = k-1, k-2, \cdots, k-m \\
&\quad \alpha_i = \rho_i \boldsymbol{s}_i^T \boldsymbol{d} \\
&\quad d = d - \alpha_i \boldsymbol{y}_i \\
&\boldsymbol{d} = d/(\rho_{k-1} \boldsymbol{y}_{k-1}^T \boldsymbol{y}_{k-1}) \\
&\text{for } i = k-m, k-m+1, \cdots, k-1 \\
&\quad \beta_i = \rho_i \boldsymbol{y}_i^T \boldsymbol{d} \\
&\quad d = d + (\alpha_i - \beta_i) \boldsymbol{s}_i
\end{aligned}$$

- Line Search: Find an α satisfying

$$\begin{aligned}
&F(\boldsymbol{x}_k + \alpha \boldsymbol{d}) \leq F(\boldsymbol{x}_k) + \alpha \rho \boldsymbol{g}_k^T \boldsymbol{d} \\
&g(\boldsymbol{x}_k + \alpha \boldsymbol{d})^T \boldsymbol{d}_k \geq F(\boldsymbol{x}_k) + \sigma \boldsymbol{g}_k^T \boldsymbol{d}
\end{aligned}$$

- Update:

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha \boldsymbol{d}$$

The time complexity for communication is $O(\log N)$.

And for calculation is $O(M \cdot N^{-1})$.

Routine of this part is modified from the serial L-BFGS C library *libLBFGS* (`http://www.chokkan.org/software/liblbfgs/`).

4 Results

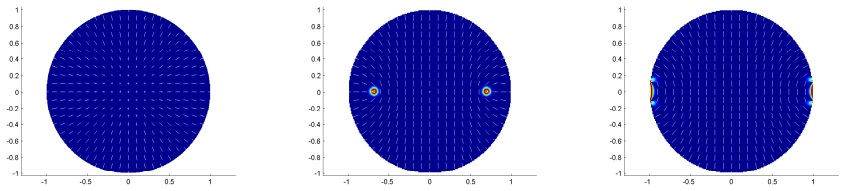
4.1 Liquid Crystal Phase

Here are some basic ideas of analyzing the numerical results. At a point, an eigenvalue of Q represent the probability of molecule pointing along its corresponding eigenvactor. Thus, to describe molecular at a point, we can use an ellipsoid whose semi-principal axes are eigenvectors of Q and axes length are corresponding eigenvalues plus $\frac{1}{3}$. Moreover, we call a point uniaxial if two of the eigenvalues are equal; otherwise biaxial. And we can define

$$\beta = 1 - 6\frac{(trQ^3)^2}{(trQ^2)^3}$$

to describe this biaxiality. For uniaxial Q, $\beta = 0$, while for biaxial Q, $\beta \neq 0$. What's more, if we cannot find an unique unit eigenvector corresponding to the biggest eigenvalue, or in other word, there are more than one biggest eigenvalue, we call the point a defect.

Now we can analyze the numerical result obtained by this routine.



(a) Radial Hedgehog (b) Ring Disclination (c) Sphere Ring

Figure 2: Three possible conguration under the relax radial anchoring condition. Figures are on yz-plane. White rods represent eigenvalues and eigenvectors at each point. The background color represents β, with red indicates biaxial and blue indicates uniaxial. In all three cases $\alpha_2 = 0.04$, and α_2s are $2(a)$ $\alpha_2 = 11$, $2(b)$ $\alpha_2 = 16$, $2(c)$ $\alpha_2 = 22$.

For the radial hedgehog solution, Q is uniaxial everywhere (Figure $2(a)$). The center of the ball is the only point of defect. For larger α_2, this point

defect broadens into a disclination ring (Figure 2(b)). Around the ring a torus of strong biaxial region (Figure $\beta \sim 1$) exists. In 2(c), two rings of isotropic points form on the sphere. On the surface between the two rings, Q is uniaxial and oblate. Inside there is a strong biaxial region close to the surface.

4.2 Performance of Parallel Computing

All the results here come from running the parallel routine in *Stampede* with one MPI task per node. The timings are for 100 iterations in optimization.

4.2.1 Strong Scaling

In strong scaling, I fix $M = 48^3$, $I \approx J \approx K$ and increase $N = I \cdot J \cdot K$. Since the leading order term of time complexity is $O((I+J+K)M \cdot N^{-1})$, in this case the total timing will approximately be $O(N^{-2/3})$, and the commnication time is $O(\log N)$. These are consistent with the experiment results shown in Figure 3.

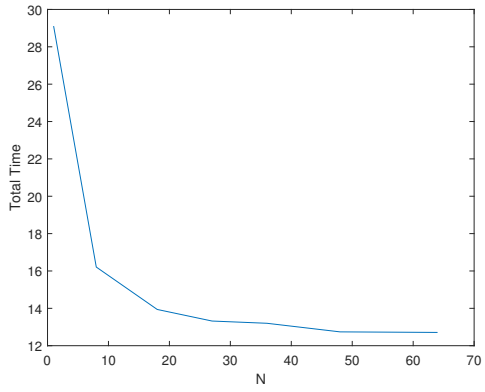
4.2.2 Weak Scaling

notice that the calculation time complexity is

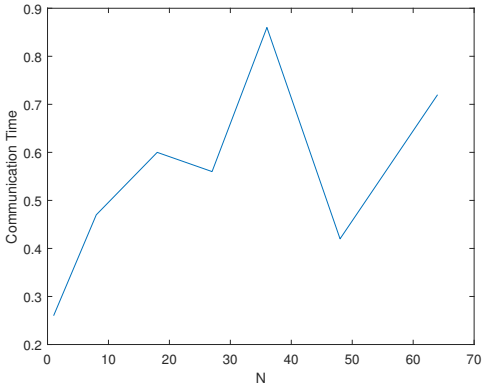
$$O((I + J + K)M \cdot N^{-1}) + O(M \cdot N^{-1}).$$

Therefore we cannot simply implement the weak scaling test since fixing the calculation amount per processor is impossible as integer I, J or K increases. Let's just fix $M \cdot N^{-1} = 16^3$ and see what happends.

As shown in Figure 4, both total time consuming and communication time are approximately linear with N, which is inconsistent with the theoretical results. I guess this may be because as the amount of variables increases, line searching in each iteration need more loops to find the satisfactory point.



(a) Total Timing



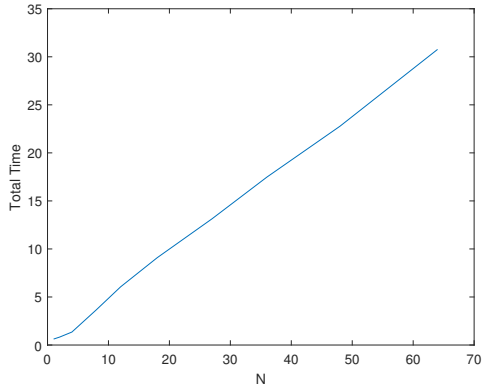
(b) Communication Time

Figure 3: The total timing and communication time for strong scaling.

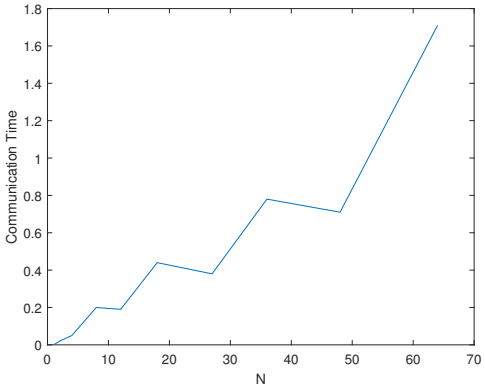
References

[1] Jiequn Han, Yi Luo, Wei Wang, Pingwen Zhang, and Zhifei Zhang. From microscopic theory to macroscopic theory: a systematic study on modeling for liquid crystals. *Archive for Rational Mechanics and Analysis*, 215(3):741–809, 2014.

[2] Mordecai Avriel. *Nonlinear programming: analysis and methods*. Courier Corporation, 2003.



(a) Total Timing



(b) Communication Time

Figure 4: The total timing and communication time for "weak" scaling.