Local Energy Dissipation Rate Preserving Approximations to Driven Gradient Flows with Applications to Graphene Growth

Lin Lu[a], Qi Wang[b], Yongzhong Song[a], Yushun Wang[a]

[a] *Jiangsu Key Laboratory for NSLSCS, School of Mathematical Sciences, Nanjing Normal University, Nanjing 210023, China*

[b] *Department of Mathematics, University of South Carolina, Columbia, SC 29208, USA*

Abstract

We develop a paradigm for developing local energy dissipation rate preserving (LEDRP) approximations to general gradient flow models driven by source terms. In driven gradient flow models, the deduced energy density transport equation possesses an indefinite source. Local energy-dissipation-rate preserving algorithms are devised to respect the mathematical structure of both the driven gradient flow model and its deduced energy density transport equation. The LEDRP algorithms are also global energy-dissipation-rate preserving (GEDRP) under proper boundary conditions such as periodic boundary conditions. However, the contrary may not be true. We then apply the paradigm to a phase field model for growth of a graphene sheet to produce a set of LEDRP algorithms. Numerical refinement tests are conducted to confirm the convergence property of the new algorithms and simulations of graphene growth are demonstrated to benchmark against existing results in the literature.

Keywords: Local energy-dissipation-rate preserving algorithms, graphene growth, driven gradient flow, finite difference, energy quadratization.

1. Introduction

A gradient flow refers to a thermodynamically consistent model describing relaxation dynamics of a nonequilibrium system with respect to a given free energy landscape subject to explicit or implicit constraints in the phase space. Two classes of well-known gradient flow models are the Allen-Cahn and Cahn-Hilliard phase field model for multiphase materials systems [1, 2]. Gradient flow models have been used in many important applications in recent years, like many self-consistent field theories for multiphase polymer solutions and melts, phase field crystal growth models, thin film models and various phase field models for crystalline alloys etc. [3, 4]. Given the system's free energy, the gradient flow model is derived using the generalized Onsager principle or equivalently the second law of thermodynamics [5, 6, 7, 8]. Gradient flow models can be either energy dissipative or energy conservative globally.

Among the gradient flow models, the crystal growth model is of great application value for it can produce a variety of complex patterns, ranging from dendritic to fractal, determined through

*Corresponding author

Email address: wangyushun@njnu.edu.cn (Yushun Wang)

interactions with the substrate and other environmental factors [9, 10, 11, 12, 13, 14]. Recently, as a result of its excellent electrical properties, high thermal conductivity and super mechanical properties, graphene has emerged as one of the promising new high performance materials and been widely applied to many high tech applications, like electronic information, environmental protection, biomedicine and others [15, 16, 17, 18]. To grow a single crystal graphene sheet, copper has been widely used as the substrate because the graphene sheet can be transferred to another substrate easily after its growth [19, 20]. Influenced by thermodynamic, kinetic, and growth factors, such as symmetry of the substrate copper, pressure of methane and hydrogen, adatom flux, morphologies of the growing single crystal graphene sheet can be dendrites, squares, stars, hexagons, butterflies and lobes, known as the kinetic Wulff shapes [10, 21, 22, 23, 24, 25, 26, 27].

To respect the global energy dissipation property of dissipative gradient flows in numerical algorithms, ones have developed a series of energy stable schemes for dissipative gradient flow models. First introduced in [28], the convex-splitting method splits the free energy density into one concave and one convex term so that they can be treated differentially when the temporal discretization is devised to ensure the global energy dissipation of the system. Several designs soon followed in various applications [29, 30, 31]. A few years ago, Chris Lieb et al. pointed out a potential shortcoming of this method however [32]. On a different front, Shen et al. used another popular technique called stabilization method, which is easy to implement and effective to many phase field models including the Allen-Cahn and Cahn-Hilliard equation [33]. After that, a novel approach that can surely preserve the energy dissipation rate by introducing an auxiliary variable to transform the free energy density into a quadratic form was introduced in [34]. It was coined the name energy quadratization (EQ) method by Yang, Zhao and Wang and applied to a host of gradient flow models [35, 36, 37, 38]. This method has since been extended to hydrodynamical models, producing linear, unconditionally energy stable schemes for the viscous binary fluid flow model [37] and various other applications [39, 40, 41, 42, 43]. Later, Shen et al. developed another EQ type method, called the scalar auxiliary variable method, and applied it to gradient flow models [44] and conducted the error analysis for L^2 and H^{-1} gradient flows with a typical form of free energy in [45]. The SAV method can be quite efficient for a class of gradient flows [46, 47]. Another class of methods, known as the projection, Lagrange multiplier or supplementary variable method [48, 49, 50, 51] have also been developed recently for gradient flow models, which not only capture dynamics but also preserve the global energy dissipation property.

To preserve the energy and other invariants in conservative gradient flow systems like Hamiltonian systems, the multisymplectic method has been introduced for years, applied to a large number of conservative PDEs and achieved remarkable success in long time simulations [52, 53, 54]. Other structure-preserving algorithms for conservative systems are constructed afterwards, like the discrete variational derivative method [55], the averaged vector method [56], the Hamiltonian boundary value method [57] and so on. Most of these methods focus on preserving the global energy and invariants of the models in which the local property and structure of the models were not seriously considered. In many models, the energy density, which represents the local energy, obeys a deduced energy transport equation describing intrinsic energy transport mechanism of the models locally. It would be desirable if one can preserve the structure and property of the energy transport equation locally when devising any numerical approximation to the gradient flow model. The numerical algorithms preserve the structure of the local energy transport equation are called local structure preserving algorithms (LSPAs). Under proper boundary conditions, the local energy preserving algorithm can yield corresponding global energy preserving property (i.e., for periodic boundary conditions), but the converse is normally not true. In 2008, Wang et al. [58] introduced the con-

cept of local structure preservation, and showed the advantage of the local structure preserving algorithms when applied to systems like solitary wave equations and the Maxwell's equation. Afterwards, this idea has been applied to a series of other conservative models with certain degree of success [59, 60, 61, 62, 63].

We note that the local energy dissipation rate preserving approximation is a special case of more general structure preserving approximation. For a dynamical system given by a partial differential equation system, if a function say f is identified as its energy density, a transport equation for f can be deduced and is called the energy transport equation. There can be other quantities of physical interest whose transport equation can be deduced in the solution manifold of the dynamical system as well. A structure-preserving numerical approximation means the numerical algorithm designed to solve the dynamical system also yields consistent approximations for transport equations of the energy density and other quantities that are of physical interest. Notice that when the dynamical system is coupled with its deduced equations, the coupled system is over-determined, consistent and structurally unstable, i.e, any small perturbations to the coupled system can render the perturbed system inconsistent and therefore unsolvable. The structural-preserving approximation requires the numerical approximation to the coupled system maintains consistency in the approximated system. This is a very stringent requirement. The success rate for meeting this requirement is normally low but not impossible. In this regard, the local energy dissipation rate preserving algorithm is one that enables one to produce a consistent approximation to a dynamical system coupled with its energy density transport equation. In this paper, we take the challenge to develop a systematic approach to devise such consistent numerical approximations.

Specifically, we develop a general framework or paradigm to devise local energy dissipation rate preserving algorithms for gradient flow models and gradient flow models with source terms, known as driven gradient flow models. *We note that for the driven gradient flow model, it becomes extremely important for the discrete system to respect the mathematical structure of the continuous PDE system's. Since the driven gradient flow system may no longer be dissipative nor conservative, local structure preservation can ensure necessary consistency between the discretized PDE system together with its deduced equations and the continuous one.* This paradigm can yield linear and nonlinear schemes. In particular, it produces linear ones when combined with the energy quadratization method. The paradigm is composed of two key ingredients. The first is to reformulate the gradient flow model into a system with auxiliary variables and low order derivatives while the second is to develop a hierarchical set of discrete Leibnitz rules using finite difference operators. Following this paradigm, local energy-dissipation-rate preserving algorithms are then devised. Under proper boundary conditions, for instance, periodic boundary conditions, the local structure-preserving schemes are also global structure-preserving. How to design a local structure-preserving schemes that are also global structure-preserving for a given physical boundary conditions however remains an open problem though. In [64], we presented a few cases where the global energy-dissipation-rate schemes can be constructed using a combination of local energy-dissipation-rate preserving schemes, where different LEDRP schemes are adopted at different part of boundaries.

In the second part of the study, we apply some newly developed algorithms to simulate crystal growth in graphene sheet on copper, using a gradient flow model with a source term introduced by Meca et al. [21]. We present three different second order algorithms using three different ways to discretize the energy density function. The algorithms are implemented in a rectangular domain and their second order convergence rates are verified numerically. Finally, numerical simulations on growth of graphene sheets on copper are carried out and benchmarked against existing results in the literature. We note that this numerical paradigm can be applied to any partial differential

equation system with deduced equations, not limited to thermodynamically consistent gradient flow models [64].

The rest of paper is organized as follows. In §2, we present the general framework for developing local energy dissipation rate preserving algorithms. In §3, we extend the method to driven gradient flow models. In §4, we apply the method to a graphene sheet growth model. In §5, mesh refinement tests for the graphene growth model and simulations for a variety of benchmark examples are presented. We give a concluding remark in the last section.

2. A framework of developing LEDRPs for gradient flows

In this section, we first present the general gradient flow model and derive the transport equation of the energy density in the model to show the local energy transport mechanism of the gradient flow system. Then, we develop a general framework to develop local energy-dissipation-rate preserving (LEDRP) algorithms for the system. Adopting the energy quadratization technique, we further develop a class of linear LEDRP algorithms.

2.1. General gradient flow models

We first derive the general gradient flow model using the generalized Onsager principle. Given the free energy of the system in terms of the functional of the phase variable ϕ

$$\mathcal{E} = \int_\Omega f(\phi, \nabla\phi, \nabla^2\phi, ..., \nabla^m\phi)\mathrm{d}\mathbf{x}, \tag{2.1}$$

where m is a positive integer and

$$E = f(\phi, \nabla\phi, \nabla^2\phi, ..., \nabla^m\phi) \tag{2.2}$$

is the free energy density.

The time derivation of the free energy $\mathcal{E}$ is given by

$$\frac{\mathrm{d}\mathcal{E}}{\mathrm{d}t} = \sum_{i=0}^{m}\left[(-1)^i\left(\nabla^i\frac{\partial f}{\partial\nabla^i\phi}, \phi_t\right) + \int_{\partial\Omega}\sum_{k=1}^{i}(-1)^{k-1}\nabla^{k-1}\frac{\partial f}{\partial\nabla^i\phi}\cdot\frac{\partial\nabla^{i-k}\phi}{\partial t}\cdot\mathbf{n}\mathrm{d}S\right], \tag{2.3}$$

where $(f, g) = \int_\Omega fg\mathrm{d}\mathbf{x}$ is the inner product of function $f(\mathbf{x})$ and $g(\mathbf{x})$. The energy time rate of change is dictated by two parts. One is the energy dissipation rate in the bulk and the other is the energy flux going through the boundary. For adiabatical boundaries, the energy flux is zero which is guaranteed by

$$\sum_{k=1}^{i}(-1)^{k-1}\nabla^{k-1}\frac{\partial f}{\partial\nabla^i\phi}\cdot\frac{\partial\nabla^{i-k}\phi}{\partial t}\cdot\mathbf{n} = 0,\ \forall\ i = 0, 1, ..., m. \tag{2.4}$$

For periodic boundary conditions, the net energy flux is zero as well. In both cases, the energy dissipation rate becomes

$$\frac{\mathrm{d}\mathcal{E}}{\mathrm{d}t} = \sum_{i=0}^{m}(-1)^i\left(\nabla^i\frac{\partial f}{\partial\nabla^i\phi}, \phi_t\right). \tag{2.5}$$

We denote

$$\frac{\delta \mathcal{E}}{\delta \phi} = \mu = \sum_{i=0}^{m} (-1)^i \nabla^i \frac{\partial f}{\partial \nabla^i \phi}. \tag{2.6}$$

This is also known as the generalized chemical potential. The free energy dissipation rate is given by

$$\frac{\mathrm{d}\mathcal{E}}{\mathrm{d}t} = \int_\Omega \frac{\delta \mathcal{E}}{\delta \phi} \cdot \phi_t \mathrm{d}\mathbf{x} = -\int_\Omega \frac{\delta \mathcal{E}}{\delta \phi} \cdot \mathcal{M} \cdot \frac{\delta \mathcal{E}}{\delta \phi} \mathrm{d}\mathbf{x}. \tag{2.7}$$

We note that when $\mathcal{M}$ is nonnegative, the free energy dissipates in time. This is what we call the dissipative gradient flow system. When $\mathcal{M}$ is antisymmetric, the energy is conserved. This is called a conservative gradient flow system.

Applying the generalized Onsager principle, we obtain the specific gradient flow system as follows

$$\begin{cases} \phi_t = -\mathcal{M}\mu, \quad \mathbf{x} \in \Omega,\ t > 0, \\ \mu = \displaystyle\sum_{i=0}^{m} (-1)^i \nabla^i \frac{\partial f}{\partial \nabla^i \phi}, \\ \phi(\mathbf{x}, 0) = \phi_0(\mathbf{x}), \end{cases} \tag{2.8}$$

where $\mathcal{M}$ is the mobility operator or coefficient.

Next, we examine how energy is transported in given gradient flow dynamics (2.8) for a given mobility operator. We state a lemma firstly.

Lemma 2.1. Based on the product rule, we have the following identity

$$(-1)^i \phi_t \cdot \nabla^i \frac{\partial f}{\partial \nabla^i \phi} = (-1)^i \nabla \cdot \left(\sum_{k=1}^{i} (-1)^{k-1} \nabla^{k-1} \phi_t \cdot \nabla^{i-k} \frac{\partial f}{\partial \nabla^i \phi} \right) + \nabla^i \phi_t \cdot \frac{\partial f}{\partial \nabla^i \phi}. \tag{2.9}$$

Applying the lemma, we arrive at the following transport equation of the energy density.

Theorem 2.1. The dissipative gradient flow system in (2.8) admits the following local energy dissipation law (LEDL)

$$\frac{\mathrm{d}E}{\mathrm{d}t} + \sum_{i=0}^{m} (-1)^i \nabla \cdot \left(\sum_{k=1}^{i} (-1)^{k-1} \nabla^{k-1} \phi_t \cdot \nabla^{i-k} \frac{\partial f}{\partial \nabla^i \phi} \right) + \mu \mathcal{M} \mu = 0, \tag{2.10}$$

where E is the free energy density defined in (2.2).

The proof of the theorem is given in the Appendix A.

This transport equation of the energy density can be rewritten into

$$\frac{\mathrm{d}E}{\mathrm{d}t} - \sum_{i=0}^{m} (-1)^i \nabla \cdot \left(\sum_{k=1}^{i} (-1)^{k-1} \nabla^{k-1} \mathcal{M}\mu \cdot \nabla^{i-k} \frac{\partial f}{\partial \nabla^i \phi} \right) + \mu \mathcal{M} \mu = 0. \tag{2.11}$$

Remark 2.1. In the free energy functional, we only consider terms that are invariant under rotation and translation. These are the terms of the following forms: $\nabla^{2i}\phi$ and $\|\nabla^{2i+1}\phi\|_2^2$, $i = 1, 2, 3, ...$, where $\|\|_2$ denotes the length of the vector in Cartesian space.

2.1.1. Reformulation of the model system

At the discrete level, we cannot derive a consistent, discretized, energy density transport equation directly unless we introduce some intermediate variables to reformulate the system. This is because of the discrete Leibnitz rule and the continuous Leibnitz rule are not compatible with any spatial discretization involving higher order derivatives. We note that when we introduce intermediate variables, we must discuss the cases with respect to the parity of index m. Here we focus on the case where m is odd, the other case is essentially the same.

We first rewrite the model as follows

$$\begin{cases} \phi_t = -\mathcal{M}\mu, \\ \mu = \displaystyle\sum_{p=0}^{\frac{m-1}{2}} \left(\Delta^p \frac{\partial f}{\partial \Delta^p \phi} - \nabla \Delta^p \frac{\partial f}{\partial \nabla \Delta^p \phi} \right), \end{cases} \tag{2.12}$$

then introduce two groups of intermediate variables

$$f'(\phi) = k_1, \quad \frac{\partial f}{\partial \Delta \phi} = k_2, \quad \frac{\partial f}{\partial \Delta^2 \phi} = k_3, \quad \cdots, \frac{\partial f}{\partial \Delta^{\frac{m-1}{2}} \phi} = k_{\frac{m+1}{2}}, \tag{2.13}$$

$$\frac{\partial f}{\partial \nabla \phi} = \mathbf{h}_1, \quad \frac{\partial f}{\partial \nabla \Delta \phi} = \mathbf{h}_2, \quad \frac{\partial f}{\partial \nabla \Delta^2 \phi} = \mathbf{h}_3, \quad \cdots, \frac{\partial f}{\partial \nabla \Delta^{\frac{m-1}{2}} \phi} = \mathbf{h}_{\frac{m+1}{2}}. \tag{2.14}$$

With these intermediate variables, the model is rewritten as follows

$$\begin{cases} \phi_t = -\mathcal{M}\mu, \\ \mu = \displaystyle\sum_{p=1}^{\frac{m+1}{2}} \Delta^{p-1} k_p - \sum_{q=1}^{\frac{m+1}{2}} \nabla \Delta^{q-1} \mathbf{h}_q, \\ \phi(\mathbf{x}, 0) = \phi_0(\mathbf{x}). \end{cases} \tag{2.15}$$

The transport equation of the free energy density for model (2.13) - (2.15) is given as follows.

Theorem 2.2. Model (2.13) - (2.15) satisfies the following LEDL

$$\begin{aligned} &\partial_t E + \nabla \cdot \left[\sum_{p=2}^{\frac{m+1}{2}} \sum_{l=0}^{2p-3} \left((-1)^{l+1} \nabla^l k_p \cdot \nabla^{2p-3-l} \phi_t \right) + \sum_{q=1}^{\frac{m+1}{2}} \sum_{r=0}^{2q-2} \left((-1)^{r+1} \nabla^r \mathbf{h}_q \cdot \nabla^{2q-2-r} \phi_t \right) \right] \\ &\quad + \mu \mathcal{M} \mu = 0, \end{aligned} \tag{2.16}$$

with E the local energy defined by (2.2).

2.1.2. Local structure preserving algorithms for the equivalent system

Next we present three LEDRP algorithms for the system in 2D space. Let N_x, N_y, N_y be three positive integers, we uniformly discretize domain $\Omega_h = [x_L, x_R] \times [y_L, y_R]$, with mesh sizes $h_x = (x_R - x_L)/N_x$, $h_y = (y_R - y_L)/N_y$. We choose time interval as $[0, T]$ and discretize it

uniformly to yield $t_n = n\tau$, with time step $\tau = T/N_t$, where $n = 0, 1, 2, ..., N_t$. Then the discrete domain reads as follows

$$\Omega_h = \{(x_j, y_k, t_n) | x_j = x_L + jh_x, y_k = y_L + kh_y, t_n = n\tau, 0 \le j \le N_x, 0 \le k \le N_y, 0 \le n \le N_t\}. \tag{2.17}$$

We denote $f^n_{j,k}$ as the approximate value $f(x, y, t)$ at node (x_j, y_k, t_n), $0 \le j \le N_x, 0 \le k \le N_y, 0 \le n \le N_t$.

We first discretize the free energy density (2.2), intermediate variables (2.13) - (2.14) and system (2.15) as follows

$$E^n_{j,k} = f\left(\phi^n_{j,k}, \nabla^+_h \phi^n_{j,k}, \Delta_h \phi^n_{j,k}, \nabla^+_h \Delta_h \phi^n_{j,k}, ..., \Delta_h^{\frac{m-1}{2}} \phi^n_{j,k}, \nabla^+_h \Delta_h^{\frac{m-1}{2}} \phi^n_{j,k}\right), \tag{2.18}$$

$$\left\{\mathcal{A}_t \frac{\partial f^n}{\partial \phi^n} = A_t k^n_1\right\}\Big|_{j,k}, \quad \left\{\mathcal{A}_t \frac{\partial f^n}{\partial \Delta_h \phi^n} = A_t k^n_2\right\}\Big|_{j,k}, \quad \cdots, \left\{\mathcal{A}_t \frac{\partial f^n}{\partial \Delta_h^{\frac{m-1}{2}} \phi^n} = A_t k^n_{\frac{m+1}{2}}\right\}\Big|_{j,k}, \tag{2.19}$$

$$\left\{\mathcal{A}_t \frac{\partial f^n}{\partial \nabla^+_h \phi^n} = A_t \mathbf{h}^n_1\right\}\Big|_{j,k}, \quad \left\{\mathcal{A}_t \frac{\partial f^n}{\partial \nabla^+_h \Delta_h \phi^n} = A_t \mathbf{h}^n_2\right\}\Big|_{j,k}, \quad \cdots, \left\{\mathcal{A}_t \frac{\partial f^n}{\partial \nabla^+_h \Delta_h^{\frac{m-1}{2}} \phi^n} = A_t \mathbf{h}^n_{\frac{m+1}{2}}\right\}\Big|_{j,k}, \tag{2.20}$$

$$\begin{cases} \left\{\delta^+_t \phi^n = -\mathcal{M} A_t \mu^n\right\}\Big|_{j,k}, \\ \left\{A_t \mu^n = \sum_{p=1}^{\frac{m+1}{2}} \Delta_h^{p-1} A_t k^n_p - \nabla^-_h \sum_{q=1}^{\frac{m+1}{2}} \Delta_h^{q-1} A_t \mathbf{h}^n_q\right\}\Big|_{j,k}. \end{cases} \tag{2.21}$$

Here, the operators are defined by

$$\begin{aligned} &A_t f^n_{j,k} = \frac{f^{n+1}_{j,k} + f^n_{j,k}}{2}, \quad \delta^+_t f^n_{j,k} = \frac{f^{n+1}_{j,k} - f^n_{j,k}}{\tau}, \quad \delta^+_x f^n_{j,k} = \frac{f^n_{j+1,k} - f^n_{j,k}}{h_x}, \\ &\delta^+_y f^n_{j,k} = \frac{f^n_{j,k+1} - f^n_{j,k}}{h_y}, \quad \delta^-_x f^n_{j,k} = \frac{f^n_{j,k} - f^n_{j-1,k}}{h_x}, \quad \delta^-_y f^n_{j,k} = \frac{f^n_{j,k} - f^n_{j,k-1}}{h_y}, \\ &\nabla^+_h = \begin{pmatrix} \delta^+_x \\ \delta^+_y \end{pmatrix}, \quad \nabla^-_h = \begin{pmatrix} \delta^-_x \\ \delta^-_y \end{pmatrix}, \quad \Delta_h = \nabla^+_h \cdot \nabla^-_h. \end{aligned} \tag{2.22}$$

The operator $\mathcal{A}_t$ is defined as follows

$$\mathcal{A}_t(\prod_{i=1}^n (a_i \phi^n + b_i)^{k_i}) = \sum_{i=1}^n \left[A_t(\prod_{j \ne i} (a_j \phi^n + b_j)^{k_j}) k_i A_t (a_i \phi^n + b_i)^{k_i - 1} a_i A_t \phi^n\right]. \tag{2.23}$$

For example,

$$\mathcal{A}_t \left(|\phi^n|^2 - 1\right)^2 = 4 A_t \left(|\phi^n|^2 - 1\right) A_t \phi^n. \tag{2.24}$$

We drop subscript (j, k) for simplicity in the following. Eliminating the intermediate variables, we arrive at our first LEDRP algorithm as follows.

Algorithm 1 (LEDRP-I)**.**

$$\delta_t^+\phi^n = -\mathcal{M}\Big\{\sum_{p=0}^{\frac{m-1}{2}}\Delta_h^p\mathcal{A}_t\frac{\partial f^n}{\partial\Delta_h^p\phi^n} - \sum_{q=0}^{\frac{m-1}{2}}\nabla_h^-\Delta_h^{q-1}\mathcal{A}_t\frac{\partial f^n}{\partial\nabla_h^+\Delta_h^{q-1}\phi^n}\Big\}. \tag{2.25}$$

We next introduce some lemmas to prove the local energy dissipation preserving property of the algorithm.

Lemma 2.2. For scalar functions f, g, we have the following discrete Leibnitz rule

$$\begin{aligned}\delta_x^+(f\cdot g)_j &= f_j\cdot\delta_x^+ g_j + \delta_x^+ f_j\cdot g_{j+1},\\ \delta_x^+(f\cdot g)_j &= A_x f_j\cdot\delta_x^+ g_j + \delta_x^+ f_j\cdot A_x g_j.\end{aligned} \tag{2.26}$$

The properties hold in other spatial directions and time as well. Based on the lemma, we can derive the following properties.

Lemma 2.3. For scalar function f and vector function $\mathbf{v}$, with $\mathbf{v} = (v_1, v_2)^T$, define $\overline{\mathbf{v}}_{j,k}^n = \left(v_{1_{j-1,k}}^n, v_{2_{j,k-1}}^n\right)$, $\widetilde{\mathbf{v}}_{j,k}^n = \left(v_{1_{j+1,k}}^n, v_{2_{j,k+1}}^n\right)$, based on *Lemma 2.2*, we have

$$\begin{aligned}\nabla_h^+\cdot(f\cdot\overline{\mathbf{v}}) &= \nabla_h^-\cdot\mathbf{v}\cdot f + \mathbf{v}\cdot\nabla_h^+ f,\\ \nabla_h^+\cdot(f\cdot\mathbf{v}) &= \nabla_h^+\cdot\mathbf{v}\cdot f + \widetilde{\mathbf{v}}\cdot\nabla_h^+ f.\end{aligned} \tag{2.27}$$

The next theorem states that *Algorithm 1* obeys a discrete LEDL analogous to that in the continuous case.

Theorem 2.3. Discrete equation (2.19)-(2.21) implies the following discrete LEDL

$$\begin{aligned}&\delta_t^+E^n + \nabla_h^+\cdot\Bigg(\sum_{p=2}^{\frac{m+1}{2}}\sum_{l=0}^{p-2}\left(\delta_t^+\Delta_h^l\phi^n\cdot\nabla_h^-\Delta_h^{p-2-l}A_t k_p^n - \delta_t^+\nabla_h^-\Delta_h^l\phi^n\cdot\Delta_h^{p-2-l}A_t k_p^n\right)\\ &\quad - \sum_{q=1}^{\frac{m+1}{2}}\delta_t^+\Delta_h^{q-1}\phi^n\cdot A_t\overline{\mathbf{h}}_q^n\\ &\quad + \sum_{q=2}^{\frac{m+1}{2}}\sum_{r=0}^{q-2}\left(-\delta_t^+\Delta_h^r\phi^n\cdot\nabla_h^-\nabla_h^-\Delta_h^{q-2-r}A_t\mathbf{h}_q^n + \delta_t^+\nabla_h^-\Delta_h^r\phi^n\cdot\nabla_h^-\Delta_h^{q-2-r}A_t\mathbf{h}_q^n\right)\Bigg)\\ &\quad + A_t\mu^n\mathcal{M}A_t\mu^n = 0,\end{aligned}$$

with E the discrete energy density defined by (2.18).

Next, we discretize the free energy density (2.2), intermediate variables (2.13) - (2.14) and system (2.15) as follows

$$E^n = f\left(\phi^n, \nabla_h^-\phi^n, \Delta_h\phi^n, \nabla_h^-\Delta_h\phi^n, ..., \Delta_h^{\frac{m-1}{2}}\phi^n, \nabla_h^-\Delta_h^{\frac{m-1}{2}}\phi^n\right). \tag{2.28}$$

$$\mathcal{A}_t\frac{\partial f^n}{\partial \phi^n}=A_t k_1^n,\quad \mathcal{A}_t\frac{\partial f^n}{\partial \Delta_h\phi^n}=A_t k_2^n,\quad \cdots,\mathcal{A}_t\frac{\partial f^n}{\partial \Delta_h^{\frac{m-1}{2}}\phi^n}=A_t k_{\frac{m+1}{2}}^n, \tag{2.29}$$

$$\mathcal{A}_t\frac{\partial f^n}{\partial \nabla_h^-\phi^n}=A_t \mathbf{h}_1^n,\quad \mathcal{A}_t\frac{\partial f^n}{\partial \nabla_h^-\Delta_h\phi^n}=A_t \mathbf{h}_2^n,\quad \cdots,\mathcal{A}_t\frac{\partial f^n}{\partial \nabla_h^-\Delta_h^{\frac{m-1}{2}}\phi^n}=A_t \mathbf{h}_{\frac{m+1}{2}}^n, \tag{2.30}$$

$$\begin{cases}\delta_t^+\phi^n=-\mathcal{M}A_t\mu^n,\\ A_t\mu^n=\displaystyle\sum_{p=1}^{\frac{m+1}{2}}\Delta_h^{p-1}A_t k_p^n-\nabla_h^+\sum_{q=1}^{\frac{m+1}{2}}\Delta_h^{q-1}A_t\mathbf{h}_q^n.\end{cases} \tag{2.31}$$

Eliminating the intermediate variables, we arrive at the second LEDRP algorithm as follows.

Algorithm 2 (LEDRP-II)**.**

$$\delta_t^+\phi^n=-\mathcal{M}\Big\{\sum_{p=0}^{\frac{m-1}{2}}\Delta_h^p\mathcal{A}_t\frac{\partial f^n}{\partial\Delta_h^p\phi^n}-\nabla_h^+\sum_{q=0}^{\frac{m-1}{2}}\Delta_h^{q-1}\mathcal{A}_t\frac{\partial f^n}{\partial\nabla_h^-\Delta_h^{q-1}\phi^n}\Big\}. \tag{2.32}$$

Lemma 2.4. For scalar function f and vector function $\mathbf{v}$, with $\mathbf{v}=(v_!,v_2)^T$, we define $\overline{\mathbf{v}}_{j,k}^n=\left(v_{1_{j-1,k}}^n,v_{2_{j,k-1}}^n\right)$, $\widetilde{\mathbf{v}}_{j,k}^n=\left(v_{1_{j+1,k}}^n,v_{2_{j,k+1}}^n\right)$. Based on *Lemma 2.2*, we have

$$\begin{aligned}\nabla_h^-\cdot(f\mathbf{v})&=\nabla_h^-\cdot\mathbf{v}\cdot f+\overline{\mathbf{v}}\cdot\nabla_h^- f,\\ \nabla_h^-\cdot(f\widetilde{\mathbf{v}})&=\nabla_h^+\cdot\mathbf{v}\cdot f+\mathbf{v}\cdot\nabla_h^- f.\end{aligned} \tag{2.33}$$

We can show that *Algorithm 2* obeys an analogous discrete LEDL as well.

Theorem 2.4. Discrete equation (2.29)-(2.31) implies the following discrete LEDL

$$\begin{aligned}&\delta_t^+E^n+\nabla_h^-\cdot\Bigg(\sum_{p=2}^{\frac{m+1}{2}}\sum_{l=0}^{p-2}\left(\delta_t^+\Delta_h^l\phi^n\cdot\nabla_h^+\Delta_h^{p-2-l}A_t k_p^n-\delta_t^+\nabla_h^+\Delta_h^l\phi^n\cdot\Delta_h^{p-2-l}A_t k_p^n\right)\\ &\quad-\sum_{q=1}^{\frac{m+1}{2}}\delta_t^+\Delta_h^{q-1}\phi^n\cdot A_t\widetilde{\mathbf{h}}_q^n\\ &\quad+\sum_{q=2}^{\frac{m+1}{2}}\sum_{r=0}^{q-2}\left(-\delta_t^+\Delta_h^r\phi^n\cdot\Delta_h^{q-1-r}A_t\widetilde{\mathbf{h}}_q^n+\delta_t^+\nabla_h^+\Delta_h^r\phi^n\cdot\nabla_h^+\Delta_h^{q-2-r}A_t\mathbf{h}_q^n\right)\Bigg)\\ &\quad+A_t\mu^n\mathcal{M}A_t\mu^n=0,\end{aligned}$$

with E the discrete energy density defined by (2.28).

Applying the implicit midpoint method in time, the forward Euler method and the implicit midpoint method in space, we derive the third LEDRP algorithm. Here we denote the discrete free energy density (2.2), intermediate variables (2.13) - (2.14) and system (2.15) as follows

$$E^n=f(A_x^mA_y^m\phi^n,\overline{\nabla}_hA_x^{m-1}A_y^{m-1}\phi^n,\overline{\Delta}_hA_x^{m-2}A_y^{m-2}\phi^n,...,\overline{\Delta}_h^{\frac{m-1}{2}}A_xA_y\phi^n,\overline{\nabla}_h\overline{\Delta}_h^{\frac{m-1}{2}}\phi^n), \tag{2.34}$$

$$\begin{cases} \mathcal{A}_t \dfrac{\partial f^n}{\partial A_x^m A_y^m \phi^n} = A_x^m A_y^m A_t k_1^n, \quad \mathcal{A}_t \dfrac{\partial f^n}{\partial \overline{\Delta}_h A_x^{m-2} A_y^{m-2} \phi^n} = A_x^m A_y^m A_t k_2^n, \\ \cdots, \\ \mathcal{A}_t \dfrac{\partial f^n}{\partial \overline{\Delta}_h^{\frac{m-1}{2}} \phi^n} = A_x^m A_y^m A_t k_{\frac{m+1}{2}}^n, \end{cases} \tag{2.35}$$

$$\begin{cases} \mathcal{A}_t \dfrac{\partial f^n}{\partial \overline{\nabla}_h A_x^{m-1} A_y^{m-1} \phi^n} = A_x^m A_y^m A_t \mathbf{h}_1^n, \quad \mathcal{A}_t \dfrac{\partial f^n}{\partial \overline{\nabla}_h \overline{\Delta}_h A_x^{m-3} A_y^{m-3} \phi^n} = A_x^m A_y^m A_t \mathbf{h}_2^n, \\ \cdots, \\ \mathcal{A}_t \dfrac{\partial f^n}{\partial \overline{\nabla}_h \overline{\Delta}_h^{\frac{m-1}{2}} A_x A_y \phi^n} = A_x^m A_y^m A_t \mathbf{h}_{\frac{m+1}{2}}^n, \end{cases} \tag{2.36}$$

$$\begin{cases} \delta_t^+ A_x^m A_y^m \phi^n = -\mathcal{M} A_t A_x^m A_y^m \mu^n, \\ A_t A_x^m A_y^m \mu^n = \displaystyle\sum_{p=1}^{\frac{m+1}{2}} \overline{\Delta}_h^{p-1} A_t A_x^{m-2p+2} A_y^{m-2p+2} k_p^n - \sum_{q=1}^{\frac{m+1}{2}} \overline{\nabla}_h \overline{\Delta}_h^{q-1} A_t A_x^{m-2q+1} A_y^{m-2q+1} \mathbf{h}_q^n, \end{cases} \tag{2.37}$$

where the operators are defined by

$$\overline{\nabla}_h = \begin{pmatrix} \delta_x^+ A_y \\ \delta_y^+ A_x \end{pmatrix}, \quad \overline{\Delta}_h = \overline{\nabla}_h \cdot \overline{\nabla}_h. \tag{2.38}$$

Eliminating the intermediate variables, we arrive at the third LEDRP algorithm.

Algorithm 3 (LEDRP-III)**.**

$$\begin{aligned} \delta_t^+ A_x^{2m} A_y^{2m} \phi^n = -\mathcal{M} \Bigg\{ & \sum_{p=1}^{\frac{m+1}{2}} \overline{\Delta}_h^{p-1} A_x^{m-2p+2} A_y^{m-2p+2} \mathcal{A}_t \frac{\partial f^n}{\partial \overline{\Delta}_h^{p-1} A_x^{m-2p+2} A_y^{m-2p+2} \phi^n} \\ & - \sum_{q=1}^{\frac{m+1}{2}} \overline{\nabla}_h \overline{\Delta}_h^{q-1} A_t A_x^{m-2q+1} A_y^{m-2q+1} \mathcal{A}_t \frac{\partial f^n}{\partial \overline{\nabla}_h \overline{\Delta}_h^{q-1} A_x^{m-2q+1} A_y^{m-2q+1} \phi^n} \Bigg\}. \end{aligned} \tag{2.39}$$

We use a lemma to state relevant discrete Leibnitz rules.

Lemma 2.5. For scalar function f and vectors $\mathbf{u}, \mathbf{v}$, with $\mathbf{u} = (u_1, u_2)^T$, $\mathbf{v} = (v_1, v_2)^T$, we define $\nabla_h^{[i]} \cdot$, $\forall\ i = 1, 2, ..., m$ as following

$$\nabla_h^{[i]} \cdot (f^n \mathbf{v}^n) = \delta_x^+ \left(A_x^{i-1} A_y^i f^n \cdot A_y v_1^n \right) + \delta_y^+ \left(A_x^i A_y^{i-1} f^n \cdot A_x v_2^n \right), \tag{2.40}$$

based on *Lemma 2.2*, the operators have the following property

$$\nabla_h^{[i]} \cdot (f^n \mathbf{u}^n + g^n \mathbf{v}^n) = \nabla_h^{[i]} \cdot (f^n \mathbf{u}^n) + \nabla_h^{[i]} \cdot (g^n \mathbf{v}^n), \tag{2.41}$$

and they obey the following discrete Leibnitz rules

$$\nabla_h^{[i]} \cdot (f^n \mathbf{v}^n) = \overline{\nabla}_h (A_x^{i-1} A_y^{i-1} f^n) \cdot A_x A_y \mathbf{v}^n + A_x^i A_y^i f^n \cdot (\overline{\nabla}_h \cdot \mathbf{v}^n), \ \forall\ i = 1, 2, ..., m. \tag{2.42}$$

We can show that *Algorithm 3* obeys a discrete LEDL.

Theorem 2.5. Discrete equation (2.35)-(2.37) implies the following discrete LEDL

$$\begin{aligned}
&\delta_t^+ E^n - \sum_{q=1}^{\frac{m+1}{2}} \sum_{l=0}^{q-1} \nabla_h^{[m-2l]} \cdot \left(\overline{\Delta}_h^l \delta_t^+ \phi^n \cdot \overline{\Delta}_h^{q-l-1} A_x^{m-2q+2l+1} A_y^{m-2q+2l+1} A_t \mathbf{h}_q^n \right) \\
&+ \sum_{q=2}^{\frac{m+1}{2}} \sum_{l=0}^{q-2} \nabla_h^{[m-2l-1]} \cdot \left(\overline{\nabla}_h \overline{\Delta}_h^l A_t \mathbf{h}_q^n \cdot \overline{\nabla}_h \overline{\Delta}_h^{q-l-2} A_x^{m-2q+2l+2} A_y^{m-2q+2l+2} \delta_t^+ \phi^n \right) \\
&+ \sum_{p=2}^{\frac{m+1}{2}} \left\{ \sum_{r=0}^{p-2} \nabla_h^{[m-2r]} \cdot \left(\overline{\Delta}_h^r \delta_t^+ \phi^n \cdot \overline{\nabla}_h \overline{\Delta}_h^{p-r-2} A_x^{m-2p+2r+2} A_y^{m-2p+2r+2} A_t k_p^n \right. \right. \\
&\left. \left. - \overline{\Delta}_h^r A_t k_p^n \cdot \overline{\nabla}_h \overline{\Delta}_h^{p-r-2} A_x^{m-2p+2r+2} A_y^{m-2p+2r+2} \delta_t^+ \phi^n \right) \right\} \\
&+ A_t A_x^m A_y^m \mu^n \mathcal{M} A_t A_x^m A_y^m \mu^n = 0,
\end{aligned}$$

with E the discrete energy density defined by (2.34).

Remark 2.2. The stencil in algorithm 3 is "too wide" when the order of the spatial derivative is high. It can bring difficulties to efficient numerical implementations if we do not treat the boundaries properly. In addition, when the mobility parameter $\mathcal{M}$ is based on the phase variable, it may become more complex in implementations.

We next turn to another numerical strategy to develop local structure preserving algorithms using the energy quadratization method.

2.2. Energy quadratization (EQ) method

In this subsection, we develop a host of linear algorithms based on the EQ method by reformulating the gradient flow system into another equivalent model with a quadratic free energy functional. Afterwards, we follow the guideline alluded to in the previous subsection to construct the LEDRP algorithms for the model.

2.2.1. EQ reformulation

In many models, the local free energy density $E = f(\phi, \nabla\phi, \nabla^2\phi, ..., \nabla^m\phi)$ can be rewritten into

$$E = \sum_{i=0}^{m} g_i(\epsilon) \cdot |\nabla^i \phi|^2 + b, \tag{2.43}$$

here $b \geq -C_0$ for $\forall\ \phi$, and is a function of $\phi, \nabla\phi, \nabla^2\phi, ..., \nabla^{m-1}\phi$. C_0 is a finite constant, $g_i(\epsilon)$ are functions of constant parameter ϵ, $i = 0, 1, 2...m$. Here we introduce an auxiliary function $q(\phi, \nabla\phi, \nabla^2\phi, ..., \nabla^{m-1}\phi)$ as follows

$$q = \sqrt{b + C_0}, \tag{2.44}$$

then energy density E can be written into

$$E = \sum_{i=0}^{m} g_i(\epsilon) \cdot |\nabla^i \phi|^2 + q^2 - C_0. \tag{2.45}$$

Then, we can write the system (2.8) into the following equivalent form

$$\begin{cases} \phi_t = -\mathcal{M}\mu, \\ \mu = 2\sum\limits_{i=0}^{m}(-1)^i g_i(\epsilon)\cdot\nabla^{2i}\phi + 2\sum\limits_{i=0}^{m-1}(-1)^i\nabla^i\left(q\cdot\dfrac{\partial q}{\partial\nabla^i\phi}\right), \\ q_t = \sum\limits_{i=0}^{m-1}\dfrac{\partial q}{\partial\nabla^i\phi}\cdot\dfrac{\partial\nabla^i\phi}{\partial t}. \end{cases} \tag{2.46}$$

Lemma 2.6. From *Lemma 2.1*, we derive the following identity

$$(-1)^i\cdot\phi_t\cdot\nabla^i\left(f\frac{\partial f}{\partial\nabla^i\phi}\right) = (-1)^i\nabla\left(\sum_{k=1}^{i}(-1)^{k-1}\nabla^{k-1}\phi_t\cdot\nabla^{i-k}\left(f\frac{\partial f}{\partial\nabla^i\phi}\right)\right) + \nabla^i\phi_t\cdot\left(f\frac{\partial f}{\partial\nabla^i\phi}\right). \tag{2.47}$$

The transport equation for the energy density of the equivalent system (2.46) is given in the following.

Theorem 2.6. Model (2.46) admits the following LEDL

$$\begin{aligned} &\frac{\mathrm{d}E}{\mathrm{d}t} + 2\sum_{i=0}^{m}\left[(-1)^i g_i(\epsilon)\nabla\left((-1)^{k-1}\sum_{k=1}^{i}\nabla^{k-1}\phi_t\cdot\nabla^{2i-k}\phi\right)\right] \\ &\quad + 2\sum_{i=0}^{m-1}\left[(-1)^i\nabla\left(\sum_{k=1}^{i}(-1)^{k-1}\nabla^{k-1}\phi_t\cdot\nabla^{i-k}\left(q\frac{\partial q}{\partial\nabla^i\phi}\right)\right)\right] + \mu\mathcal{M}\mu = 0, \end{aligned} \tag{2.48}$$

with E the energy density defined by (2.45).

2.2.2. Reformulation of the EQ reformulated system

Analogous to the previous practice, we first introduce two groups of intermediate variables to reformulate the system (2.46) into an equivalent form, which makes it feasible for us to construct the LEDRP algorithm. The intermediate variables are listed in the following:

$$q\frac{\partial q}{\partial\phi} = k_1, \quad q\frac{\partial q}{\partial\Delta\phi} = k_2, \quad q\frac{\partial q}{\partial\Delta^2\phi} = k_3, \quad ..., q\frac{\partial q}{\partial\Delta^{\frac{m-1}{2}}\phi} = k_{\frac{m+1}{2}}, \tag{2.49}$$

$$q\frac{\partial q}{\partial\nabla\phi} = \mathbf{h}_1, \quad q\frac{\partial q}{\partial\nabla\Delta\phi} = \mathbf{h}_2, \quad q\frac{\partial q}{\partial\nabla\Delta^2\phi} = \mathbf{h}_3, \quad ..., q\frac{\partial q}{\partial\nabla\Delta^{\frac{m-3}{2}}\phi} = \mathbf{h}_{\frac{m-1}{2}}. \tag{2.50}$$

With the intermediate variables, the model is rewritten into

$$\begin{cases} \phi_t = -\mathcal{M}\mu, \\ \mu = 2\sum\limits_{i=0}^{m}(-1)^i g_i(\epsilon)\cdot\nabla^{2i}\phi + 2\left[\sum\limits_{p=1}^{\frac{m-1}{2}}\Delta^{p-1}k_p - \sum\limits_{q=1}^{\frac{m-3}{2}}\nabla\Delta^{q-1}\mathbf{h}_q\right], \\ q_t = \sum\limits_{i=0}^{m-1}\dfrac{\partial q}{\partial\nabla^i\phi}\cdot\dfrac{\partial\nabla^i\phi}{\partial t}. \end{cases} \tag{2.51}$$

Theorem 2.7. Model (2.51) admits the following LEDL

$$\partial_t E + \nabla \cdot \left[\sum_{p=2}^{\frac{m+1}{2}} \sum_{l=0}^{2p-3} \left((-1)^{l+1} \nabla^l k_p \cdot \nabla^{2p-3-l} \phi_t \right) + \sum_{q=1}^{\frac{m-1}{2}} \sum_{r=0}^{2q-2} \left((-1)^{r+1} \nabla^r \mathbf{h}_q \cdot \nabla^{2q-1-r} \phi_t \right) \right] + \mu \mathcal{M} \mu = 0, \tag{2.52}$$

with E the energy density defined by (2.45).

2.2.3. Local structure preserving algorithms for the equivalent system

We discretize free energy density (2.45), the intermediate variables (2.49) - (2.50) and system (2.51) the same way as we did previously. We discuss the specific discretization method in detail, and present the precise algorithms and the deduced transport equations of the energy density in Appendix B.

First, we discretize the energy density (2.45), intermediate variables (2.49) - (2.50) and model (2.51) as follows

$$E^n = \sum_{i=0\ 2|i}^{m} g_i(\epsilon) \cdot |\Delta_h^{\frac{i}{2}} \phi^n|^2 + \sum_{i=0\ 2\nmid i}^{m} g_i(\epsilon) \cdot |\nabla_h^+ \Delta_h^{\frac{i-1}{2}} \phi^n|^2 + \left| q^n \left(\phi^n, \nabla_h^+ \phi^n, \Delta_h \phi^n, ..., \Delta_h^{\frac{m-1}{2}} \right) \right|^2 - C_0, \tag{2.53}$$

$$A_t q^n \frac{\partial q^{n,\star}}{\partial \phi^{n,\star}} = A_t k_1, \quad A_t q^n \frac{\partial q^{n,\star}}{\partial \Delta_h \phi^{n,\star}} = A_t k_2, \quad ..., A_t q^n \frac{\partial q^{n,\star}}{\partial \Delta_h^{\frac{m-1}{2}} \phi^{n,\star}} = A_t k_{\frac{m+1}{2}}, \tag{2.54}$$

$$A_t q^n \frac{\partial q^{n,\star}}{\partial \nabla_h^+ \phi^{n,\star}} = A_t \mathbf{h}_1, \quad A_t q^n \frac{\partial q^{n,\star}}{\partial \nabla_h^+ \Delta_h \phi^{n,\star}} = A_t \mathbf{h}_2, \quad ..., A_t q^n \frac{\partial q^{n,\star}}{\partial \nabla_h^+ \Delta_h^{\frac{m-3}{2}} \phi^{n,\star}} = A_t \mathbf{h}_{\frac{m-1}{2}}, \tag{2.55}$$

where $\phi^{n,\star} = (3\phi^n - \phi^{n-1})/2$,

$$\begin{cases} \delta_t^+ \phi^n = -\mathcal{M} A_t \mu^n, \\ A_t \mu^n = 2 \sum_{i=0}^{m} (-1)^i g_i(\epsilon) \cdot A_t \Delta_h^i \phi^n + 2 \left[\sum_{p=1}^{\frac{m+1}{2}} \Delta_h^{p-1} A_t k_p - \sum_{q=1}^{\frac{m-1}{2}} \nabla_h^- \Delta_h^{q-1} A_t \mathbf{h}_q \right], \\ \delta_t^+ q^n = \sum_{p=1}^{\frac{m+1}{2}} \frac{\partial q^{n,\star}}{\partial \Delta_h^{p-1} \phi^{n,\star}} \cdot \delta_t^+ \Delta_h^{p-1} \phi^n + \sum_{q=1}^{\frac{m-1}{2}} \frac{\partial q^{n,\star}}{\partial \nabla_h^+ \Delta_h^{q-1} \phi^{n,\star}} \cdot \delta_t^+ \nabla_h^+ \Delta_h^{q-1} \phi^n, \end{cases} \tag{2.56}$$

this yields the first LEDRP based on the EQ technique.

Similarly, we develop our second LEDRP based on EQ technique. We define the discrete energy density (2.45), intermediate variables (2.49) - (2.50), model (2.51) as follows

$$E^n = \sum_{i=0\ 2|i}^{m} g_i(\epsilon) \cdot |\Delta_h^{\frac{i}{2}} \phi^n|^2 + \sum_{i=0\ 2\nmid i}^{m} g_i(\epsilon) \cdot |\nabla_h^- \Delta_h^{\frac{i-1}{2}} \phi^n|^2 + \left| q^n \left(\phi^n, \nabla_h^- \phi^n, \Delta_h \phi^n, ..., \Delta_h^{\frac{m-1}{2}} \phi^n \right) \right|^2 - C_0, \tag{2.57}$$

$$A_t q^n \frac{\partial q^{n,\star}}{\partial \phi^{n,\star}} = A_t k_1, \quad A_t q^n \frac{\partial q^{n,\star}}{\partial \Delta_h \phi^{n,\star}} = A_t k_2, \quad ..., A_t q^n \frac{\partial q^{n,\star}}{\partial \Delta_h^{\frac{m-1}{2}} \phi^{n,\star}} = A_t k_{\frac{m+1}{2}}, \tag{2.58}$$

$$A_t q^n \frac{\partial q^{n,\star}}{\partial \nabla_h^- \phi^{n,\star}} = A_t \mathbf{h}_1, \quad A_t q^n \frac{\partial q^{n,\star}}{\partial \nabla_h^- \Delta_h \phi^{n,\star}} = A_t \mathbf{h}_2, \quad ..., A_t q^n \frac{\partial q^{n,\star}}{\partial \nabla_h^- \Delta_h^{\frac{m-3}{2}} \phi^{n,\star}} = A_t \mathbf{h}_{\frac{m-1}{2}}, \tag{2.59}$$

$$\begin{cases} \delta_t^+ \phi^n = -\mathcal{M} A_t \mu^n, \\ A_t \mu^n = 2 \sum_{i=0}^{m} (-1)^i g_i(\epsilon) \cdot A_t \Delta_h^i \phi^n + 2 \left[\sum_{p=1}^{\frac{m+1}{2}} \Delta_h^{p-1} A_t k_p - \sum_{q=1}^{\frac{m-1}{2}} \nabla_h^+ \Delta_h^{q-1} A_t \mathbf{h}_q \right], \\ \delta_t^+ q^n = \sum_{p=1}^{\frac{m+1}{2}} \frac{\partial q^{n,\star}}{\partial \Delta_h^{p-1} \phi^{n,\star}} \cdot \delta_t^+ \Delta_h^{p-1} \phi^n + \sum_{q=1}^{\frac{m-1}{2}} \frac{\partial q^{n,\star}}{\partial \nabla_h^- \Delta_h^{q-1} \phi^{n,\star}} \cdot \delta_t^+ \nabla_h^- \Delta_h^{q-1} \phi^n. \end{cases} \tag{2.60}$$

Applying the implicit midpoint method in time, the forward Euler method and implicit midpoint method in space, we derive yet another LEDRP algorithm based on the EQ method. We denote the discrete free energy density (2.45), intermediate variables (2.49) - (2.50), model (2.51) as follows

$$\begin{aligned} E^n = & \sum_{i=0}^{m} g_i(\epsilon) \cdot |\overline{\nabla}_h^i A_x^{m-i} A_y^{m-i} \phi^n|^2 \\ & + \left| q^n \left(A_x^m A_y^m \phi, \overline{\nabla}_h A_x^{m-1} A_y^{m-1} \phi^n, \overline{\Delta}_h A_x^{m-2} A_y^{m-2} \phi^n, ..., \overline{\Delta}_h^{\frac{m-1}{2}} A_x A_y \phi^n \right) \right|^2 - C_0, \end{aligned} \tag{2.61}$$

$$\begin{cases} A_t A_x^m A_y^m q^n \dfrac{\partial q^{n,\star}}{\partial A_x^m A_y^m \phi^{n,\star}} = A_t A_x^m A_y^m k_1^n, \\ A_t A_x^m A_y^m q^n \dfrac{\partial q^{n,\star}}{\partial \overline{\Delta}_h A_x^{m-2} A_y^{m-2} \phi^{n,\star}} = A_t A_x^m A_y^m k_2^n, \\ ... \\ A_t A_x^m A_y^m q^n \dfrac{\partial q^{n,\star}}{\partial \overline{\Delta}_h^{\frac{m-1}{2}} A_x A_y \phi^{n,\star}} = A_t A_x^m A_y^m k_{\frac{m+1}{2}}^n, \end{cases} \tag{2.62}$$

$$\begin{cases} A_t A_x^m A_y^m q^n \dfrac{\partial q^{n,\star}}{\partial \overline{\nabla}_h A_x^{m-1} A_y^{m-1} \phi^{n,\star}} = A_t A_x^m A_y^m \mathbf{h}_1^n, \\ A_t A_x^m A_y^m q^n \dfrac{\partial q^{n,\star}}{\partial \overline{\nabla}_h \overline{\Delta}_h A_x^{m-3} A_y^{m-3} \phi^{n,\star}} = A_t A_x^m A_y^m \mathbf{h}_2^n, \\ ... \\ A_t A_x^m A_y^m q^n \dfrac{\partial q^{n,\star}}{\partial \overline{\nabla}_h \overline{\Delta}_h^{\frac{m-3}{2}} A_x^2 A_y^2 \phi^{n,\star}} = A_t A_x^m A_y^m \mathbf{h}_{\frac{m-1}{2}}^n, \end{cases} \tag{2.63}$$

$$
\begin{cases}
\delta_t^+ A_x^m A_y^m \phi^n = -\mathcal{M} A_t A_x^m A_y^m \mu^n, \\
A_t A_x^m A_y^m \mu^n = 2 \sum_{i=0}^{m} (-1)^i g_i(\epsilon) \cdot A_t A_x^{m-2i} A_y^{m-2i} \overline{\Delta}_h^i \phi^n \\
\qquad\qquad + 2 \left[\sum_{p=1}^{\frac{m+1}{2}} \overline{\Delta}_h^{p-1} A_t A_x^{m-2p+2} A_y^{m-2p+2} k_p^n - \sum_{q=1}^{\frac{m-1}{2}} \overline{\nabla}_h \overline{\Delta}_h^{q-1} A_t A_x^{m-2q+2} A_y^{m-2q+2} \mathbf{h}_q^n \right], \\
\delta_t^+ A_x^m A_y^m q^n = \sum_{p=1}^{\frac{m+1}{2}} \frac{\partial q^{n,\star}}{\partial \overline{\Delta}_h^{p-1} A_x^{m-2p+2} A_y^{m-2p+2} \phi^{n,\star}} \cdot \delta_t^+ \overline{\Delta}_h^{p-1} A_x^{m-2p+2} A_y^{m-2p+2} \phi^n \\
\qquad\qquad + \sum_{q=1}^{\frac{m-1}{2}} \frac{\partial q^{n,\star}}{\partial \overline{\nabla}_h \overline{\Delta}_h^{q-1} A_x^{m-2q+2} A_y^{m-2q+2} \phi^{n,\star}} \cdot \delta_t^+ \overline{\nabla}_h \overline{\Delta}_h^{q-1} A_x^{m-2q+2} A_y^{m-2q+2} \phi^n.
\end{cases}
\tag{2.64}
$$

These three local energy dissipation law preserving numerical algorithms are linear for the general gradient flow model. In practice, we also encounter such equation systems driven by a forcing term like in some crystal growth models. The driven gradient flow model may not have a definitive energy dissipation law. But, the time evolutionary equation for the energy density exists, which is unavoidably affected by the forcing term. Next, we discuss how to design numerical algorithms that preserve the energy transport equation in the discretized gradient flow system with forcing terms.

3. Local energy dissipation rate preserving (LEDRP) algorithms for driven gradient flows

Given free energy (2.2), a driven gradient flow system reads as follows

$$
\begin{cases}
\phi_t = -\mathcal{M}\mu + g(\phi), \\
\mu = \dfrac{\delta \mathcal{E}}{\delta \phi},
\end{cases}
\tag{3.1}
$$

with $g(\phi)$ the forcing term. The framework for developing LEDRPs for the gradient flow models can be applied equally well to this model.

With free energy density E defined in (2.2), driven gradient flow (3.1) has the following energy density transport equation

$$
\partial_t E + \sum_{i=0}^{m} (-1)^i \nabla \cdot \left(\sum_{k=1}^{i} (-1)^{k-1} \nabla^{k-1} \phi_t \cdot \nabla^{i-k} \frac{\partial f}{\partial \nabla^i \phi} \right) E + \mu \mathcal{M} \mu = g(\phi) \mu. \tag{3.2}
$$

Compared with the local energy density transport equation of an "unforced" gradient flow model, we add forcing term $'g(\phi)\mu'$ to the energy density transport equation here. Hence, designing an algorithm to preserve the structure of this equation upon discretization should be focused primarily on how the forcing term is discretized to match the existing LEDRP algorithms. In fact, the discretization of the forcing term is much less strict than the other terms in the energy density

transport equation, so it is a simple matter for us to extend the previous LEDRP algorithms to the driven system. Practically, we just need to discretize the forcing term $'g(\phi)'$ to ensure the appropriate order of the entire discretized system.

Introducing the same intermediate variables in gradient flow system (2.8), we arrive at

$$\begin{cases} \phi_t = -\mathcal{M}\mu + g(\phi), \\ \mu = \displaystyle\sum_{p=1}^{\frac{m+1}{2}} \Delta^{p-1} k_p - \sum_{q=1}^{\frac{m+1}{2}} \nabla \Delta^{q-1} \mathbf{h}_q. \end{cases} \tag{3.3}$$

Apparently, system (3.3) admits an energy density transport equation.

Theorem 3.1. Model (3.3) admits the following energy transport equation

$$\begin{aligned} &\partial_t E + \nabla \cdot \left[\sum_{p=2}^{\frac{m+1}{2}} \sum_{l=0}^{2p-3} \left((-1)^{l+1} \nabla^l k_p \cdot \nabla^{2p-3-l} \phi_t \right) + \sum_{q=1}^{\frac{m+1}{2}} \sum_{r=0}^{2q-2} \left((-1)^{r+1} \nabla^r \mathbf{h}_q \cdot \nabla^{2q-2-r} \phi_t \right) \right] \\ &\quad + \mu \mathcal{M} \mu = g(\phi) \mu, \end{aligned} \tag{3.4}$$

with E the energy density defined by (2.2).

We use the same methods proposed in §2 to derive LEDRP algorithms for the driven system. In particular, we apply the implicit midpoint method in time to forcing term $g(\phi)$. The local energy dissipation rate property (3.4) for the driven gradient flow can be preserved strictly at the discrete level. The details are given in Appendix C.

Applying the EQ method to the system, we arrive at

$$\begin{cases} \phi_t = -\mathcal{M}\mu + g(\phi), \\ \mu = 2 \displaystyle\sum_{i=0}^{m} (-1)^i g_i(\epsilon) \cdot \nabla^{2i} \phi + 2 \sum_{i=0}^{m-1} (-1)^i \nabla^i \left(q \cdot \frac{\partial q}{\partial \nabla^i \phi} \right), \\ q_t = \displaystyle\sum_{i=0}^{m-1} \frac{\partial q}{\partial \nabla^i \phi} \cdot \frac{\partial \nabla^i \phi}{\partial t}, \end{cases} \tag{3.5}$$

which has the following energy density transport equation

$$\begin{aligned} &\frac{\mathrm{d}E}{\mathrm{d}t} + 2 \sum_{i=0}^{m} \left[(-1)^i g_i(\epsilon) \nabla \left((-1)^{k-1} \sum_{k=1}^{i} \nabla^{k-1} \phi_t \cdot \nabla^{2i-k} \phi \right) \right] \\ &\quad + 2 \sum_{i=0}^{m-1} \left[(-1)^i \nabla \left(\sum_{k=1}^{i} (-1)^{k-1} \nabla^{k-1} \phi_t \cdot \nabla^{i-k} \left(q \frac{\partial q}{\partial \nabla^i \phi} \right) \right) \right] + \mu \mathcal{M} \mu = g(\phi) \mu, \end{aligned} \tag{3.6}$$

with E the energy density defined by (2.45).

To derive linear LEDRP algorithms for the driven gradient flow system formulated in EQ form, we can discretize "$g(\phi)$" either explicitly or semi-implicitly. We give the LEDRP algorithms in Appendix C.

4. Algorithms for the graphene growth model

In this section, we apply the LEDRP algorithms to the epitaxial graphene growth model [21]. Firstly, we rewrite the coupled model into the form of a driven gradient flow using the Onsager principle and derive the energy density transport equation. Secondly, we construct several LEDRP algorithms based on the EQ technique. We present the algorithms in 2D space.

4.1. Graphene growth model

We first reformulate the model from a new perspective following the Onsager principle for nonequilibrium thermodynamics. We define the free energy as follows

$$\mathcal{E}(\phi,u)=\int_{\Omega}\left[\frac{1}{2}\epsilon^2\xi_{s,n}(\Theta)^2|\nabla\phi|^2+f(\phi)+\frac{u^2}{2\tau_v}+\frac{1}{2}\nabla u^T\cdot D\cdot\nabla u\right]\mathrm{d}\mathbf{x}, \tag{4.1}$$

where ϵ is the interface width, ϕ is the phase variable ($\phi=1$ corresponds to the thin field crystal domain), while u is the concentration of the diatom, $\xi_{s,n}$ is the coefficient of the anisotropic conformation entropy defined by

$$\xi_{s,n}(\Theta)=1+\epsilon_{s,n}\cos(n\Theta), \tag{4.2}$$

in which $\epsilon_{s,n}$ is the step energy anisotropy (n-fold), Θ is the orientation of the normal to the edge given by $\Theta=\arctan\frac{\partial_y\phi}{\partial_x\phi}$, τ_v is a desorption time parameter and D is the diffusivity tensor defined by

$$D=\begin{pmatrix}1+\delta\cos(2\Psi) & \delta\sin(2\Psi)\\ \delta\sin(2\Psi) & 1-\delta\cos(2\Psi)\end{pmatrix}, \tag{4.3}$$

with δ the diffusion anisotropy, and Ψ the rotated angle of the principle diffusion axes. The functions $f(\phi)$ and $g(\phi)$ are chosen as $f(\phi)=\phi^2(1-\phi)^2/4$, $g(\phi)=\phi^3(10-15\phi+6\phi^2)/120$, respectively. Function $f(\phi)$ serves as the double well bulk free energy and parameterizes the deposition flux at the interface and $g(\phi)$ is a switch function monotonically connecting 0 to 1 in the phase space ϕ and $g'(\phi)=f(\phi)$.

The time derivative of the free energy (4.1) is calculated as follows

$$\begin{aligned}\frac{\mathrm{d}\mathcal{E}}{\mathrm{d}t}=&\left(-\epsilon^2\nabla\cdot\left[\xi_{s,n}(\Theta)^2\nabla\phi\right]+\epsilon^2\partial_x\left[\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_y\phi\right]-\epsilon^2\partial_y\left[\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_x\phi\right]+f'(\phi),\phi_t\right)\\&+\left(\frac{u}{\tau_v}-\nabla\cdot\left(D\cdot\nabla u\right),u_t\right)\\&+\int_{\partial\Omega}\left(-\epsilon^2\xi_{s,n}(\Theta)^2\nabla\phi\cdot\phi_t+\begin{pmatrix}\epsilon^2\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_y\phi\\ -\epsilon^2\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_x\phi\end{pmatrix}\cdot\phi_t-D\cdot\nabla u\cdot u_t\right)\cdot\mathbf{n}\mathrm{d}S,\end{aligned} \tag{4.4}$$

which is consisted of the bulk and the boundary contribution. The variation of the free energy with respect to the two thermodynamical variables ϕ and u, also known as the chemical potentials, are

given by

$$\begin{cases} \dfrac{\delta \mathcal{E}}{\delta \phi} = -\epsilon^2 \nabla \cdot \left[\xi_{s,n}(\Theta)^2 \nabla \phi\right] + \epsilon^2 \partial_x \left[\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_y \phi\right] - \epsilon^2 \partial_y \left[\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_x \phi\right] + f'(\phi), \\ \dfrac{\delta \mathcal{E}}{\delta u} = \dfrac{u}{\tau_v} - \nabla \cdot (D \cdot \nabla u)\,. \end{cases} \tag{4.5}$$

With the following adiabatic boundary conditions

$$\left(-\epsilon^2 \xi_{s,n}(\Theta)^2 \nabla \phi \cdot \phi_t + \begin{pmatrix} \epsilon^2 \xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_y \phi \\ -\epsilon^2 \xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_x \phi \end{pmatrix} \cdot \phi_t - D \cdot \nabla u \cdot u_t \right) \cdot \mathbf{n} = 0, \tag{4.6}$$

the surface energy flux vanishes at the boundary so that the time derivative of the free energy is given exclusively by the bulk effect:

$$\begin{aligned} \frac{\mathrm{d}\mathcal{E}}{\mathrm{d}t} = &\Big(-\epsilon^2 \nabla \cdot \left[\xi_{s,n}(\Theta)^2 \nabla \phi\right] + \epsilon^2 \partial_x \left[\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_y \phi\right] - \epsilon^2 \partial_y \left[\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_x \phi\right] + f'(\phi), \phi_t \Big) \\ &+ \left(\frac{u}{\tau_v} - \nabla \cdot (D \cdot \nabla u)\,, u_t \right). \end{aligned} \tag{4.7}$$

Notice that the following boundary conditions

$$\nabla \phi \cdot \mathbf{n} = \begin{pmatrix} \partial_y \phi \\ -\partial_x \phi \end{pmatrix} \cdot \mathbf{n} = \nabla u \cdot \mathbf{n} = 0 \tag{4.8}$$

imply (4.6). Applying the generalized Onsager principle, we can formulate the dynamical equations in the graphene growth model as follows

$$\begin{cases} \mathbf{u}_t = -\boldsymbol{M}(\Theta)\boldsymbol{\mu} + \mathbf{g}(\mathbf{u}), \quad \mathbf{u} = (\phi, u)^T, \\ \boldsymbol{\mu} = \begin{pmatrix} -\epsilon^2 \nabla \cdot \left[\xi_{s,n}(\Theta)^2 \nabla \phi\right] + \epsilon^2 \partial_x \left[\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_y \phi\right] - \epsilon^2 \partial_y \left[\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_x \phi\right] + f'(\phi) \\ \frac{u}{\tau_v} - \nabla \cdot (D \cdot \nabla u) \end{pmatrix}, \\ \mathbf{g}(\mathbf{u}) = \boldsymbol{M}(\Theta) \begin{pmatrix} \epsilon \lambda g'(\phi) u \\ f \end{pmatrix}, \quad \boldsymbol{M}(\Theta) = \begin{pmatrix} \frac{1}{\alpha \tau(\Theta) \epsilon^2} & 0 \\ -\frac{1}{\alpha \tau(\Theta) \epsilon^2} & 1 \end{pmatrix}, \\ \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}), \\ \nabla \phi \cdot \mathbf{n} = \begin{pmatrix} \partial_y \phi \\ -\partial_x \phi \end{pmatrix} \cdot \mathbf{n} = \nabla u \cdot \mathbf{n} = 0, \ \mathbf{x} \in \partial\Omega, \end{cases} \tag{4.9}$$

where $\tau(\Theta) = \tau_0(\Theta) + \tau_1(\Theta)\epsilon$, $\tau_0(\Theta) = \xi_{s,n}(\Theta)\xi_{k,n}(\Theta)$, $\tau_1(\Theta) = a_2 \frac{\xi_{s,n}(\Theta)^2}{\beta\{1+\delta\cos[2(\Psi-\Theta)]\}}$, with $a_2 = 47\sqrt{2}/60$. $\xi_{k,n}(\Theta)$ is the kinetic energy anisotropy defined as

$$\xi_{k,n}(\Theta) = 1 + \epsilon_{k,n} \cos(n\Theta - n\Theta_0), \tag{4.10}$$

225 Θ_0 is a given angle. The model parameters α and λ are defined as $\lambda = \frac{a_1}{d_0}$, $\alpha = \frac{\beta}{d_0}$, with $a_1 = 10\sqrt{2}$.

In componentwise forms, the phase field graphene growth model given by system (4.9) together with the initial and boundary conditions is summarized as follows

$$\begin{cases} \alpha\tau(\Theta)\epsilon^2\phi_t = \epsilon^2\nabla\cdot\left[\xi_{s,n}(\Theta)^2\nabla\phi\right] - \epsilon^2\partial_x\left[\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_y\phi\right] + \epsilon^2\partial_y\left[\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_x\phi\right] - f'(\phi) \\ \qquad\qquad + \epsilon\lambda g'(\phi)u, \\ u_t = -\phi_t - \dfrac{u}{\tau_v} + \nabla\cdot(D\cdot\nabla u) + f, \\ \phi(\mathbf{x},0) = \phi_0(\mathbf{x}), \\ u(\mathbf{x},0) = u_0(\mathbf{x}), \\ \nabla\phi\cdot\mathbf{n} = \begin{pmatrix} \partial_y\phi \\ -\partial_x\phi \end{pmatrix}\cdot\mathbf{n} = \nabla u\cdot\mathbf{n} = 0,\ \ \mathbf{x}\in\partial\Omega, \end{cases} \tag{4.11}$$

For any smooth domain U in Ω, , the time derivative of the free energy is given by

$$\frac{\mathrm{d}\mathcal{E}}{\mathrm{d}t} = \int_U \frac{\delta\mathcal{E}}{\delta\mathbf{u}}\cdot\mathbf{u}_t\mathrm{d}\mathbf{x} = -\int_U (\frac{\delta\mathcal{E}}{\delta\mathbf{u}})^T\cdot\boldsymbol{M}\cdot\frac{\delta\mathcal{E}}{\delta\mathbf{u}}\mathrm{d}\mathbf{x} + \int_U \frac{\delta\mathcal{E}}{\delta\mathbf{u}}\cdot\mathbf{g}\mathrm{d}\mathbf{x} - \int_{\partial U}\mathbf{n}\cdot\mathbf{h}\mathrm{d}s, \tag{4.12}$$

where $\mathbf{h}$ is the terms generated when computing the variation vector, known as the energy flux, given below. Corresponding to the free energy given in (4.1), the local free energy density is identified as

$$E(\phi,u) = \frac{1}{2}\epsilon^2\xi_{s,n}(\Theta)^2|\nabla\phi|^2 + f(\phi) + \frac{u^2}{2\tau_v} + \frac{1}{2}\nabla u^T\cdot D\cdot\nabla u. \tag{4.13}$$

Energy flux $\mathbf{h}$ is identified by the following theorem.

Theorem 4.1. System (4.9) has a local energy density transport equation:

$$\frac{\mathrm{d}E}{\mathrm{d}t} + \nabla\cdot\mathbf{h} + \boldsymbol{\mu}^T\cdot\boldsymbol{M}(\Theta)\cdot\boldsymbol{\mu} = \mathbf{g}(\mathbf{u})\cdot\boldsymbol{\mu}, \tag{4.14}$$

with E the free energy density defined by (4.13) and the energy flux is given by

$$\mathbf{h} = \left[-\epsilon^2\xi_{s,n}(\Theta)^2\nabla\phi\cdot\phi_t + \begin{pmatrix} \epsilon^2\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_y\phi \\ -\epsilon^2\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_x\phi \end{pmatrix}\cdot\phi_t - D\cdot\nabla u\cdot u_t\right]. \tag{4.15}$$

4.1.1. EQ reformulation

We reformulate the system based on the EQ technique, i.e., we first rewrite the energy density given by (4.13) as a quadratic form:

$$E(\phi,u) = \frac{1}{2}\epsilon^2\sqrt{|\xi_{s,n}(\Theta)\nabla\phi|^2 + A}^2 - \frac{1}{2}\epsilon^2 A + \frac{1}{4}[\phi(\phi-1)]^2 + \frac{u^2}{2\tau_v} + \frac{1}{2}\nabla u^T\cdot D\cdot\nabla u, \tag{4.16}$$

230 where A is a positive constant to ensure the positivity of $|\xi_{s,n}(\Theta)\nabla\phi|^2 + A$.

We then define scalar functions $U(\phi)$ and $V(\phi)$ as follows

$$\begin{cases} U(\phi) = \sqrt{|\xi_{s,n}(\Theta)\nabla\phi|^2 + A}, \\ V(\phi) = \phi\,(1-\phi)\,. \end{cases} \tag{4.17}$$

Thus, the local free energy density can be quadratized in the three variables (U, V, u) as follows

$$E(U,V,u) = \frac{1}{2}\epsilon^2 U^2 - \frac{1}{2}\epsilon^2 A + \frac{1}{4}V^2 + \frac{u^2}{2\tau_v} + \frac{1}{2}\nabla u^T \cdot D \cdot \nabla u. \tag{4.18}$$

We then rewrite system (4.9) into an equivalent form as follows

$$\begin{cases} \mathbf{u}_t = -\boldsymbol{M}(\Theta)\cdot\boldsymbol{\mu} + \mathbf{g}(\mathbf{u}), \\ \boldsymbol{\mu} = \begin{pmatrix} -\epsilon^2\nabla\cdot(U(\phi)\boldsymbol{R}(\phi)) + \frac{1}{2}V(\phi)P(\phi) \\ \frac{u}{\tau_v} - \nabla\cdot(D\cdot\nabla u) \end{pmatrix}, \\ \partial_t U = \boldsymbol{R}(\phi)\cdot\nabla\phi_t, \\ \partial_t V = P(\phi)\cdot\phi_t, \end{cases} \tag{4.19}$$

where

$$\begin{cases} \boldsymbol{R}(\phi) = \dfrac{1}{U(\phi)}\left\{\xi_{s,n}(\Theta)^2\nabla\phi + \begin{pmatrix} -\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_y\phi \\ \xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_x\phi \end{pmatrix}\right\}, \\ P(\phi) = 1 - 2\phi. \end{cases} \tag{4.20}$$

The initial conditions are given by

$$\phi(\mathbf{x},0) = \phi_0(\mathbf{x}), \quad u(\mathbf{x},0) = u_0(\mathbf{x}), \quad U(\phi_0) = \sqrt{|\xi_s(\Theta(\phi_0))\nabla\phi_0|^2 + A}, \quad V(\phi_0) = \phi_0\,(1-\phi_0)\,. \tag{4.21}$$

With the newly introduced variables, system (4.19) - (4.21) admits a reformulated energy density transport equation.

Theorem 4.2. System (4.19) - (4.20) have the following energy density transport equation

$$\frac{\mathrm{d}E}{\mathrm{d}t} + \nabla\cdot\left(-\epsilon^2 U(\phi)\boldsymbol{R}(\phi)\phi_t - D\cdot\nabla u\cdot u_t\right) + \boldsymbol{\mu}^T\cdot\boldsymbol{M}(\Theta)\cdot\boldsymbol{\mu} = \mathbf{g}(\mathbf{u})\cdot\boldsymbol{\mu}, \tag{4.22}$$

with E the free energy density defined by (4.18).

Following the paradigm established in the previous section, we introduce intermediate variables to reformulate system (4.19) - (4.21) into one with lower spatial derivatives suitable for constructing local structure preserving algorithms. The intermediate variables are introduced as follows

$$\begin{pmatrix} \frac{1}{2}V(\phi)P(\phi) \\ \frac{u}{\tau_v} \end{pmatrix} = \boldsymbol{k}, \quad \begin{pmatrix} \epsilon^2 U(\phi)\boldsymbol{R}(\phi) \\ D\cdot\nabla u \end{pmatrix} = \boldsymbol{H}. \tag{4.23}$$

Then the system becomes

$$\begin{cases} \mathbf{u}_t = -\boldsymbol{M}(\Theta)\cdot\boldsymbol{\mu} + \mathbf{g}(\mathbf{u}), \\ \boldsymbol{\mu} = \boldsymbol{k} - \nabla\cdot\boldsymbol{H}, \\ \partial_t U = \boldsymbol{R}(\phi)\cdot\nabla\phi_t, \\ \partial_t V = P(\phi)\cdot\phi_t. \end{cases} \tag{4.24}$$

System (4.23) - (4.24) admits the following energy transport equation

$$\frac{\mathrm{d}E}{\mathrm{d}t} + \nabla\cdot(-\boldsymbol{H}\cdot\mathbf{u}) + \boldsymbol{\mu}^T\cdot\boldsymbol{M}(\Theta)\cdot\boldsymbol{\mu} = \mathbf{g}(\mathbf{u})\cdot\boldsymbol{\mu}, \tag{4.25}$$

with E the free energy density defined by (4.18).

4.2. Local energy dissipation rate preserving (LEDRP) algorithms

We now construct our local energy dissipation rate preserving algorithms for system (4.23) - (4.24).

4.2.1. LEDRP algorithm I

We first discretize the free energy density (4.18) as follows

$$E(U^n, V^n, u^n) = \frac{1}{2}\epsilon^2|U^n|^2 - \frac{1}{2}\epsilon^2 A + \frac{1}{4}|V^n|^2 + \frac{|u^n|^2}{2\tau_v} + \frac{1}{2}(\nabla_h^+ u^n)^T\cdot D\cdot\nabla_h^+ u^n. \tag{4.26}$$

With the aid of *Lemma 2.2*, we obtain the time derivative of the discrete energy density

$$\delta_t^+ E^n = \epsilon^2 A_t U^n\cdot\delta_t^+ U^n + \frac{1}{2}A_t V^n\cdot\delta_t^+ V^n + \frac{A_t u^n}{\tau_v}\delta_t^+ u^n + (\delta_t^+\nabla_h^+ u^n)^T\cdot D\cdot\nabla_h^+ A_t u^n. \tag{4.27}$$

Applying the linear-implicit Crank-Nicolson method in time, the Euler method in space to system (4.23) - (4.24), we obtain the first LEDRP algorithm.

Graphene Algorithm 1. Given the physical variables at time t_{n-1} and t_n, we compute their values at t_{n+1} using the following formula.

$$\begin{cases} \begin{pmatrix} \frac{1}{2}A_t V^n\cdot P^{n,\star} \\ \frac{A_t u^n}{\tau_v} \end{pmatrix} = A_t\boldsymbol{k}^{\boldsymbol{n}}, \quad \begin{pmatrix} \epsilon^2 A_t U^n\cdot\boldsymbol{R}^{n,\star} \\ D\cdot\nabla_h^+ A_t u^n \end{pmatrix} = A_t\boldsymbol{H}^{\boldsymbol{n}}, \\ \delta_t^+\mathbf{u}^n = -\boldsymbol{M}\left(\overline{\Theta}^{n,\star}\right)\cdot A_t\boldsymbol{\mu}^{\boldsymbol{n}} + \mathbf{g}(\mathbf{u}^{n,\star}), \\ A_t\boldsymbol{\mu}^n = A_t\boldsymbol{k}^n - \nabla_h^-\cdot A_t\boldsymbol{H}^n, \\ \delta_t^+ U^n = \boldsymbol{R}^{n,\star}\cdot\delta_t^+\nabla_h^+\phi^n, \\ \delta_t^+ V^n = P^{n,\star}\cdot\delta_t^+\phi^n, \end{cases} \tag{4.28}$$

with

$$\begin{cases} \boldsymbol{R}^{n,\star} = \dfrac{1}{\sqrt{|\xi_{s,n}(\Theta^{n,\star})\nabla_h^+\phi^{n,\star}|^2 + A}} \left\{ \xi_{s,n}(\Theta^{n,\star})^2 \nabla_h^+ \phi^{n,\star} + \begin{pmatrix} -\xi_{s,n}(\Theta^{n,\star})\xi'_{s,n}(\Theta^{n,\star})\delta_y^+\phi^{n,\star} \\ \xi_{s,n}(\Theta^{n,\star})\xi'_{s,n}(\Theta^{n,\star})\delta_x^+\phi^{n,\star} \end{pmatrix} \right\}, \\ P^{n,\star} = 1 - 2\phi^{n,\star}, \\ \mathbf{g}(\mathbf{u}^{n,\star}) = \boldsymbol{M}\left(\overline{\Theta}^{n,\star}\right) \cdot \begin{pmatrix} \epsilon\lambda\frac{1}{4}A_t V^n V^{n,\star} u^{n,\star} \\ f \end{pmatrix}, \end{cases} \tag{4.29}$$

and $\overline{\Theta}^{n,\star} = \arctan\frac{\delta_x^c\phi^{n,\star}}{\delta_y^c\phi^{n,\star}}$, $\Theta^{n,\star} = \arctan\frac{\delta_x^+\phi^{n,\star}}{\delta_y^+\phi^{n,\star}}$, $\delta_x^c\phi_{j,k} = \frac{\phi_{j+1,k}-\phi_{j-1,k}}{2h_x}$, $\delta_y^c\phi_{j,k} = \frac{\phi_{j,k+1}-\phi_{j,k-1}}{2h_y}$.

These can be specifically written into

$$\begin{cases} \phi^{n+1} = \phi^n + dt * \dfrac{1}{\alpha\tau(\overline{\Theta}^{n,\star})\epsilon^2}\left[-\frac{1}{2}A_t V^n \cdot P^{n,\star} + \nabla_h^- \cdot \left(\epsilon^2 A_t U^n \cdot \boldsymbol{R}^{n,\star}\right) + \epsilon\lambda \cdot \frac{1}{4}A_t V^n \cdot V^{n,\star} u^{n,\star}\right], \\ u^{n+1} = \left(\left(\dfrac{1}{dt} - \dfrac{1}{2\tau_v}\right)u^n - \dfrac{\phi^{n+1}-\phi^n}{dt} + \nabla_h^-\cdot\left(D\cdot\nabla_h^+ A_t u^n\right) + f\right) / \left(\dfrac{1}{dt} + \dfrac{1}{2\tau_v}\right), \\ U^{n+1} = U^n + \boldsymbol{R}^{n,\star}\cdot\nabla_h^+\left(\phi^{n+1}-\phi^n\right), \\ V^{n+1} = V^n + P^{n,\star}\cdot\left(\phi^{n+1}-\phi^n\right), \\ \boldsymbol{R}^{n,\star} = \dfrac{1}{\sqrt{|\xi_{s,n}(\Theta^{n,\star})\nabla_h^+\phi^{n,\star}|^2 + A}} \left\{ \xi_{s,n}(\Theta^{n,\star})^2 \nabla_h^+ \phi^{n,\star} + \begin{pmatrix} -\xi_{s,n}(\Theta^{n,\star})\xi'_{s,n}(\Theta^{n,\star})\delta_y^+\phi^{n,\star} \\ \xi_{s,n}(\Theta^{n,\star})\xi'_{s,n}(\Theta^{n,\star})\delta_x^+\phi^{n,\star} \end{pmatrix} \right\}, \\ P^{n,\star} = 1 - 2\phi^{n,\star}. \end{cases} \tag{4.30}$$

Graphene Algorithm 1 admits a discrete energy density transport equation derivable by *Lemma 2.3*.

Theorem 4.3. System (4.28)-(4.29) admits the following discrete energy density transport equation consistent with the continuous one

$$\delta_t^+ E^n - \nabla_h^+ \cdot \left(A_t\overline{\boldsymbol{H}}^n \cdot \delta_t^+ \mathbf{u}^n\right) + \left(A_t\boldsymbol{\mu}^n\right)^T \cdot \boldsymbol{M}(\overline{\Theta}^{n,\star}) \cdot A_t\boldsymbol{\mu}^n = \mathbf{g}\left(\mathbf{u}^{n,\star}\right) \cdot A_t\boldsymbol{\mu}^n, \tag{4.31}$$

with E the energy density defined by (4.26).

4.2.2. LEDRP algorithm II

We discretize energy density (4.18) as follows

$$E(U^n, V^n, u^n) = \frac{1}{2}\epsilon^2|U^n|^2 - \frac{1}{2}\epsilon^2 A + \frac{1}{4}|V^n|^2 + \frac{|u^n|^2}{2\tau_v} + \frac{1}{2}(\nabla_h^- u^n)^T \cdot D \cdot \nabla_h^- u^n. \tag{4.32}$$

With the aid of *Lemma 2.2*, we have

$$\delta_t^+ E^n = \epsilon^2 A_t U^n \cdot \delta_t^+ U^n + \frac{1}{2}A_t V^n \cdot \delta_t^+ V^n + \frac{A_t u^n}{\tau_v}\delta_t^+ u^n + (\delta_t^+\nabla_h^- u^n)^T \cdot D \cdot \nabla_h^- A_t u^n. \tag{4.33}$$

Graphene Algorithm 2. Given physical variables at time t_{n-1} and t_n, we compute their values at t_{n+1} using the following.

$$\begin{cases} \begin{pmatrix} \frac{1}{2}A_tV^n\cdot P^{n,\star} \\ \frac{A_tu^n}{\tau_v} \end{pmatrix} = A_t\boldsymbol{k^n}, \quad \begin{pmatrix} \epsilon^2 A_tU^n\cdot \boldsymbol{R}^{n,\star} \\ D\cdot\nabla_h^- A_tu^n \end{pmatrix} = A_t\boldsymbol{H^n}, \\ \delta_t^+\mathbf{u}^n = -\boldsymbol{M}\left(\overline{\Theta}^{n,\star}\right)\cdot A_t\boldsymbol{\mu^n} + \mathbf{g}(\mathbf{u}^{n,\star}), \\ A_t\boldsymbol{\mu}^n = A_t\boldsymbol{k}^n - \nabla_h^+\cdot A_t\boldsymbol{H}^n, \\ \delta_t^+U^n = \boldsymbol{R}^{n,\star}\cdot\delta_t^+\nabla_h^-\phi^n, \\ \delta_t^+V^n = P^{n,\star}\cdot\delta_t^+\phi^n, \end{cases} \tag{4.34}$$

with

$$\begin{cases} \boldsymbol{R}^{n,\star} = \dfrac{1}{\sqrt{|\xi_{s,n}(\Theta^{n,*})\nabla_h^-\phi^{n,\star}|^2 + A}}\left\{\xi_{s,n}(\Theta^{n,\star})^2\nabla_h^-\phi^{n,\star} + \begin{pmatrix} -\xi_{s,n}(\Theta^{n,\star})\xi'_{s,n}(\Theta^{n,\star})\delta_y^-\phi^{n,\star} \\ \xi_{s,n}(\Theta^{n,\star})\xi'_{s,n}(\Theta^{n,\star})\delta_x^-\phi^{n,\star} \end{pmatrix}\right\}, \\ P^{n,\star} = 1-2\phi^{n,\star}, \\ \mathbf{g}(\mathbf{u}^{n,\star}) = \boldsymbol{M}\left(\overline{\Theta}^{n,\star}\right)\cdot\begin{pmatrix} \epsilon\lambda\frac{1}{4}A_tV^nV^{n,\star}u^{n,\star} \\ f \end{pmatrix}, \end{cases} \tag{4.35}$$

and $\overline{\Theta}^{n,\star} = \arctan\frac{\delta_x^c\phi^{n,\star}}{\delta_y^c\phi^{n,\star}}$, $\Theta^{n,\star} = \arctan\frac{\delta_x^-\phi^{n,\star}}{\delta_y^-\phi^{n,\star}}$.

Similarly, system (4.34)-(4.35) admits a discrete energy density transport equation derived using *Lemma 2.4*.

Theorem 4.4. System (4.34)-(4.35) admits the following discrete energy transport equation

$$\delta_t^+E^n - \nabla_h^-\cdot\left(A_t\widetilde{\boldsymbol{H}}^n\cdot\delta_t^+\mathbf{u}^n\right) + (A_t\boldsymbol{\mu}^n)^T\cdot\boldsymbol{M}(\overline{\Theta}^{n,\star})\cdot A_t\boldsymbol{\mu}^n = \mathbf{g}\left(\mathbf{u}^{n,\star}\right)\cdot A_t\boldsymbol{\mu}^n, \tag{4.36}$$

with E the energy density defined by (4.32).

4.2.3. LEDRP algorithm III

We discretize energy density (4.18) as follows

$$E(U^n,V^n,u^n) = \frac{1}{2}\epsilon^2|A_xA_yU^n|^2 - \frac{1}{2}\epsilon^2A + \frac{1}{4}|A_xA_yV^n|^2 + \frac{|A_xA_yu^n|^2}{2\tau_v} + \frac{1}{2}(\overline{\nabla}_hu^n)^T\cdot D\cdot\overline{\nabla}_hu^n. \tag{4.37}$$

We obtain the time derivative of the discrete energy density as follows

$$\begin{aligned} \delta_t^+E^n = {}& \epsilon^2A_tA_xA_yU^n\cdot\delta_t^+A_xA_yU^n + \frac{1}{2}A_tA_xA_yV^n\cdot\delta_t^+A_xA_yV^n \\ & + \frac{A_tA_xA_yu^n}{\tau_v}\delta_t^+A_xA_yu^n + (\delta_t^+\overline{\nabla}_hu^n)^T\cdot D\cdot\overline{\nabla}_hA_tu^n. \end{aligned} \tag{4.38}$$

Graphene Algorithm 3. We present the third LEDRP algorithm as follows

$$\begin{cases} \begin{pmatrix} \frac{1}{2}A_tA_xA_yV^n \cdot A_xA_yP^{n,\star} \\ \frac{A_tA_xA_yu^n}{\tau_v} \end{pmatrix} = A_tA_xA_y\boldsymbol{k^n}, \quad \begin{pmatrix} \epsilon^2 A_tA_xA_yU^n \cdot A_xA_y\boldsymbol{R}^{n,\star} \\ D \cdot \overline{\nabla}_h A_t u^n \end{pmatrix} = A_tA_xA_y\boldsymbol{H^n}, \\ \delta_t^+ A_xA_y\mathbf{u}^n = -\boldsymbol{M}\left(\overline{\Theta}^{n,\star}\right) \cdot A_tA_xA_y\boldsymbol{\mu^n} + \mathbf{g}(A_xA_y\mathbf{u}^{n,\star}), \\ A_tA_xA_y\boldsymbol{\mu}^n = A_tA_xA_y\boldsymbol{k}^n - \overline{\nabla}_h \cdot A_t\boldsymbol{H}^n, \\ \delta_t^+ A_xA_yU^n = A_xA_y\boldsymbol{R}^{n,\star} \cdot \delta_t^+\overline{\nabla}_h\phi^n, \\ \delta_t^+ A_xA_yV^n = A_xA_yP^{n,\star} \cdot \delta_t^+ A_xA_y\phi^n, \end{cases} \tag{4.39}$$

with

$$\begin{cases} A_xA_y\boldsymbol{R}^{n,\star} = \dfrac{1}{\sqrt{|\xi_{s,n}(\Theta^{n,\star})\overline{\nabla}_h\phi^{n,\star}|^2 + A}} \left\{ \xi_{s,n}(\Theta^{n,\star})^2\overline{\nabla}_h\phi^{n,\star} + \begin{pmatrix} -\xi_{s,n}(\Theta^{n,\star})\xi'_{s,n}(\Theta^{n,\star})\delta_y^+ A_x\phi^{n,\star} \\ \xi_{s,n}(\Theta^{n,\star})\xi'_{s,n}(\Theta^{n,\star})\delta_x^+ A_y\phi^{n,\star} \end{pmatrix} \right\}, \\ A_xA_yP^{n,\star} = 1 - 2A_xA_y\phi^{n,\star}, \\ \mathbf{g}(\mathbf{u}^{n,\star}) = \boldsymbol{M}\left(\overline{\Theta}^{n,\star}\right) \cdot \begin{pmatrix} \epsilon\lambda\frac{1}{4}A_tA_xA_yV^nA_xA_yV^{n,\star}A_xA_yu^{n,\star} \\ f \end{pmatrix}, \end{cases} \tag{4.40}$$

and $\overline{\Theta}^{n,\star} = \arctan\frac{\delta_x^c A_y\phi^{n,\star}}{\delta_y^c A_x\phi^{n,\star}}$, $\Theta^{n,\star} = \arctan\frac{\delta_x^+ A_y\phi^{n,\star}}{\delta_y^+ A_x\phi^{n,\star}}$.

To derive the energy density transport equation for system (4.39)-(4.40), we need the following lemma.

Lemma 4.1. For scalar function f and vector $\mathbf{v} = (v_1, v_2)^T$, we define $\nabla_h^{[1]}\cdot$ as follows

$$\nabla_h^{[1]} \cdot (f^n\mathbf{v}^n) = \delta_x^+ \left(A_y f^n \cdot A_y v_1^n\right) + \delta_y^+ \left(A_x f^n \cdot A_x v_2^n\right). \tag{4.41}$$

Based on *Lemma 2.2*, the operator has the following property

$$\nabla_h^{[1]} \cdot (f^n\mathbf{v}^n) = \overline{\nabla}_h f^n \cdot A_xA_y\mathbf{v}^n + A_xA_yf^n \cdot (\overline{\nabla}_h \cdot \mathbf{v}^n). \tag{4.42}$$

Theorem 4.5. System (4.39) - (4.40) admits the following discrete energy density transport equation

$$\delta_t^+ E^n - \nabla_h^{[1]} \cdot \left(A_t\boldsymbol{H}^n \cdot \delta_t^+\mathbf{u}^n\right) + \left(A_tA_xA_y\boldsymbol{\mu}^n\right)^T \cdot \boldsymbol{M}(\overline{\Theta}^{n,\star}) \cdot A_tA_xA_y\boldsymbol{\mu}^n = \mathbf{g}(A_xA_y\mathbf{u}^{n,\star}) \cdot A_tA_xA_y\boldsymbol{\mu}^n, \tag{4.43}$$

with E^n the energy density defined in (4.37).

With these new algorithms for the graphene growth model, we will verify their rates of convergence numerically next.

5. Numerical results

We first conduct mesh refinement tests on the first two algorithms subject to periodic boundary conditions. The implementation of the third algorithm is too complicated to be included in this paper. Then, we simulate graphene growth numerically using the driven phase field model.

5.1. Mesh refinement tests

We use $[0, 2\pi] \times [0, 2\pi]$ as the spatial domain in the mesh refinement tests and choose the initial conditions as follows

$$\phi(\mathbf{x}, 0) = 0.1 \sin(x) \sin(y), \quad u(\mathbf{x}, 0) = 0.1 \cos(x) \cos(y). \tag{5.1}$$

We use $1e-4 * \mathcal{M}$ as the mobility value here and the other parameter values used are listed in Table 1.

In the test, we take a linear refinement path $\tau = \frac{0.1}{2^k}$, $k = 1...6$, and $N_x = N_y = 16 * 2^k$, $k = 1, 2, ..., 6$. Then we calculate the errors and orders in the L^2 norm, respectively. Table 2 - Table 3 demonstrate the second-order convergence of *Graphene Algorithm 1* for ϕ and u, respectively, while Table 4 - Table 5 confirm the second-order convergence of *Graphene Algorithm 2* for ϕ and u, respectively.

Table 1: Parameter values used in mesh refinement tests

parameter	symbol	value
capillary parameter	d_0	6e-4
interfacial width	ϵ	1
desorption time	τ_v	1
step energy anisotropy(n-fold)	$\epsilon_{s,n}$	0.001
kinetic anisotropy(n-fold)	$\epsilon_{k,n}$	0.08
crystal symmetry of copper	n	6
diffusion anisotropy	δ	0
rotated angle of principle diffusion axes	Ψ	0
deposition flux	f	2.708
kinetic coefficient	β	5.54e-3

5.2. Graphene sheet growth

In experiments and reported numerical simulations of graphene growth, we observe that growth patterns can be quite complex. This can be contributed to many factors in the experiments. In our model, we will select proper parameter values to simulate the growth and benchmark with the available results. In our simulations, we use the codes implemented using the two algorithms to simulate benchmark examples in a periodic domain $[0, 2\pi]^2$ with 128^2 meshes and $\tau = 1.0e-4$. We fix the initial conditions as follows

$$\phi(\mathbf{x}, 0) = \frac{1}{2}\left(1 + \tanh\frac{0.3 - r}{0.01}\right), \quad u(\mathbf{x}, 0) = 1, \tag{5.2}$$

Table 2: Mesh refinement tests for ϕ for *Graphene Algorithm 1* at $T = 0.1$, $N_x = N_y = N$.

N		τ		L^2 Error	L^2 Order
coarse N	fine N	coarse τ	fine τ		
$16 * 2^1$	$16 * 2^2$	$1e-1/2^1$	$1e-1/2^2$	3.0006e-09	-
$16 * 2^2$	$16 * 2^3$	$1e-1/2^2$	$1e-1/2^3$	8.1434e-10	1.8815
$16 * 2^3$	$16 * 2^4$	$1e-1/2^3$	$1e-1/2^4$	2.0885e-10	1.9631
$16 * 2^4$	$16 * 2^5$	$1e-1/2^4$	$1e-1/2^5$	5.3577e-11	1.9627
$16 * 2^5$	$16 * 2^6$	$1e-1/2^5$	$1e-1/2^6$	1.3937e-11	1.9426

Table 3: Mesh refinement tests for u for *Graphene Algorithm 1* at $T = 0.1$, $N_x = N_y = N$.

N		τ		L^2 Error	L^2 Order
coarse N	fine N	coarse τ	fine τ		
$16 * 2^1$	$16 * 2^2$	$1e-1/2^1$	$1e-1/2^2$	1.2089e-07	-
$16 * 2^2$	$16 * 2^3$	$1e-1/2^2$	$1e-1/2^3$	3.0553e-08	1.9843
$16 * 2^3$	$16 * 2^4$	$1e-1/2^3$	$1e-1/2^4$	7.6590e-09	1.9961
$16 * 2^4$	$16 * 2^5$	$1e-1/2^4$	$1e-1/2^5$	1.9160e-09	1.9991
$16 * 2^5$	$16 * 2^6$	$1e-1/2^5$	$1e-1/2^6$	4.7894e-10	2.0001

Table 4: Mesh refinement tests for ϕ for *Graphene Algorithm 2* at $T = 0.1$, $N_x = N_y = N$.

N		τ		L^2 Error	L^2 Order
coarse N	fine N	coarse τ	fine τ		
$16 * 2^1$	$16 * 2^2$	$1e-1/2^1$	$1e-1/2^2$	3.0006e-09	-
$16 * 2^2$	$16 * 2^3$	$1e-1/2^2$	$1e-1/2^3$	8.1434e-10	1.8815
$16 * 2^3$	$16 * 2^4$	$1e-1/2^3$	$1e-1/2^4$	2.0885e-10	1.9631
$16 * 2^4$	$16 * 2^5$	$1e-1/2^4$	$1e-1/2^5$	5.3577e-11	1.9627
$16 * 2^5$	$16 * 2^6$	$1e-1/2^5$	$1e-1/2^6$	1.3937e-11	1.9426

where $r = \sqrt{(x-\pi)^2 + (y-\pi)^2}$.

In Figure 1 and Figure 2, we show the growth patterns simulated using *Graphene Algorithm 1* and *Graphene Algorithm 2*, respectively. Among these simulations, the results in the corresponding rows of Figure 1 and Figure 2 use the same parameter values. Clearly, we observe that the patterns computed using the two algorithms are visually identical while using the same set of parameter values. The graphene pattern grows from a small circular seed with radius $r = 0.3$ and a given diffusive diatom field, to a shaped pattern gradually. From our numerical experiments, the patterns are clear when $T = 0.1$, therefore it grows rather fast under certain parameter values. The patterns are consistent with the results for Cu(100), Cu(221), Cu(110) and Cu(310) reported in [21].

Table 5: Mesh refinement tests for u for *Graphene Algorithm 2* at $T = 0.1$, $N_x = N_y = N$.

N		τ		L^2 Error	L^2 Order
coarse N	fine N	coarse τ	fine τ		
$16 * 2^1$	$16 * 2^2$	$1e-1/2^1$	$1e-1/2^2$	1.2089e-07	-
$16 * 2^2$	$16 * 2^3$	$1e-1/2^2$	$1e-1/2^3$	3.0553e-08	1.9843
$16 * 2^3$	$16 * 2^4$	$1e-1/2^3$	$1e-1/2^4$	7.6590e-09	1.9961
$16 * 2^4$	$16 * 2^5$	$1e-1/2^4$	$1e-1/2^5$	1.9160e-09	1.9991
$16 * 2^5$	$16 * 2^6$	$1e-1/2^5$	$1e-1/2^6$	4.7894e-10	2.0001

In order to numerically verify the accuracy of the energy density transport equation, we define the maximal residue of the equation the respective algorithms as follows

$$LE_I^{n+1/2} = \max_{j,k} \left| \delta_t^+ E^n - \nabla_h^+ \cdot \left(A_t \overline{\boldsymbol{H}}^n \cdot \delta_t^+ \mathbf{u}^n \right) + (A_t \boldsymbol{\mu}^n)^T \cdot \boldsymbol{M}(\overline{\Theta}^{n,*}) \cdot A_t \boldsymbol{\mu}^n - \mathbf{g}\left(\mathbf{u}^{n,*}\right) \cdot A_t \boldsymbol{\mu}^n \right|, \tag{5.3}$$

with the variables defined in *Graphene Algorithm 1*, and

$$LE_{II}^{n+1/2} = \max_{j,k} \left| \delta_t^+ E^n - \nabla_h^- \cdot \left(A_t \widetilde{\boldsymbol{H}}^n \cdot \delta_t^+ \mathbf{u}^n \right) + (A_t \boldsymbol{\mu}^n)^T \cdot \boldsymbol{M}(\overline{\Theta}^{n,*}) \cdot A_t \boldsymbol{\mu}^n - \mathbf{g}\left(\mathbf{u}^{n,*}\right) \cdot A_t \boldsymbol{\mu}^n \right|, \tag{5.4}$$

with the variables defined in *Graphene Algorithm 2*.

In Figure 3, we depict the maximal residue in $t \in [0, 0.1]$ for *Graphene Algorithm 1* and *Graphene Algorithm 2*, respectively. We observe that both algorithms keep the maximum residue well under 10^{-9}, which confirms our theoretical energy dissipation rate preserving results.

6. Concluding remarks

We have developed a general framework for developing LEDRP algorithms for gradient flows, a general thermodynamically consistent model, and gradient flows driven by source terms. By employing the EQ technique, we can develop linear LEDRP algorithms for the gradient flow systems, including the driven system. For the driven gradient flow, LEDRP is important since it ensures the discrete energy density transport equation respects the property and structure of its continuous counterpart. In proper boundary conditions, for instance, periodic boundary conditions, the local structure-preserving property implies the global energy dissipation property. We then apply the general approach to construct three second-order, LEDRP algorithms for the graphene growth model. Numerical experiments confirm the second order accuracy of the algorithms and demonstrate good match with the benchmarking examples when simulating graphene growth in different parameter regimes.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (Grant No.11771213, 61872422), the National Key Research and Development Project of China (Grant No.2018YFC1504205),

the Major Projects of Natural Sciences of University in Jiangsu Province of China (Grant No.18KJA110003). Qi Wang's work is partially supported by National Science Foundation (award DMS-1815921, DMS-1954532 and OIA-1655740), DOE DE-SC0020272 award and a GEAR award from SC EPSCoR/IDeA Program.

Appendix A: Proof of some theorems in §2

a. Proof for Theorem 2.1: Multiplying μ and ϕ_t on both sides of the first equation and the second equation of (2.8), respectively, we obtain

$$\begin{cases} \phi_t \mu = -\mu \mathcal{M} \mu, \\ \phi_t \mu = \phi_t \displaystyle\sum_{i=0}^{m} (-1)^i \nabla^i \frac{\partial f}{\partial \nabla^i \phi}. \end{cases} \tag{6.1}$$

Combining the two equations in (6.1), we have

$$-\mu \mathcal{M} \mu = \phi_t \sum_{i=0}^{m} (-1)^i \nabla^i \frac{\partial f}{\partial \nabla^i \phi}. \tag{6.2}$$

Applying *Lemma 2.1* and substituting all the terms on the right hand of (6.2), we arrive at

$$-\mu \mathcal{M} \mu = \sum_{i=0}^{m} (-1)^i \nabla \cdot \left((-1)^{k-1} \sum_{k=1}^{i} \nabla^{k-1} \phi_t \cdot \nabla^{i-k} \frac{\partial f}{\partial \nabla^i \phi} \right) + \sum_{i=0}^{m} \nabla^i \phi_t \cdot \frac{\partial f}{\partial \nabla^i \phi}. \tag{6.3}$$

Meanwhile, the time derivative of energy density (2.2) is given by

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \sum_{i=0}^{m} \nabla^i \phi_t \cdot \frac{\partial f}{\partial \nabla^i \phi}. \tag{6.4}$$

Inserting (6.4) into (6.3), we complete the proof.

b. Proof for Theorem 2.2: Multiplying μ and ϕ_t on both sides of the first equation and the second equation in (2.15), respectively, we have

$$\begin{cases} \phi_t \mu = -\mu \mathcal{M} \mu, \\ \mu \phi_t = \left(\displaystyle\sum_{p=1}^{\frac{m+1}{2}} \Delta^{p-1} k_p - \sum_{q=1}^{\frac{m+1}{2}} \nabla \Delta^{q-1} \mathbf{h}_q \right) \cdot \phi_t. \end{cases} \tag{6.5}$$

Using the Leibnitz rule repeatedly, we obtain

$$\begin{aligned} &\nabla \cdot \left[\sum_{p=2}^{\frac{m+1}{2}} \sum_{l=0}^{2p-3} \left((-1)^{l+1} \nabla^l k_p \cdot \nabla^{2p-3-l} \phi_t \right) + \sum_{q=1}^{\frac{m+1}{2}} \sum_{r=0}^{2q-2} \left((-1)^{r+1} \nabla^r \mathbf{h}_q \cdot \nabla^{2q-2-r} \phi_t \right) \right] \\ &+ \sum_{p=1}^{\frac{m+1}{2}} \left(k_p \cdot \Delta^{p-1} \phi_t \right) + \sum_{q=1}^{\frac{m+1}{2}} \left(\mathbf{h}_q \cdot \nabla \Delta^{q-1} \phi_t \right) = -\mu \mathcal{M} \mu. \end{aligned} \tag{6.6}$$

It follows from the definition of the energy density (2.2)

$$\partial_t E = \sum_{p=1}^{\frac{m+1}{2}} \left(k_p \cdot \Delta^{p-1}\phi_t\right) + \sum_{q=1}^{\frac{m+1}{2}} \left(\mathbf{h}_q \cdot \nabla\Delta^{q-1}\phi_t\right), \tag{6.7}$$

substituting (6.7) into (6.6), we complete the proof.

c. Proof for Theorem 2.3: Using *Lemma 2.3* m times, we have

$$\begin{aligned}
-A_t\mu^n\mathcal{M}A_t\mu^n =& \nabla_h^+ \cdot \left(\sum_{p=2}^{\frac{m+1}{2}} \sum_{l=0}^{p-2} \left(\delta_t^+\Delta_h^l\phi^n \cdot \nabla_h^-\Delta_h^{p-2-l}A_t k_p^n - \delta_t^+\nabla_h^-\Delta_h^l\phi^n \cdot \Delta_h^{p-2-l}A_t k_p^n \right) \right. \\
& - \sum_{q=1}^{\frac{m+1}{2}} \delta_t^+\Delta_h^{q-1}\phi^n \cdot A_t\overline{\mathbf{h}}_q^n \\
& \left. + \sum_{q=2}^{\frac{m+1}{2}} \sum_{r=0}^{q-2} \left(-\delta_t^+\Delta_h^r\phi^n \cdot \nabla_h^-\nabla_h^-\Delta_h^{q-2-r}A_t\mathbf{h}_q^n + \delta_t^+\nabla_h^-\Delta_h^r\phi^n \cdot \nabla_h^-\Delta_h^{q-2-r}A_t\mathbf{h}_q^n \right) \right) \\
& + \sum_{p=1}^{\frac{m+1}{2}} A_t k_p^n \cdot \delta_t^+\Delta_h^{p-1}\phi^n + \sum_{q=1}^{\frac{m+1}{2}} A_t\mathbf{h}_q^n \cdot \delta_t^+\nabla_h^+\Delta_h^{q-1}\phi^n.
\end{aligned} \tag{6.8}$$

It follows from the definition of the energy density, E^n, (2.18),

$$\delta_t^+ E^n = \sum_{p=1}^{\frac{m+1}{2}} A_t k_p^n \cdot \delta_t^+\Delta_h^{p-1}\phi^n + \sum_{q=1}^{\frac{m+1}{2}} A_t\mathbf{h}_q^n \cdot \delta_t^+\nabla_h^+\Delta_h^{q-1}\phi^n. \tag{6.9}$$

Applying (6.9), we complete the proof.

d. Proof for Theorem 2.4: With the aid of *Lemma 2.4*, we arrive at the conclusion readily.

e. Proof for Theorem 2.5: It follows from the definition of energy density, E^n, (2.34)

$$\begin{aligned}
\delta_t^+ E^n =& \sum_{p=1}^{\frac{m+1}{2}} A_t A_x^m A_y^m k_p^n \cdot \delta_t^+\overline{\Delta}_h^{p-1} A_x^{m-2p+2} A_y^{m-2p+2}\phi^n \\
& + \sum_{q=1}^{\frac{m+1}{2}} A_t A_x^m A_y^m \mathbf{h}_q^n \cdot \delta_t^+\overline{\nabla}_h\overline{\Delta}_h^{q-1} A_x^{m-2q+1} A_y^{m-2q+1}\phi^n.
\end{aligned} \tag{6.10}$$

With the aid of *Lemma 2.5*, we complete the proof.

f. Proof for Theorem 2.6: Multiply μ, ϕ_t, q on both sides of the three equations in (2.46),

respectively,

$$\begin{cases} \mu\phi_t = -\mu\mathcal{M}\mu, \\ \mu\phi_t = 2\sum_{i=0}^{m}(-1)^i g_i(\epsilon)\cdot\nabla^{2i}\phi\cdot\phi_t + 2\sum_{i=0}^{m-1}(-1)^i\nabla^i\left(q\cdot\frac{\partial q}{\partial\nabla^i\phi}\right)\cdot\phi_t, \\ qq_t = \sum_{i=0}^{m-1} q\frac{\partial q}{\partial\nabla^i\phi}\cdot\frac{\partial\nabla^i\phi}{\partial t}. \end{cases} \tag{6.11}$$

With *Lemma 2.6*, we derive from (6.11)

$$\begin{aligned} -\mu\mathcal{M}\mu = & 2\sum_{i=0}^{m}\left[(-1)^i g_i(\epsilon)\nabla\left(\sum_{k=1}^{i}(-1)^{k-1}\nabla^{k-1}\phi_t\cdot\nabla^{2i-k}\phi\right)\right] \\ & + 2\sum_{i=0}^{m-1}\left[(-1)^i\nabla\left((-1)^{k-1}\sum_{k=1}^{i}\nabla^{k-1}\phi_t\cdot\nabla^{i-k}\left(q\frac{\partial q}{\partial\nabla^i\phi}\right)\right)\right] \\ & + 2\sum_{i=0}^{m} g_i(\epsilon)\nabla^i\phi_t\cdot\nabla^i\phi_t + 2qq_t, \end{aligned} \tag{6.12}$$

Meanwhile, the time derivative of energy density E, (2.45), is given by

$$\frac{\mathrm{d}E}{\mathrm{d}t} = 2\sum_{i=0}^{m} g_i(\epsilon)\nabla^i\phi_t\cdot\nabla^i\phi_t + 2qq_t. \tag{6.13}$$

Inserting (6.13) into (6.12), we arrive at the conclusion.

Appendix B: LEDRP algorithms in §2.2.3

Here we list the three LEDRP algorithms based on the energy quadratization technique presented in §2.2.3. We present the first algorithm based on the energy quadratization method next. Eliminating the intermediate variables in system (2.54)-(2.56), we arrive at the first LEDRP algorithm based on the EQ method as follows.

Algorithm 4 (EQ-LEDRP-I)**.**

$$\begin{cases} \delta_t^+\phi^n = -2\mathcal{M}\left[\sum_{i=0}^{m}(-1)^i g_i(\epsilon)\cdot A_t\Delta_h^i\phi^n + \sum_{p=1}^{\frac{m+1}{2}}\Delta_h^{p-1}\left(A_t q^n\frac{\partial q^{n,\star}}{\partial\Delta_h^{p-1}\phi^{n,\star}}\right)\right. \\ \qquad \left. - \sum_{q=1}^{\frac{m-1}{2}}\nabla_h^-\Delta_h^{q-1}\left(A_t q^n\frac{\partial q^{n,\star}}{\partial\nabla_h^+\Delta_h^{q-1}\phi^{n,\star}}\right)\right], \\ \delta_t^+ q^n = \sum_{p=1}^{\frac{m+1}{2}}\frac{\partial q^{n,\star}}{\partial\Delta_h^{p-1}\phi^{n,\star}}\cdot\delta_t^+\Delta_h^{p-1}\phi^n + \sum_{q=1}^{\frac{m-1}{2}}\frac{\partial q^{n,\star}}{\partial\nabla_h^+\Delta_h^{q-1}\phi^{n,\star}}\cdot\delta_t^+\nabla_h^+\Delta_h^{q-1}\phi^n. \end{cases}$$

The transport equation for the discrete energy density in *Algorithm 4* is given below.

Theorem 6.1. Model (2.54)-(2.56) satisfies the following discrete LEDL

$$
\begin{aligned}
&\delta_t^+ E^n + 2\nabla_h^+ \cdot \left(\sum_{p=2}^{\frac{m+1}{2}} \sum_{l=0}^{p-2} \left(\delta_t^+ \Delta_h^l \phi^n \cdot \nabla_h^- \Delta_h^{p-2-l} A_t k_p^n + \delta_t^+ \nabla_h^- \Delta_h^l \phi^n \cdot \Delta_h^{p-2-l} A_t k_p^n \right) - \delta_t^+ \phi^n \cdot A_t \overline{\mathbf{h}}_1^n \right. \\
&+ \sum_{q=2}^{\frac{m-1}{2}} \left(\sum_{r=0}^{q-2} \left(-\delta_t^+ \Delta_h^r \phi^n \cdot \nabla_h^- \nabla_h^- \Delta_h^{q-2-r} A_t \mathbf{h}_q^n + \delta_t^+ \nabla_h^- \Delta_h^r \phi^n \cdot \nabla_h^- \Delta_h^{q-2-r} A_t \mathbf{h}_q^n \right) - \delta_t^+ \Delta_h^{q-1} \phi^n \cdot A_t \overline{\mathbf{h}}_q^n \right) \\
&+ \sum_{i=1}^{m} g_i(\epsilon) \left(- \sum_{l=\frac{i-1}{2}}^{i-1} \nabla_h^- \Delta_h^l A_t \phi^n \cdot \delta_t^+ \Delta_h^{i-1-l} \phi^n + \sum_{l=\frac{i+1}{2}}^{i-1} \Delta_h^l A_t \phi^n \cdot \delta_t^+ \nabla_h^- \Delta_h^{i-1-l} \phi^n \right) \\
&\left. + \sum_{i=1}^{m} g_i(\epsilon) \sum_{l=\frac{i}{2}}^{i-1} \left(\nabla_h^- \Delta_h^l A_t \phi^n \cdot \delta_t^+ \Delta_h^{i-1-l} \phi^n - \Delta_h^l A_t \phi^n \cdot \delta_t^+ \nabla_h^- \Delta_h^{i-1-l} \phi^n \right) \right) \\
&+ A_t \mu^n \mathcal{M} A_t \mu^n = 0,
\end{aligned}
$$

with discrete energy density E^n defined in (2.53).

Proof. We note that the time derivative of discrete energy density (2.53) is given by

$$
\begin{aligned}
\delta_t^+ E^n = & 2 \sum_{p=1}^{\frac{m+1}{2}} A_t k_p^n \cdot \delta_t^+ \Delta_h^{p-1} \phi^n + 2 \sum_{q=1}^{\frac{m-1}{2}} A_t \mathbf{h}_q^n \cdot \delta_t^+ \nabla_h^+ \Delta_h^{q-1} \phi^n \\
& + 2 \sum_{i=0}^{m} g_i(\epsilon) \Delta_h^{\frac{i}{2}} A_t \phi^n \cdot \Delta_h^{\frac{i}{2}} \delta_t^+ \phi^n + 2 \sum_{i=1}^{m} g_i(\epsilon) \nabla_h^+ \Delta_h^{\frac{i}{2}} A_t \phi^n \cdot \nabla_h^+ \Delta_h^{\frac{i-1}{2}} \delta_t^+ \phi^n.
\end{aligned} \tag{6.14}
$$

Analogous to the proof of Theorem 2.3, we can easily arrive at the conclusion. □

Then we present second LEDRP algorithm. Eliminating the intermediate variables, system (2.58)-(2.60) can be written into the following two-equation system.

Algorithm 5 (EQ-LEDRP-II)**.**

$$
\begin{cases}
\delta_t^+ \phi^n = -2\mathcal{M} \left[\sum_{i=0}^{m} (-1)^i g_i(\epsilon) \cdot A_t \Delta_h^i \phi^n + \sum_{p=1}^{\frac{m+1}{2}} \Delta_h^{p-1} \left(A_t q^n \dfrac{\partial q^{n,\star}}{\partial \Delta_h^{p-1} \phi^{n,\star}} \right) \right. \\
\qquad \left. - \sum_{q=1}^{\frac{m-1}{2}} \nabla_h^+ \Delta_h^{q-1} \left(A_t q^n \dfrac{\partial q^{n,\star}}{\partial \nabla_h^- \Delta_h^{q-1} \phi^{n,\star}} \right) \right], \\
\delta_t^+ q^n = \sum_{p=1}^{\frac{m+1}{2}} \dfrac{\partial q^{n,\star}}{\partial \Delta_h^{p-1} \phi^{n,\star}} \cdot \delta_t^+ \Delta_h^{p-1} \phi^n + \sum_{q=1}^{\frac{m-1}{2}} \dfrac{\partial q^{n,\star}}{\partial \nabla_h^- \Delta_h^{q-1} \phi^{n,\star}} \cdot \delta_t^+ \nabla_h^- \Delta_h^{q-1} \phi^n.
\end{cases}
$$

The discrete energy density in *Algorithm 5* obeys a transport equation given in the following theorem.

Theorem 6.2. Model (2.58)-(2.60) satisfies the following discrete local energy density transport equation

$$
\begin{aligned}
&\delta_t^+ E^n + 2\nabla_h^- \cdot \Bigg(\sum_{p=2}^{\frac{m+1}{2}} \sum_{l=0}^{p-2} \Big(\delta_t^+ \Delta_h^l \phi^n \cdot \nabla_h^+ \Delta_h^{p-2-l} A_t k_p^n - \delta_t^+ \nabla_h^+ \Delta_h^l \phi^n \cdot \Delta_h^{p-2-l} A_t k_p^n \Big) - \delta_t^+ \phi^n \cdot A_t \widetilde{\mathbf{h}}_1^n \\
&+ \sum_{q=2}^{\frac{m-1}{2}} \Bigg(\sum_{r=0}^{q-2} \Big(-\delta_t^+ \Delta_h^r \phi^n \cdot \Delta_h^{q-1-r} A_t \widetilde{\mathbf{h}}_q^n + \delta_t^+ \nabla_h^+ \Delta_h^r \phi^n \cdot \nabla_h^+ \Delta_h^{q-2-r} A_t \mathbf{h}_q^n \Big) - \delta_t^+ \Delta_h^{q-1} \nabla_h^- \phi^n \cdot A_t \widetilde{\mathbf{h}}_q^n \Bigg) \\
&+ \sum_{i=1}^{m} g_i(\epsilon) \Bigg(- \sum_{l=\frac{i-1}{2}}^{i-1} \nabla_h^+ \Delta_h^l A_t \phi^n \cdot \delta_t^+ \Delta_h^{i-1-l} \phi^n + \sum_{l=\frac{i+1}{2}}^{i-1} \Delta_h^l A_t \phi^n \cdot \delta_t^+ \nabla_h^+ \Delta_h^{i-1-l} \phi^n \Bigg) \\
&+ \sum_{i=1}^{m} g_i(\epsilon) \sum_{l=\frac{i}{2}}^{i-1} \left(\nabla_h^+ \Delta_h^l A_t \phi^n \cdot \delta_t^+ \Delta_h^{i-1-l} \phi^n - \Delta_h^l A_t \phi^n \cdot \delta_t^+ \nabla_h^+ \Delta_h^{i-1-l} \phi^n \right) \Bigg) \\
&+ A_t \mu^n \mathcal{M} A_t \mu^n = 0,
\end{aligned}
$$

with discrete energy density E^n defined in (2.57).

Analogously, we obtain the third LEDRP algorithm the same way.

Algorithm 6 (EQ-LEDRP-III)**.**

$$
\begin{cases}
\delta_t^+ A_x^{2m} A_y^{2m} \phi^n = 2 \sum_{i=0}^{m} (-1)^i g_i(\epsilon) \cdot A_t A_x^{2m-2i} A_y^{2m-2i} \overline{\Delta}_h^i \phi^n \\
\qquad + 2 \Bigg[\sum_{p=1}^{\frac{m+1}{2}} \overline{\Delta}_h^{p-1} A_x^{m-2p+2} A_y^{m-2p+2} A_t A_x^m A_y^m q^n \frac{\partial q^{n,\star}}{\partial \overline{\Delta}_h^{p-1} A_x^{m-2p+2} A_y^{m-2p+2} \phi^{n,\star}} \\
\qquad - \sum_{q=1}^{\frac{m-1}{2}} \overline{\nabla}_h \overline{\Delta}_h^{q-1} A_x^{m-2q+2} A_y^{m-2q+2} A_t A_x^m A_y^m q^n \frac{\partial q^{n,\star}}{\partial \overline{\nabla}_h \overline{\Delta}_h^{q-1} A_x^{m-2q+1} A_y^{m-2q+1} \phi^{n,\star}} \Bigg], \\
\delta_t^+ A_x^m A_y^m q^n = \sum_{p=1}^{\frac{m+1}{2}} \frac{\partial q^{n,\star}}{\partial \overline{\Delta}_h^{p-1} A_x^{m-2p+2} A_y^{m-2p+2} \phi^{n,\star}} \cdot \delta_t^+ \overline{\Delta}_h^{p-1} A_x^{m-2p+2} A_y^{m-2p+2} \phi^n \\
\qquad + \sum_{q=1}^{\frac{m-1}{2}} \frac{\partial q^{n,\star}}{\partial \overline{\nabla}_h \overline{\Delta}_h^{q-1} A_x^{m-2q+2} A_y^{m-2q+2} \phi^{n,\star}} \cdot \delta_t^+ \overline{\nabla}_h \overline{\Delta}_h^{q-1} A_x^{m-2q+2} A_y^{m-2q+2} \phi^n.
\end{cases}
$$

Algorithm 6 has a discrete transport equation for the discrete energy density.

Theorem 6.3. Model (2.62)-(2.64) satisfies the following discrete energy density transport equation

$$
\begin{aligned}
&\delta_t^+ E^n + 2\sum_{i=0}^{m}\sum_{2|i} g_i(\epsilon)\sum_{j=0}^{\frac{i}{2}}\Big(\nabla_h^{[m-j]}\cdot\Big(\delta_t^+\overline{\nabla}_h^j\phi^n\cdot A_tA_x^{m-(2i-j)}A_y^{m-(2i-j)}\overline{\nabla}_h^{2i-j-1}\phi^n\Big)\\
&-\nabla_h^{[m-(2i-2-j)]}\cdot\Big(A_t\overline{\nabla}_h^{2i-2-j}\phi^n\cdot\delta_t^+A_x^{m-(j+2)}A_y^{m-(j+2)}\overline{\nabla}_h^{j+1}\phi^n\Big)\Big)\\
&-2\sum_{i=0}^{m}\sum_{2\nmid i} g_i(\epsilon)\Bigg(\sum_{j=0}^{\frac{i+1}{2}}\nabla_h^{[m-j]}\cdot\Big(\delta_t^+\overline{\nabla}_h^j\phi^n\cdot A_tA_x^{m-(2i-j)}A_y^{m-(2i-j)}\overline{\nabla}_h^{2i-j-1}\phi^n\Big)\\
&-\sum_{j=0}^{\frac{i-1}{2}}\nabla_h^{[m-(2i-2-j)]}\cdot\Big(A_t\overline{\nabla}_h^{2i-2-j}\phi^n\cdot\delta_t^+A_x^{m-(j+2)}A_y^{m-(j+2)}\overline{\nabla}_h^{j+1}\phi^n\Big)\Bigg)\\
&-2\sum_{q=1}^{\frac{m-1}{2}}\sum_{l=0}^{q-1}\nabla_h^{[m-2l]}\cdot\Big(\overline{\Delta}_h^l\delta_t^+\phi^n\cdot\overline{\Delta}_h^{q-l-1}A_x^{m-2q+2l+1}A_y^{m-2q+2l+1}A_t\mathbf{h}_q^n\Big)\\
&+2\sum_{q=2}^{\frac{m-1}{2}}\sum_{l=0}^{q-2}\nabla_h^{[m-2l-1]}\cdot\Big(\overline{\nabla}_h\overline{\Delta}_h^lA_t\mathbf{h}_q^n\cdot\overline{\nabla}_h\overline{\Delta}_h^{q-l-2}A_x^{m-2q+2l+2}A_y^{m-2q+2l+2}\delta_t^+\phi^n\Big)\\
&+2\sum_{p=2}^{\frac{m+1}{2}}\Bigg\{\sum_{r=0}^{p-2}\nabla_h^{[m-2r]}\cdot\Big(\overline{\Delta}_h^r\delta_t^+\phi^n\cdot\overline{\nabla}_h\overline{\Delta}_h^{p-r-2}A_x^{m-2p+2r+2}A_y^{m-2p+2r+2}A_tk_p^n\\
&-\overline{\Delta}_h^rA_tk_p^n\cdot\overline{\nabla}_h\overline{\Delta}_h^{p-r-2}A_x^{m-2p+2r+2}A_y^{m-2p+2r+2}\delta_t^+\phi^n\Big)\Bigg\}\\
&+A_tA_x^mA_y^m\mu^n\mathcal{M}A_tA_x^mA_y^m\mu^n=0,
\end{aligned}
$$

with E^n the discrete energy density defined by (2.61).

Proof. We again omit the proof here. □

Appendix C: LEDRP algorithms for driven gradient flows

We list the algorithms and the corresponding discrete energy density transport equations for the driven gradient flow system in the following.

Algorithm 7 (LEDRP-I for driven gradient flows)**.** Given the free energy density in (2.18), the intermediate variables (2.19) - (2.20), the discrete system reads as follows

$$
\begin{cases}
\delta_t^+\phi^n=-\mathcal{M}A_t\mu^n+A_tg(\phi^n),\\
A_t\mu^n=\displaystyle\sum_{p=1}^{\frac{m+1}{2}}\Delta_h^{p-1}A_tk_p^n-\nabla_h^-\sum_{q=1}^{\frac{m+1}{2}}\Delta_h^{q-1}A_t\mathbf{h}_q^n.
\end{cases}
\tag{6.15}
$$

Next we state the local energy dissipation rate property for *Algorithm 7* as follows.

Theorem 6.4. Model (2.19) - (2.20), (6.15) has the following discrete energy density transport equation

$$\begin{aligned}
&\delta_t^+ E^n + \nabla_h^+ \cdot \Bigg(\sum_{p=2}^{\frac{m+1}{2}} \sum_{l=0}^{p-2} \Big(\delta_t^+ \Delta_h^l \phi^n \cdot \nabla_h^- \Delta_h^{p-2-l} A_t k_p^n - \delta_t^+ \nabla_h^- \Delta_h^l \phi^n \cdot \Delta_h^{p-2-l} A_t k_p^n \Big) \\
&\quad - \sum_{q=1}^{\frac{m+1}{2}} \delta_t^+ \Delta_h^{q-1} \phi^n \cdot A_t \overline{\mathbf{h}}_q^n \\
&\quad + \sum_{q=2}^{\frac{m+1}{2}} \sum_{r=0}^{q-2} \Big(-\delta_t^+ \Delta_h^r \phi^n \cdot \nabla_h^- \nabla_h^- \Delta_h^{q-2-r} A_t \mathbf{h}_q^n + \delta_t^+ \nabla_h^- \Delta_h^r \phi^n \cdot \nabla_h^- \Delta_h^{q-2-r} A_t \mathbf{h}_q^n \Big) \Bigg) \\
&\quad + A_t \mu^n \mathcal{M} A_t \mu^n = A_t g(\phi^n) \cdot A_t \mu^n,
\end{aligned}$$

with E the discrete energy density defined by (2.18).

Algorithm 8 (LEDRP-II for driven gradient flows)**.** Given the free energy density (2.28), the intermediate variables (2.29) - (2.30), the discrete gradient flow system reads as follows

$$\begin{cases} \delta_t^+ \phi^n = -\mathcal{M} A_t \mu^n + A_t g(\phi^n), \\ A_t \mu^n = \sum_{p=1}^{\frac{m+1}{2}} \Delta_h^{p-1} A_t k_p^n - \nabla_h^+ \sum_{q=1}^{\frac{m+1}{2}} \Delta_h^{q-1} A_t \mathbf{h}_q^n. \end{cases} \tag{6.16}$$

Theorem 6.5. Model (2.29) - (2.30), (6.16) has the following discrete energy density transport equation

$$\begin{aligned}
&\delta_t^+ E^n + \nabla_h^- \cdot \Bigg(\sum_{p=2}^{\frac{m+1}{2}} \sum_{l=0}^{p-2} \Big(\delta_t^+ \Delta_h^l \phi^n \cdot \nabla_h^+ \Delta_h^{p-2-l} A_t k_p^n - \delta_t^+ \nabla_h^+ \Delta_h^l \phi^n \cdot \Delta_h^{p-2-l} A_t k_p^n \Big) \\
&\quad - \sum_{q=1}^{\frac{m+1}{2}} \delta_t^+ \Delta_h^{q-1} \phi^n \cdot A_t \widetilde{\mathbf{h}}_q^n \\
&\quad + \sum_{q=2}^{\frac{m+1}{2}} \sum_{r=0}^{q-2} \Big(-\delta_t^+ \Delta_h^r \phi^n \cdot \Delta_h^{q-1-r} A_t \widetilde{\mathbf{h}}_q^n + \delta_t^+ \nabla_h^+ \Delta_h^r \phi^n \cdot \nabla_h^+ \Delta_h^{q-2-r} A_t \mathbf{h}_q^n \Big) \Bigg) \\
&\quad + A_t \mu^n \mathcal{M} A_t \mu^n = A_t g(\phi^n) \cdot A_t \mu^n,
\end{aligned}$$

with E the discrete energy density defined by (2.28).

Algorithm 9 (LEDRP-III for driven gradient flows)**.** Given the free energy density in (2.34), the intermediate variables (2.35) - (2.36), the discrete gradient flow system reads as follows

$$\begin{cases} \delta_t^+ A_x^m A_y^m \phi^n = -\mathcal{M} A_t A_x^m A_y^m \mu^n + A_t g(A_x^m A_y^m \phi^n), \\ A_t A_x^m A_y^m \mu^n = \sum_{p=1}^{\frac{m+1}{2}} \overline{\Delta}_h^{p-1} A_t A_x^{m-2p+2} A_y^{m-2p+2} k_p^n - \sum_{q=1}^{\frac{m+1}{2}} \overline{\nabla}_h \overline{\Delta}_h^{q-1} A_t A_x^{m-2q+1} A_y^{m-2q+1} \mathbf{h}_q^n. \end{cases} \tag{6.17}$$

Theorem 6.6. Model (2.35) - (2.36), (6.17) has the following energy density transport equation

$$
\begin{aligned}
&\delta_t^+ E^n - \sum_{q=1}^{\frac{m+1}{2}} \sum_{l=0}^{q-1} \nabla_h^{[m-2l]} \cdot \left(\overline{\Delta}_h^l \delta_t^+ \phi^n \cdot \overline{\Delta}_h^{q-l-1} A_x^{m-2q+2l+1} A_y^{m-2q+2l+1} A_t \mathbf{h}_q^n \right) \\
&+ \sum_{q=2}^{\frac{m+1}{2}} \sum_{l=0}^{q-2} \nabla_h^{[m-2l-1]} \cdot \left(\overline{\nabla}_h \overline{\Delta}_h^l A_t \mathbf{h}_q^n \cdot \overline{\nabla}_h \overline{\Delta}_h^{q-l-2} A_x^{m-2q+2l+2} A_y^{m-2q+2l+2} \delta_t^+ \phi^n \right) \\
&+ \sum_{p=2}^{\frac{m+1}{2}} \Big\{ \sum_{r=0}^{p-2} \nabla_h^{[m-2r]} \cdot \Big(\overline{\Delta}_h^r \delta_t^+ \phi^n \cdot \overline{\nabla}_h \overline{\Delta}_h^{p-r-2} A_x^{m-2p+2r+2} A_y^{m-2p+2r+2} A_t k_p^n \\
&- \overline{\Delta}_h^r A_t k_p^n \cdot \overline{\nabla}_h \overline{\Delta}_h^{p-r-2} A_x^{m-2p+2r+2} A_y^{m-2p+2r+2} \delta_t^+ \phi^n \Big) \Big\} \\
&+ A_t A_x^m A_y^m \mu^n \mathcal{M} A_t A_x^m A_y^m \mu^n = A_t g(A_x^m A_y^m \phi^n) \cdot A_t A_x^m A_y^m \mu^n,
\end{aligned}
$$

with E the discrete energy density defined by (2.34).

In the following, we list the algorithms and the discrete energy density transport equation for the driven gradient flow system based on the EQ technique.

Algorithm 10 (EQ-LEDRP-I for driven gradient flows)**.** Given the free energy density in (2.53) and intermediate variables (2.54) - (2.55), we discritize driven gradient flow (3.5) as follows

$$
\begin{cases}
\delta_t^+ \phi^n = -\mathcal{M} A_t \mu^n + g(\phi^{n,\star}), \\
A_t \mu^n = 2 \sum_{i=0}^{m} (-1)^i g_i(\epsilon) \cdot A_t \Delta_h^i \phi^n + 2 \left[\sum_{p=1}^{\frac{m+1}{2}} \Delta_h^{p-1} A_t k_p - \sum_{q=1}^{\frac{m-1}{2}} \nabla_h^- \Delta_h^{q-1} A_t \mathbf{h}_q \right], \\
\delta_t^+ q^n = \sum_{p=1}^{\frac{m+1}{2}} \frac{\partial q^{n,\star}}{\partial \Delta_h^{p-1} \phi^{n,\star}} \cdot \delta_t^+ \Delta_h^{p-1} \phi^n + \sum_{q=1}^{\frac{m-1}{2}} \frac{\partial q^{n,\star}}{\partial \nabla_h^+ \Delta_h^{q-1} \phi^{n,\star}} \cdot \delta_t^+ \nabla_h^+ \Delta_h^{q-1} \phi^n.
\end{cases}
\tag{6.18}
$$

Theorem 6.7. System (6.18) admits the following discrete energy density transport equation

$$
\begin{aligned}
&\delta_t^+ E^n + 2 \nabla_h^+ \cdot \Bigg(\sum_{p=2}^{\frac{m+1}{2}} \sum_{l=0}^{p-2} \left(\delta_t^+ \Delta_h^l \phi^n \cdot \nabla_h^- \Delta_h^{p-2-l} A_t k_p^n + \delta_t^+ \nabla_h^- \Delta_h^l \phi^n \cdot \Delta_h^{p-2-l} A_t k_p^n \right) - \delta_t^+ \phi^n \cdot A_t \overline{\mathbf{h}}_1^n \\
&+ \sum_{q=2}^{\frac{m-1}{2}} \left(\sum_{r=0}^{q-2} \left(-\delta_t^+ \Delta_h^r \phi^n \cdot \nabla_h^- \nabla_h^- \Delta_h^{q-2-r} A_t \mathbf{h}_q^n + \delta_t^+ \nabla_h^- \Delta_h^r \phi^n \cdot \nabla_h^- \Delta_h^{q-2-r} A_t \mathbf{h}_q^n \right) - \delta_t^+ \Delta_h^{q-1} \phi^n \cdot A_t \overline{\mathbf{h}}_q^n \right) \\
&+ \sum_{i=1}^{m} g_i(\epsilon) \left(- \sum_{l=\frac{i-1}{2}}^{i-1} \nabla_h^- \Delta_h^l A_t \phi^n \cdot \delta_t^+ \Delta_h^{i-1-l} \phi^n + \sum_{l=\frac{i+1}{2}}^{i-1} \Delta_h^l A_t \phi^n \cdot \delta_t^+ \nabla_h^- \Delta_h^{i-1-l} \phi^n \right) \\
&+ \sum_{i=1}^{m} g_i(\epsilon) \sum_{l=\frac{i}{2}}^{i-1} \left(\nabla_h^- \Delta_h^l A_t \phi^n \cdot \delta_t^+ \Delta_h^{i-1-l} \phi^n - \Delta_h^l A_t \phi^n \cdot \delta_t^+ \nabla_h^- \Delta_h^{i-1-l} \phi^n \right) \Bigg) \\
&+ A_t \mu^n \mathcal{M} A_t \mu^n = g(\phi^{n,\star}) A_t \mu^n,
\end{aligned}
$$

Algorithm 11 (EQ-LEDRP-II for driven gradient flows)**.** Given the discrete energy density in (2.57), and intermediate variables (2.58) - (2.59), we discretize driven gradient flow (3.5) as follows

$$\begin{cases} \delta_t^+ \phi^n = -\mathcal{M} A_t \mu^n + g(\phi^{n,\star}), \\ A_t \mu^n = 2 \sum_{i=0}^{m} (-1)^i g_i(\epsilon) \cdot A_t \Delta_h^i \phi^n + 2 \left[\sum_{p=1}^{\frac{m+1}{2}} \Delta_h^{p-1} A_t k_p - \sum_{q=1}^{\frac{m-1}{2}} \nabla_h^+ \Delta_h^{q-1} A_t \mathbf{h}_q \right], \\ \delta_t^+ q^n = \sum_{p=1}^{\frac{m+1}{2}} \frac{\partial q^{n,\star}}{\partial \Delta_h^{p-1} \phi^{n,\star}} \cdot \delta_t^+ \Delta_h^{p-1} \phi^n + \sum_{q=1}^{\frac{m-1}{2}} \frac{\partial q^{n,\star}}{\partial \nabla_h^- \Delta_h^{q-1} \phi^{n,\star}} \cdot \delta_t^+ \nabla_h^- \Delta_h^{q-1} \phi^n. \end{cases} \tag{6.19}$$

Theorem 6.8. Model (6.19) satisfies the following discrete energy density transport equation

$$\begin{aligned} &\delta_t^+ E^n + 2 \nabla_h^- \cdot \left(\sum_{p=2}^{\frac{m+1}{2}} \sum_{l=0}^{p-2} \left(\delta_t^+ \Delta_h^l \phi^n \cdot \nabla_h^+ \Delta_h^{p-2-l} A_t k_p^n - \delta_t^+ \nabla_h^+ \Delta_h^l \phi^n \cdot \Delta_h^{p-2-l} A_t k_p^n \right) - \delta_t^+ \phi^n \cdot A_t \widetilde{\mathbf{h}}_1^n \right. \\ &+ \sum_{q=2}^{\frac{m-1}{2}} \left(\sum_{r=0}^{q-2} \left(-\delta_t^+ \Delta_h^r \phi^n \cdot \Delta_h^{q-1-r} A_t \widetilde{\mathbf{h}}_q^n + \delta_t^+ \nabla_h^+ \Delta_h^r \phi^n \cdot \nabla_h^+ \Delta_h^{q-2-r} A_t \mathbf{h}_q^n \right) - \delta_t^+ \Delta_h^{q-1} \nabla_h^- \phi^n \cdot A_t \widetilde{\mathbf{h}}_q^n \right) \\ &+ \sum_{i=1}^{m} g_i(\epsilon) \left(- \sum_{l=\frac{i-1}{2}}^{i-1} \nabla_h^+ \Delta_h^l A_t \phi^n \cdot \delta_t^+ \Delta_h^{i-1-l} \phi^n + \sum_{l=\frac{i+1}{2}}^{i-1} \Delta_h^l A_t \phi^n \cdot \delta_t^+ \nabla_h^+ \Delta_h^{i-1-l} \phi^n \right) \\ &\left. + \sum_{i=1}^{m} g_i(\epsilon) \sum_{l=\frac{i}{2}}^{i-1} \left(\nabla_h^+ \Delta_h^l A_t \phi^n \cdot \delta_t^+ \Delta_h^{i-1-l} \phi^n - \Delta_h^l A_t \phi^n \cdot \delta_t^+ \nabla_h^+ \Delta_h^{i-1-l} \phi^n \right) \right) \\ &+ A_t \mu^n \mathcal{M} A_t \mu^n = g(\phi^{n,\star}) A_t \mu^n, \end{aligned}$$

with discrete energy density E^n defined in (2.57).

Algorithm 12 (EQ-LEDRP-III for driven gradient flows)**.** Given the discrete energy density in (2.61), and intermediate variables (2.62) - (2.63), we discretize system (3.5) as follows

$$\begin{cases} \delta_t^+ A_x^m A_y^m \phi^n = -\mathcal{M} A_t A_x^m A_y^m \mu^n + g(A_x^m A_y^m \phi^{n,\star}), \\ A_t A_x^m A_y^m \mu^n = 2 \sum_{i=0}^{m} (-1)^i g_i(\epsilon) \cdot A_t A_x^{m-2i} A_y^{m-2i} \overline{\Delta}_h^i \phi^n \\ \qquad\qquad + 2 \left[\sum_{p=1}^{\frac{m+1}{2}} \overline{\Delta}_h^{p-1} A_t A_x^{m-2p+2} A_y^{m-2p+2} k_p^n - \sum_{q=1}^{\frac{m-1}{2}} \overline{\nabla}_h \overline{\Delta}_h^{q-1} A_t A_x^{m-2q+2} A_y^{m-2q+2} \mathbf{h}_q^n \right], \\ \delta_t^+ A_x^m A_y^m q^n = \sum_{p=1}^{\frac{m+1}{2}} \frac{\partial q^{n,\star}}{\partial \overline{\Delta}_h^{p-1} A_x^{m-2p+2} A_y^{m-2p+2} \phi^{n,\star}} \cdot \delta_t^+ \overline{\Delta}_h^{p-1} A_x^{m-2p+2} A_y^{m-2p+2} \phi^n \\ \qquad\qquad + \sum_{q=1}^{\frac{m-1}{2}} \frac{\partial q^{n,\star}}{\partial \overline{\nabla}_h \overline{\Delta}_h^{q-1} A_x^{m-2q+2} A_y^{m-2q+2} \phi^{n,\star}} \cdot \delta_t^+ \overline{\nabla}_h \overline{\Delta}_h^{q-1} A_x^{m-2q+2} A_y^{m-2q+2} \phi^n. \end{cases} \tag{6.20}$$

Theorem 6.9. Model (6.20) satisfies the following discrete energy density transport equation

$$
\begin{aligned}
&\delta_t^+ E^n + 2\sum_{i=0}^{m}\sum_{2|i} g_i(\epsilon)\sum_{j=0}^{\frac{i}{2}}\Big(\nabla_h^{[m-j]}\cdot\Big(\delta_t^+\overline{\nabla}_h^j\phi^n\cdot A_tA_x^{m-(2i-j)}A_y^{m-(2i-j)}\overline{\nabla}_h^{2i-j-1}\phi^n\Big)\\
&\quad-\nabla_h^{[m-(2i-2-j)]}\cdot\Big(A_t\overline{\nabla}_h^{2i-2-j}\phi^n\cdot\delta_t^+A_x^{m-(j+2)}A_y^{m-(j+2)}\overline{\nabla}_h^{j+1}\phi^n\Big)\Big)\\
&\quad-2\sum_{i=0}^{m}\sum_{2\nmid i} g_i(\epsilon)\Bigg(\sum_{j=0}^{\frac{i+1}{2}}\nabla_h^{[m-j]}\cdot\Big(\delta_t^+\overline{\nabla}_h^j\phi^n\cdot A_tA_x^{m-(2i-j)}A_y^{m-(2i-j)}\overline{\nabla}_h^{2i-j-1}\phi^n\Big)\\
&\quad-\sum_{j=0}^{\frac{i-1}{2}}\nabla_h^{[m-(2i-2-j)]}\cdot\Big(A_t\overline{\nabla}_h^{2i-2-j}\phi^n\cdot\delta_t^+A_x^{m-(j+2)}A_y^{m-(j+2)}\overline{\nabla}_h^{j+1}\phi^n\Big)\Bigg)\\
&\quad-2\sum_{q=1}^{\frac{m-1}{2}}\sum_{l=0}^{q-1}\nabla_h^{[m-2l]}\cdot\Big(\overline{\Delta}_h^l\delta_t^+\phi^n\cdot\overline{\Delta}_h^{q-l-1}A_x^{m-2q+2l+1}A_y^{m-2q+2l+1}A_t\mathbf{h}_q^n\Big)\\
&\quad+2\sum_{q=2}^{\frac{m-1}{2}}\sum_{l=0}^{q-2}\nabla_h^{[m-2l-1]}\cdot\Big(\overline{\nabla}_h\overline{\Delta}_h^lA_t\mathbf{h}_q^n\cdot\overline{\nabla}_h\overline{\Delta}_h^{q-l-2}A_x^{m-2q+2l+2}A_y^{m-2q+2l+2}\delta_t^+\phi^n\Big)\\
&\quad+2\sum_{p=2}^{\frac{m+1}{2}}\Bigg\{\sum_{r=0}^{p-2}\nabla_h^{[m-2r]}\cdot\Big(\overline{\Delta}_h^r\delta_t^+\phi^n\cdot\overline{\nabla}_h\overline{\Delta}_h^{p-r-2}A_x^{m-2p+2r+2}A_y^{m-2p+2r+2}A_tk_p^n\\
&\quad-\overline{\Delta}_h^rA_tk_p^n\cdot\overline{\nabla}_h\overline{\Delta}_h^{p-r-2}A_x^{m-2p+2r+2}A_y^{m-2p+2r+2}\delta_t^+\phi^n\Big)\Bigg\}\\
&\quad+A_tA_x^mA_y^m\mu^n\mathcal{M}A_tA_x^mA_y^m\mu^n=g(A_x^mA_y^m\phi^{n,\star})A_tA_x^mA_y^m\mu^n,
\end{aligned}
$$

with E^n the discrete energy density defined by (2.61).

Appendix D: Proof of some theorems in §4

a. Proof for Theorem 4.1: Multiplying $\boldsymbol{\mu}$ and $\mathbf{u}_t$ on both sides of the first and the second equation in (4.9), we have

$$
\begin{cases}
\mathbf{u}_t\cdot\boldsymbol{\mu}=-\boldsymbol{\mu}^T\cdot\boldsymbol{M}(\Theta)\cdot\boldsymbol{\mu}+\mathbf{g}(\mathbf{u})\cdot\boldsymbol{\mu},\\
\boldsymbol{\mu}\cdot\mathbf{u}_t=\begin{pmatrix}-\epsilon^2\nabla\cdot\left[\xi_{s,n}(\Theta)^2\nabla\phi\right]+\epsilon^2\partial_x\left[\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_y\phi\right]-\epsilon^2\partial_y\left[\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_x\phi\right]+f'(\phi)\\ \frac{u}{\tau_v}-\nabla\cdot(D\cdot\nabla u)\end{pmatrix}\\
\qquad\cdot\mathbf{u}_t.
\end{cases}
\tag{6.21}
$$

Using derivative rule, we arrive at

$$\begin{aligned}
&\left[\epsilon^2\xi_{s,n}(\Theta)^2\nabla\phi+\begin{pmatrix}-\epsilon^2\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_y\phi\\ \epsilon^2\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_x\phi\end{pmatrix}\right]\cdot\nabla\phi_t+f'(\phi)\phi_t+\frac{u}{\tau_v}u_t\\
&+\nabla u^T\cdot D\cdot\nabla u_t+\nabla\cdot\left[-\epsilon^2\xi_{s,n}(\Theta)^2\nabla\phi\cdot\phi_t+\begin{pmatrix}\epsilon^2\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_y\phi\\ -\epsilon^2\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_x\phi\end{pmatrix}\cdot\phi_t-D\cdot\nabla u\cdot u_t\right]\\
&+\boldsymbol{\mu}^T\cdot\boldsymbol{M}(\Theta)\cdot\boldsymbol{\mu}=\mathbf{g}(\mathbf{u})\cdot\boldsymbol{\mu},
\end{aligned}\tag{6.22}$$

from the definition of the local free energy, (4.13), we have

$$\frac{\mathrm{d}E}{\mathrm{d}t}=\left[\epsilon^2\xi_{s,n}(\Theta)^2\nabla\phi+\begin{pmatrix}-\epsilon^2\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_y\phi\\ \epsilon^2\xi_{s,n}(\Theta)\xi'_{s,n}(\Theta)\partial_x\phi\end{pmatrix}\right]\cdot\nabla\phi_t+f'(\phi)\phi_t+\frac{u}{\tau_v}u_t+\nabla u^T\cdot D\cdot\nabla u_t,\tag{6.23}$$

Combing the results, we complete the proof.

b. Proof of Theorem 4.2: The proof is similar to that of Theorem 4.1 and is thus omitted.

c. Proof of Theorem 4.3: Multiplying $A_t\boldsymbol{\mu}^n$ and $\delta_t^+\mathbf{u}^n$ on both sides of the second and third line in (4.28), respectively, we have

$$\begin{cases}\delta_t^+\mathbf{u}^n\cdot A_t\boldsymbol{\mu}^n=-(A_t\boldsymbol{\mu}^n)^T\cdot\boldsymbol{M}\left(\overline{\Theta}^{n,\star}\right)\cdot A_t\boldsymbol{\mu}^{\boldsymbol{n}}+\mathbf{g}(\mathbf{u}^{n,\star})\cdot A_t\boldsymbol{\mu}^n,\\ A_t\boldsymbol{\mu}^n\cdot\delta_t^+\mathbf{u}^n=A_t\boldsymbol{k}^n\cdot\delta_t^+\mathbf{u}^n-\nabla_h^-\cdot A_t\boldsymbol{H}^n\cdot\delta_t^+\mathbf{u}^n.\end{cases}\tag{6.24}$$

Inserting the first and second equation which are in the first line of (4.28) into (6.24), we obtain

$$\begin{aligned}
&\begin{pmatrix}\frac{1}{2}A_tV^n\cdot P^{n,\star}\\ \frac{A_tu^n}{\tau_v}\end{pmatrix}\cdot\delta_t^+\mathbf{u}^n-\nabla_h^-\cdot\begin{pmatrix}\epsilon^2A_tU^n\cdot\boldsymbol{R}^{n,\star}\\ D\cdot\nabla_h^+A_tu^n\end{pmatrix}\cdot\delta_t^+\mathbf{u}^n\\
&=-(A_t\boldsymbol{\mu}^n)^T\cdot\boldsymbol{M}\left(\overline{\Theta}^{n,\star}\right)\cdot A_t\boldsymbol{\mu}^{\boldsymbol{n}}+\mathbf{g}(\mathbf{u}^{n,\star})\cdot A_t\boldsymbol{\mu}^n.
\end{aligned}\tag{6.25}$$

With the aid of *Lemma 2.3*, we have

$$\begin{aligned}
&\epsilon^2A_tU^n\cdot\boldsymbol{R}^{n,\star}\cdot\delta_t^+\nabla_h^+\phi^n+\frac{1}{2}A_tV^n\cdot P^{n,\star}\cdot\delta_t^+\phi^n\\
&+\frac{A_tu^n}{\tau_v}\delta_t^+u^n+(\delta_t^+\nabla_h^+u^n)^T\cdot D\cdot\nabla_h^+A_tu^n-\nabla_h^+\cdot\left(A_t\overline{\boldsymbol{H}}^n\cdot\delta_t^+\mathbf{u}^n\right)\\
&+(A_t\boldsymbol{\mu}^n)^T\cdot\boldsymbol{M}(\overline{\Theta}^{n,\star})\cdot A_t\boldsymbol{\mu}^n=\mathbf{g}\left(\mathbf{u}^{n,\star}\right)\cdot A_t\boldsymbol{\mu}^n.
\end{aligned}\tag{6.26}$$

Inserting the last two equations of (4.28) into the time derivative of local energy (4.27), we arrive at

$$\begin{aligned}
\delta_t^+E^n=&\,\epsilon^2A_tU^n\cdot\boldsymbol{R}^{n,\star}\cdot\delta_t^+\nabla_h^+\phi^n+\frac{1}{2}A_tV^n\cdot P^{n,\star}\cdot\delta_t^+\phi^n\\
&+\frac{A_tu^n}{\tau_v}\delta_t^+u^n+(\delta_t^+\nabla_h^+u^n)^T\cdot D\cdot\nabla_h^+A_tu^n.
\end{aligned}\tag{6.27}$$

Combining all the results, we complete the proof.

d. Proof of Theorem 4.4: We multiply $A_t\boldsymbol{\mu}^n$ and $\delta_t^+\mathbf{u}^n$ on both sides of the second and third equation in (4.34), respectively, we have

$$\begin{cases} \delta_t^+\mathbf{u}^n \cdot A_t\boldsymbol{\mu}^n = -(A_t\boldsymbol{\mu}^n)^T \cdot \boldsymbol{M}\left(\overline{\Theta}^{n,\star}\right) \cdot A_t\boldsymbol{\mu}^{\boldsymbol{n}} + \mathbf{g}(\mathbf{u}^{n,\star}) \cdot A_t\boldsymbol{\mu}^n, \\ A_t\boldsymbol{\mu}^n \cdot \delta_t^+\mathbf{u}^n = A_t\boldsymbol{k}^n \cdot \delta_t^+\mathbf{u}^n - \nabla_h^+ \cdot A_t\boldsymbol{H}^n \cdot \delta_t^+\mathbf{u}^n, \end{cases} \tag{6.28}$$

Inserting the first and second equation which are in the first line of (4.34) into (6.28), we have

$$\begin{aligned} &\begin{pmatrix} \frac{1}{2}A_tV^n \cdot P^{n,\star} \\ \frac{A_tu^n}{\tau_v} \end{pmatrix} \cdot \delta_t^+\mathbf{u}^n - \nabla_h^+ \cdot \begin{pmatrix} \epsilon^2 A_tU^n \cdot \boldsymbol{R}^{n,\star} \\ D \cdot \nabla_h^- A_tu^n \end{pmatrix} \cdot \delta_t^+\mathbf{u}^n \\ &= -(A_t\boldsymbol{\mu}^n)^T \cdot \boldsymbol{M}\left(\overline{\Theta}^{n,\star}\right) \cdot A_t\boldsymbol{\mu}^{\boldsymbol{n}} + \mathbf{g}(\mathbf{u}^{n,\star}) \cdot A_t\boldsymbol{\mu}^n. \end{aligned} \tag{6.29}$$

With the aid of *Lemma 2.4*, we have

$$\begin{aligned} &\epsilon^2 A_tU^n \cdot \boldsymbol{R}^{n,\star} \cdot \delta_t^+\nabla_h^-\phi^n + \frac{1}{2}A_tV^n \cdot P^{n,\star} \cdot \delta_t^+\phi^n \\ &+ \frac{A_tu^n}{\tau_v}\delta_t^+u^n + (\delta_t^+\nabla_h^-u^n)^T \cdot D \cdot \nabla_h^- A_tu^n - \nabla_h^- \cdot \left(A_t\widetilde{\boldsymbol{H}}^n \cdot \delta_t^+\mathbf{u}^n\right) \\ &+ (A_t\boldsymbol{\mu}^n)^T \cdot \boldsymbol{M}(\overline{\Theta}^{n,\star}) \cdot A_t\boldsymbol{\mu}^n = \mathbf{g}\left(\mathbf{u}^{n,\star}\right) \cdot A_t\boldsymbol{\mu}^n. \end{aligned} \tag{6.30}$$

Inserting the last two equations of (4.34) into the time derivative of energy density (4.33), we arrive at

$$\delta_t^+E^n = \epsilon^2 A_tU^n \cdot \boldsymbol{R}^{n,\star} \cdot \delta_t^+\nabla_h^-\phi^n + \frac{1}{2}A_tV^n \cdot P^{n,\star} \cdot \delta_t^+\phi^n + \frac{A_tu^n}{\tau_v}\delta_t^+u^n + (\delta_t^+\nabla_h^-u^n)^T \cdot D \cdot \nabla_h^- A_tu^n. \tag{6.31}$$

340 Inserting (6.31) into (6.30), we complete the proof.

e. Proof of Theorem 4.5: Multiplying $A_tA_xA_y\boldsymbol{\mu}^n$, $\delta_t^+A_xA_y\mathbf{u}^n$ on both sides of the second and third equation in(4.39), we obtain

$$\begin{cases} \delta_t^+A_xA_y\mathbf{u}^n \cdot A_tA_xA_y\boldsymbol{\mu}^n = -\left(A_tA_xA_y\boldsymbol{\mu}^n\right)^T \cdot \boldsymbol{M}(\overline{\Theta}^{n,\star}) \cdot A_tA_xA_y\boldsymbol{\mu}^n + \mathbf{g}(A_xA_y\mathbf{u}^{n,\star}) \cdot A_tA_xA_y\boldsymbol{\mu}^n, \\ A_tA_xA_y\boldsymbol{\mu}^n \cdot \delta_t^+A_xA_y\mathbf{u}^n = A_tA_xA_y\boldsymbol{k}^n \cdot \delta_t^+A_xA_y\mathbf{u}^n - \overline{\nabla}_h \cdot A_t\boldsymbol{H}^n \cdot \delta_t^+A_xA_y\mathbf{u}^n. \end{cases} \tag{6.32}$$

Combing the two equations, we have

$$\begin{aligned} &\begin{pmatrix} \frac{1}{2}A_tA_xA_yV^n \cdot A_xA_yP^{n,\star} \\ \frac{A_tA_xA_yu^n}{\tau_v} \end{pmatrix} \cdot \delta_t^+A_xA_y\mathbf{u}^n - \overline{\nabla}_h \cdot A_t\boldsymbol{H}^n \cdot \delta_t^+A_xA_y\mathbf{u}^n \\ &= -\left(A_tA_xA_y\boldsymbol{\mu}^n\right)^T \cdot \boldsymbol{M}(\overline{\Theta}^{n,\star}) \cdot A_tA_xA_y\boldsymbol{\mu}^n + \mathbf{g}(A_xA_y\mathbf{u}^{n,\star}) \cdot A_tA_xA_y\boldsymbol{\mu}^n, \end{aligned} \tag{6.33}$$

with the aid of *Lemma 4.1*, the term $-\overline{\nabla}_h \cdot A_t\boldsymbol{H}^n \cdot \delta_t^+A_xA_y\mathbf{u}^n$ can be transformed into

$$-\overline{\nabla}_h \cdot A_t\boldsymbol{H}^n \cdot \delta_t^+A_xA_y\mathbf{u}^n = -\nabla_h^{[1]} \cdot \left(\delta_t^+\mathbf{u}^n \cdot A_t\boldsymbol{H}^n\right) + \begin{pmatrix} \epsilon^2 A_tA_xA_yU^n \cdot A_xA_y\boldsymbol{R}^{n,\star} \\ D \cdot \overline{\nabla}_h A_tu^n \end{pmatrix} \cdot \delta_t^+\overline{\nabla}_h\mathbf{u}^n. \tag{6.34}$$

Then, we have

$$\begin{aligned}
&\epsilon^2 A_t A_x A_y U^n \cdot A_x A_y \boldsymbol{R}^{n,\star} \cdot \delta_t^+ \overline{\nabla}_h \phi^n + \frac{1}{2} A_t A_x A_y V^n \cdot A_x A_y P^{n,\star} \cdot \delta_t^+ A_x A_y \phi^n + \frac{A_t A_x A_y u^n}{\tau_v} \delta_t^+ A_x A_y u^n \\
&+ (\delta_t^+ \overline{\nabla}_h u^n)^T \cdot D \cdot \overline{\nabla}_h A_t u^n - \nabla_h^{[1]} \cdot \left(A_t \boldsymbol{H}^n \cdot \delta_t^+ \mathbf{u}^n \right) + (A_t A_x A_y \boldsymbol{\mu}^n)^T \cdot \boldsymbol{M}(\overline{\Theta}^{n,\star}) \cdot A_t A_x A_y \boldsymbol{\mu}^n \\
&= \mathbf{g}(A_x A_y \mathbf{u}^{n,\star}) \cdot A_t A_x A_y \boldsymbol{\mu}^n
\end{aligned} \tag{6.35}$$

Inserting the last two equations of (4.39) into (4.38) arrives at

$$\begin{aligned}
\delta_t^+ E^n = &\epsilon^2 A_t A_x A_y U^n \cdot A_x A_y \boldsymbol{R}^{n,\star} \cdot \delta_t^+ \overline{\nabla}_h \phi^n + \frac{1}{2} A_t A_x A_y V^n \cdot A_x A_y P^{n,\star} \cdot \delta_t^+ A_x A_y \phi^n \\
&+ \frac{A_t A_x A_y u^n}{\tau_v} \delta_t^+ A_x A_y u^n + (\delta_t^+ \overline{\nabla}_h u^n)^T \cdot D \cdot \overline{\nabla}_h A_t u^n.
\end{aligned} \tag{6.36}$$

Inserting (6.36) into (6.35), we complete the proof.

7. References

References

[1] S.M. Allen and J.W. Cahn. Ground state structures in ordered binary alloys with second neighbor interactions. *Acta Metall.*, 20(423), 1972.

[2] J.W. Cahn and J.E. Hilliard. Free energy of a nonuniform system. I. Interfacial free energy. *J. Chem. Phys.*, 28(2):258–267, 1958.

[3] M. Doi and S.F. Edwards. *The Theory of Polymer Dynamics*. Oxford University Press, United Kingdom, 1986.

[4] T.W. Heo, K.B. Colas, A.T. Motta, and L.Q. Chen. A phase-field model for hydride formation in polycrystalline metals: Application to δ-hydride in zirconium alloys. *Acta Mater.*, 181:262–277, 2019.

[5] M. Doi. Onsager's variational principle in soft matter. *J. Phys.: Condens. Matter*, 23:284118, 2011.

[6] X. Yang, M.G. Forest, and Q. Wang. Near equilibrium dynamics and one-dimensional spatial-temporal structures of polar active liquid crystals. *Chin. Phys. B*, 23:118701, 2014.

[7] X. Yang, J. Li, M.G. Forest, and Q. Wang. Hydrodynamic theories for flows of active liquid crystals and the generalized onsager principle. *Entropy*, 18:202, 2016.

[8] Q. Wang. Generalized onsager principle and its application. *Frontiers and Progress of Current Soft Matter Research*, edited by Xiang-you Liu, 2020.

[9] T. Michely and J. Krug. *Islands, Mounds, and Atoms: Patterns andProcesses in Crystal Growth Far from Equilibrium*. Springer, 2004.

[10] J. Zhuang, W. Zhao, L. Qiu, J. Xin, J. Dong, and F. Ding. Morphology evolution of graphene during chemical vapor deposition growth: A phase-field theory simulation. *J. Phys. Chem. C*, 123:9902–9908, 2019.

[11] C. Mattevi, H. Kim, and M. Chhowalla. A review of chemical vapour deposition of graphene on copper. *J. Materials Chem.*, 21:3324, 2011.

[12] Y. Hao, M.S. Bharathi, L. Wang, Y. Liu, and H. Chen et al. The role of surface oxygen in the growth of large single-crystal graphene on copper. *Science*, 342:720–723, 2013.

[13] B. Wu, D. Geng, Z. Xu, Y. Guo, L. Huang, Y. Xue, J. Chen, G. Yu, and Y. Liu. Self-organized graphene crystal patterns. *Npg Asia Mater.*, 5:e36, 2013.

[14] B. Wu, D. Geng, Z. Xu, Y. Guo, L. Huang, Y. Xue, J. Chen, G. Yu, and Y. Liu. Phase-field modeling of two-dimensional crystal growth with anisotropic diffusion. *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.*, 88:052409, 2013.

[15] K.M.F. Shahil and A.A. Balandin. Thermal properties of graphene and multilayer graphene: Applications in thermal interface materials. *Solid State Commun.*, 152(15):1331–1340, 2012.

[16] P. Avouris and F. Xia. Graphene applications in electronics and photonics. *Mrs Bull.*, 37(12):1225, 2012.

[17] Y. Zhu, S. Murali, W. Cai, X. Li, J.W. Suk, J.R. Potts, and R.S. Ruoff. Graphene and graphene oxide: synthesis, properties, and applications. *Adv. Mater.*, 22(35):3906–3924, 2010.

[18] G. Jo, M. Choe, S. Lee, W. Park, Y. H. Kahng, and T. Lee. The application of graphene as electrodes in electrical and optical devices. *Nanotechnology*, 23(11):112001, 2012.

[19] X. Li, W. Cai, J. An, S. Kim, J. Nah, D. Yang, R. Piner, A. Velamakanni, I. Jung, E. Tutuc, S.K. Banerjee, L. Colombo, and R.S. Ruoff. Large-area synthesis of high-quality and uniform graphene films on copper foils. *Science*, 324:1312–1314, 2009.

[20] J. Song, F.Y. Kam, R.Q. Png, W.L. Seah, J.M. Zhuo, G.K. Lim, P.K.H. Ho, and L.L. Chua. A general method for transferring graphene onto soft surfaces. *Nat. Nanotechnol.*, 8:356–362, 2013.

[21] E. Meca, J. Lowengrub, H. Kim, C. Mattevi, and V.B. Shenoy. Epitaxial graphene growth and shape dynamics on copper: Phase- field modeling and experiments. *Nano Lett.*, 13:5692–5697, 2013.

[22] L. Sun, L. Lin, J. Zhang, H. Wang, H. Peng, and Z. Liu. Visualizing fast growth of large single-crystalline graphene by tunable isotopic carbon source. *Nano Res.*, 10:355–363, 2016.

[23] Y. Zhang, L. Zhang, P. Kim, M. Ge, Z. Li, and C. Zhou. Vapor trapping growth of single-crystalline graphene flowers: Synthesis, morphology, and electronic properties. *Nano Lett.*, 12:2810–2816, 2012.

[24] J.M. Wofford, S. Nie, K.F. McCarty, N.C. Bartelt, , and O.D. Dubon. Graphene islands on cu foils: The interplay between shape, orientation, and defects. *Nano Lett.*, 10:4890–4896, 2010.

[25] H.I. Rasool, E.B. Song, M. Mecklenburg, B.C. Regan, K.L. Wang, B.H. Weiller, and J.K. Gimzewski. Atomic-scale characterization of graphene grown on copper (100) single crystals. *J. Am. Chem. Soc.*, 133:12536–12543, 2011.

[26] H. Wang, G. Wang, P. Bao, S. Yang, W. Zhu, X. Xie, and W. Zhang. Controllable synthesis of submillimeter single-crystal monolayer graphene domains on copper foils by suppressing nucleation. *J. Am. Chem. Soc.*, 134:3627–3630, 2012.

[27] E. Yokoyama and R.F. Sekerka. A numerical study of the combined effect of anisotropic surface tension and interface kinetics on pattern formation during the growth of two-dimensional crystals. *J. Cryst. Growth*, 125:389–403, 1992.

[28] C.M. Elliott and A.M. Stuart. The global dynamics of discrete semilinear parabolic equations. *SIAM J. Numer. Anal.*, 30(6):1622–1663, 1993.

[29] W. Chen, S. Conde, C. Wang, X. Wang, and S. Wise. A linear energy stable scheme for a thin film model without slope selection. *J. Sci. Comput.*, 52:546–562, 2011.

[30] C. Wang and S.M. Wise. An energy stable and convergent finite-difference scheme for the modified phase field crystal equation. *SIAM J. Numer. Anal.*, 49(3):945–969, 2011.

[31] Z. Guan, J.S. Lowengrub, C. Wang, and S.M. Wise. Second order convex splitting schemes for periodic nonlocal Cahn-Hilliard and Allen-Cahn equations. *J. Comput. Phys.*, 277:48–71, 2014.

[32] A. Christlieb, J. Jones, K. Promislow, B. Wetton, and M. Willoughby. High accuracy solutions to energy gradient flows from material science models. *J. Comput. Phys.*, 257:193–215, 2014.

[33] J. Shen and X. Yang. Numerical approximations of Allen-Cahn and Cahn-Hilliard equations. *Discrete Contin. Dyn. Syst*, 28(4):1669–1691, 2010.

[34] F. Guillén-González and G. Tierra. On linear schemes for a Cahn-Hilliard diffuse interface model. *J. Comput. Phys.*, 234:140–171, 2013.

[35] X. Yang and L. Ju. Efficient linear schemes with unconditional energy stability for the phase field elastic bending energy model. *Comput. Methods Appl. M.*, 315:691–712, 2017.

[36] X. Yang, J. Zhao, and Q. Wang. Numerical approximations for the molecular beam epitaxial growth model based on the invariant energy quadratization method. *J. Comput. Phys.*, 333:104–127, 2017.

[37] X. Yang, J. Zhao, and X. He. Linear, second order and unconditionally energy stable schemes for the viscous Cahn–Hilliard equation with hyperbolic relaxation using the invariant energy quadratization method. *J. Comput. Appl. Math.*, 343:80–97, 2018.

[38] X. Yang. Numerical approximations for the cahn–hilliard phase field model of the binary fluid-surfactant system. *Journal of Scientific Computing*, 74(3):1533–1553, 2018.

[39] X. Zhao and Q. Wang. A second order fully-discrete linear energy stable scheme for a binary compressible viscous fluid model. *J. Comput. Phys.*, 395:382–409, 2019.

[40] Y. Gong and J. Zhao. Energy-stable runge-kutta schemes for gradient flow models using the energy quadratization approach. *Appl. Math. Lett.*, 94:224–231, 2019.

[41] J. Zhao, X. Yang, Y. Gong, and Q. Wang. A novel linear second order unconditionally energy stable scheme for a hydrodynamic Q-tensor model of liquid crystals. *Comput. Method. Appl. M.*, 318:803–825, 2017.

[42] J. Li, J. Zhao, and Q. Wang. Energy and entropy preserving numerical approximations of thermodynamically consistent crystal growth models. *J. Comput. Phys.*, 382:202–220, 2019.

[43] J. Zhao, X. Yang, Y. Gong, X. Zhao, X. Yang, J. Li, and Q. Wang. A general strategy for numerical approximations of non-equilibrium models—part I: thermodynamical systems. *Int. J. Numer. Anal. Model*, 15(6):884–918, 2018.

[44] J. Shen, J. Xu, and J. Yang. The scalar auxiliary variable (SAV) approach for gradient flows. *J. Comput. Phys.*, 353:407–416, 2018.

[45] J. Shen and J. Xu. Convergence and error analysis for the scalar auxiliary variable (SAV) schemes to gradient flows. *SIAM J. Numer. Anal.*, 56(5):2895–2912, 2018.

[46] Y. Zhao, J. Li, J. Zhao, and Q. Wang. A linear energy and entropy-production-rate preserving scheme for thermodynamically consistent crystal growth models. *Appl. Math. Lett.*, 98:142–148, 2019.

[47] Y. Gong, J. Zhao, and Q. Wang. Arbitrarily high-order unconditionally energy stable SAV schemes for gradient flow models. *Comput. Phys. Commun.*, 249:107033, 2020.

[48] E. Hairer, C. Lubich, and G. Wanner. *Geometric Numerical Integration: structure-preserving algorithms for ordinary differential equations*, volume 31. Springer, 2006.

[49] Q. Hong, J. Li, and Q. Wang. Supplementary Variable Method for Structure-Preserving Approximations to Partial Differential Equations with Deduced Equations. *Appl. Math. Lett.*, 110:106576, 2020.

[50] S. Sun, J. Li, J. Zhao, and Q. Wang. Structure-Preserving Numerical Approximations to Thermodynamically Consistent Non-isothermal Models of Binary Viscous Fluid Flows. *J. Sci. Comput.*, 83:50, 2020.

[51] Q. Cheng, C. Liu, and J. Shen. A new lagrange multiplier approach for gradient flows. *Comput. Method Appl. M.*, 367:113070, 2020.

[52] J.E. Marsden, G.W. Patrick, and S. Shkoller. Multisymplectic Geometry, Variational Integrators, and Nonlinear PDEs. *Commun. Math. Phys.*, 199:351–395, 1998.

[53] T. J. Bridges. Multi-symplectic structures and wave propagation. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 121, pages 147–190. Cambridge University Press, 1997.

[54] S. Reich. Multi-symplectic Runge-Kutta collocation methods for Hamiltonian wave equations. *J. Comput. Phys.*, 157:473–499, 2000.

[55] D. Furihata. Finite difference schemes for $\frac{\partial u}{\partial t} = (\frac{\partial}{\partial x})^{\alpha} \frac{\delta G}{\delta u}$ that inherit energy conservation or dissipation property. *J. Comput. Phys.*, 156:181–205, 1999.

[56] E. Celledoni, V. Grimm, R.I. McLachlan, D.I. McLaren, D. O'Neale, B. Owren, and G.R.W. Quispel. Preserving energy resp. dissipation in numerical PDEs using the "Average Vector Field" method. *J. Comput. Phys.*, 231:6770–6789, 2012.

[57] L. Brugnano, F. Iavernaro, and D. Trigiante. Hamiltonian boundary value methods (Energy preserving discrete line integral methods). *J. Numer. Anal. Ind. Appl. Math.*, 5:17–37, 2010.

[58] Y. Wang, B. Wang, and M. Qin. Local structure-preserving algorithms for partial differential equations. *Sci. China Ser. A: Mathematics*, 51:2115–2136, 2008.

[59] J. Cai, Y. Wang, and H. Liang. Local energy-preserving and momentum-preserving algorithms for coupled nonlinear schrödinger system. *J. Comput. Phys.*, 239:30–50, 2013.

[60] J. Cai and Y. Wang. Local structure-preserving algorithms for the "good" boussinesq equation. *J. Comput. Phys.*, 239:72–89, 2013.

[61] Y. Gong, J. Cai, , and Y. Wang. Some new structure-preserving algorithms for general multi-symplectic formulations of Hamiltonian PDEs. *J. Comput. Phys.*, 279:80–102, 2014.

[62] J. Cai, Y. Wang, and C. Jiang. Local structure-preserving algorithms for general multi-symplectic Hamiltonian PDEs. *Comput. Phys. Comm.*, 235:210–220, 2019.

[63] Z. Mu, Y. Gong, W. Cai, and Y. Wang. Efficient local energy dissipation preserving algorithms for the Cahn-Hilliard equation. *J. Comput. Phys.*, 374:654–667, 2018.

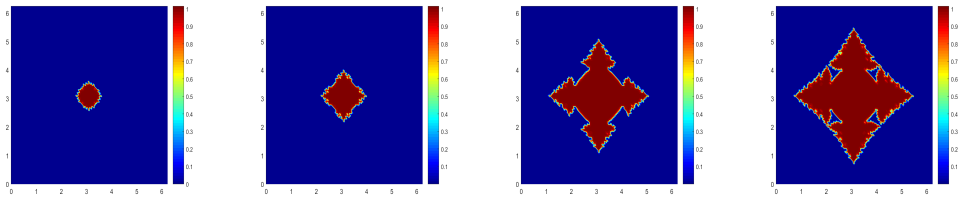
[64] L. Lu, Q. Wang, Y. Song, and Y. Wang. Local structure-preserving algorithms for the molecular beam epitaxy model with slope selection. *Discrete Cont. Dyn.- B*, 26:4745–4765, 2021.

8. Data availability statement

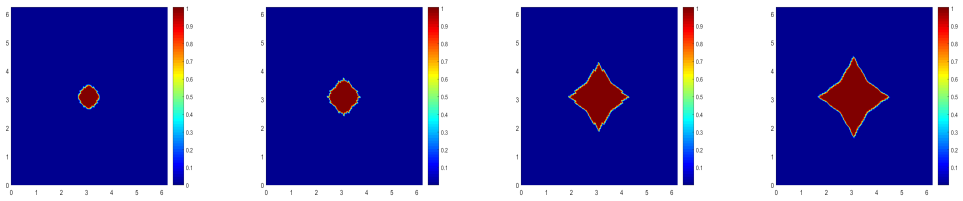
The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

9. Declarations

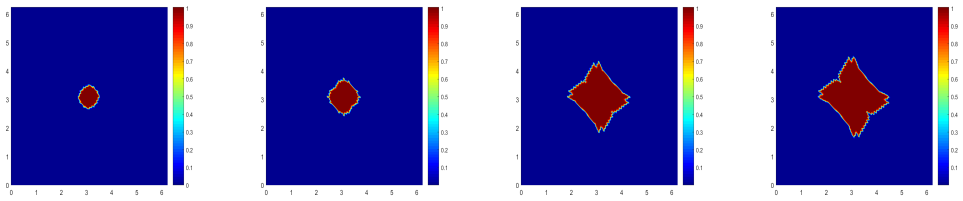
We declare that we have no financial and personal relationships with other people or organizations that can inappropriately influence our work, there is no professional or other personal interest of any nature or kind in any product, service and/or company that could be construed as influencing the position presented in, or the review of, the manuscript entitled "Local Energy Dissipation Rate Preserving Approximations to Driven Gradient Flows with Applications to Graphene Growth".



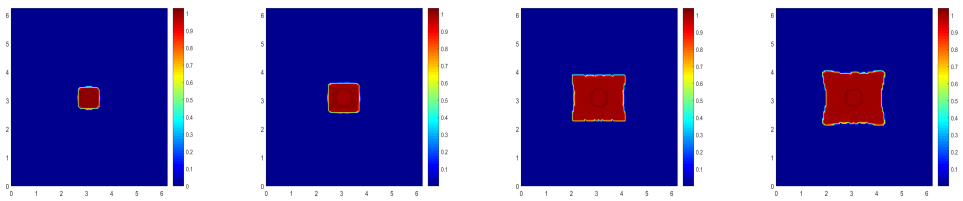
(a) $t = 0.01$ (b) $t = 0.03$ (c) $t = 0.08$ (d) $t = 0.1$



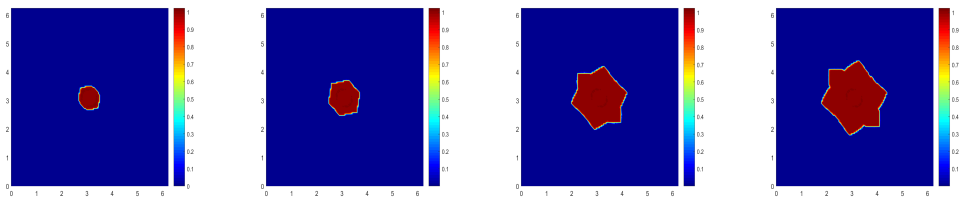
(e) $t = 0.01$ (f) $t = 0.03$ (g) $t = 0.08$ (h) $t = 0.1$



(i) $t = 0.01$ (j) $t = 0.03$ (k) $t = 0.08$ (l) $t = 0.1$

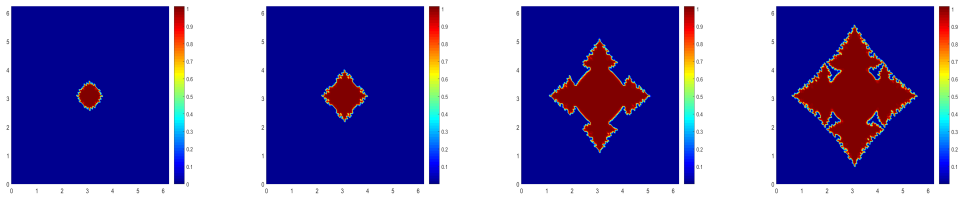


(m) $t = 0.01$ (n) $t = 0.03$ (o) $t = 0.08$ (p) $t = 0.1$

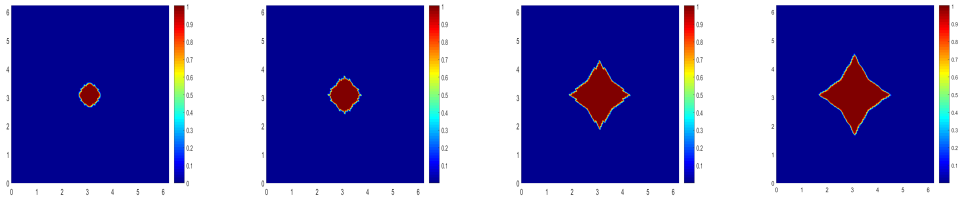


(q) $t = 0.01$ (r) $t = 0.03$ (s) $t = 0.08$ (t) $t = 0.1$

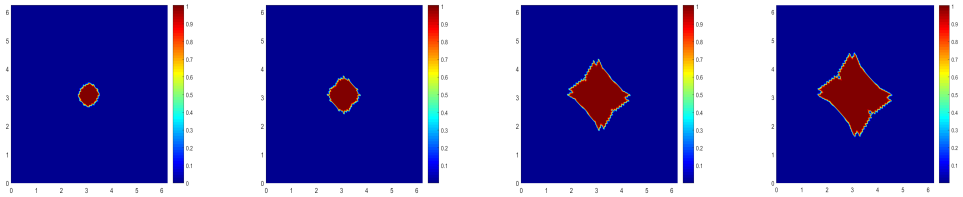
Figure 1: Row 1: Simulation of Cu (100) for *Graphene Algorithm 1*. Row 2: Cu (100) (2). Row 3: Cu (221). Row 4: Cu (110). Row 5: Cu (310).



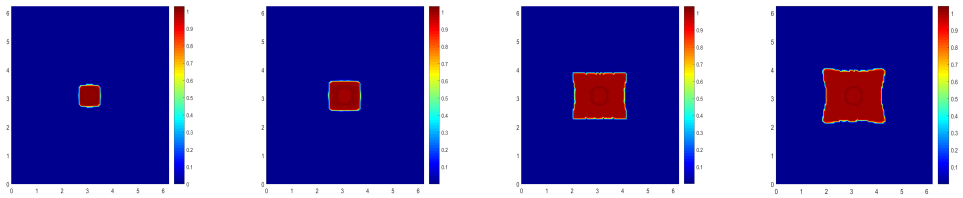
(a) $t = 0.01$ (b) $t = 0.03$ (c) $t = 0.08$ (d) $t = 0.1$



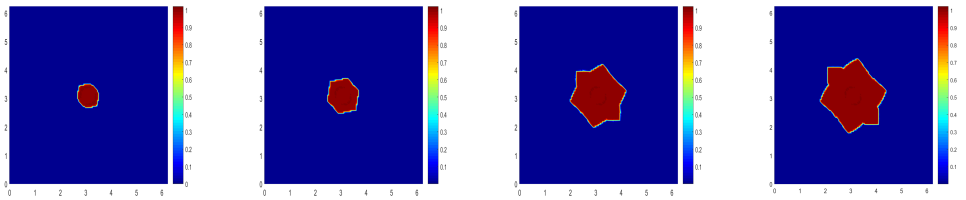
(e) $t = 0.01$ (f) $t = 0.03$ (g) $t = 0.08$ (h) $t = 0.1$



(i) $t = 0.01$ (j) $t = 0.03$ (k) $t = 0.08$ (l) $t = 0.1$

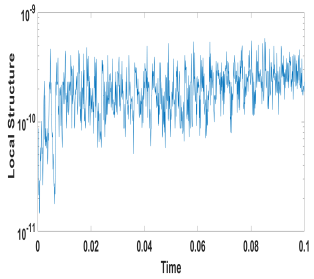


(m) $t = 0.01$ (n) $t = 0.03$ (o) $t = 0.08$ (p) $t = 0.1$

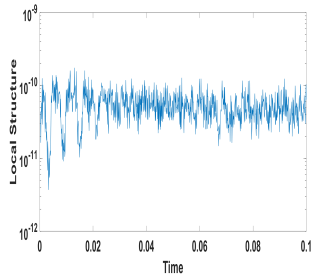


(q) $t = 0.01$ (r) $t = 0.03$ (s) $t = 0.08$ (t) $t = 0.1$

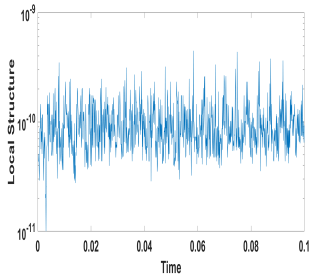
Figure 2: Row 1: Simulation of Cu (100) for *Graphene Algorithm 2*. Row 2: Cu (100) (2). Row 3: Cu (221). Row 4: Cu (110). Row 5: Cu (310).



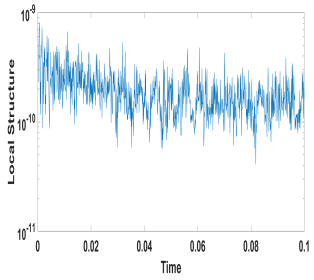
(a) Cu(100)



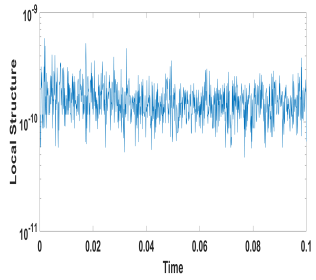
(b) Cu(100)(2)



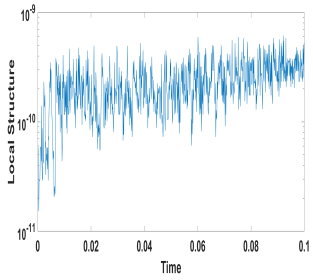
(c) Cu(221)



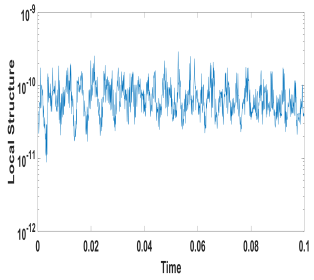
(d) Cu(110)



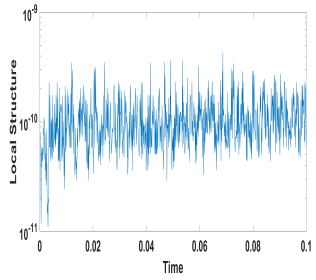
(e) Cu(310)



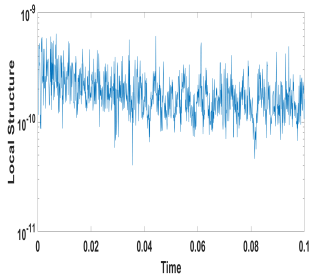
(f) Cu(100)



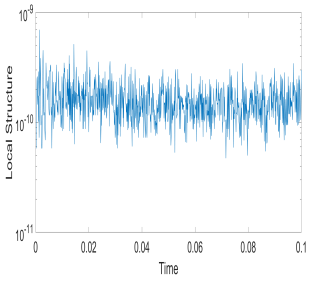
(g) Cu(100)(2)



(h) Cu(221)



(i) Cu(110)



(j) Cu(310)

Figure 3: (a)-(e):Local Structures for *Graphene Algorithm 1*, (f)-(j):Local Structures for *Graphene Algorithm 2*