Noname manuscript No.
(will be inserted by the editor)

A Level Set Method for the Dirichlet k-Partition Problem

Kwunlun Chu · Shingyu Leung

Received: date / Accepted: date

Abstract We propose a simple level set method for the Dirichlet k-partition problem which aims to partition an open domain into K different subdomains as to minimize the sum of the smallest eigenvalue of the Dirichlet Laplace operator in each subdomain. We first formulate the problem as a nested minimization problem of a functional of the level set function and the eigenfunction defined in each subdomain. As an approximation, we propose to simply replace the eigenfunction by the level set function so that the nested minimization can then be converted to a single minimization problem. We apply the standard gradient descent method so that the problem leads to a Hamilton-Jacobi type equation. Various numerical examples will be given to demonstrate the effectiveness of our proposed method.

Keywords Partial differential equations · Level set method · Numerical methods · Optimal partition · Spectral partition.

1 Introduction

The Dirichlet k-partition problem, or the spectral partition problem, aims to partition an open domain $U \subset \mathbb{R}^d$ into K different non-overlapping subdomains so to minimize the sum of the first eigenvalue of the Dirichlet-Laplace operator. Mathematically, we solve the following minimization problem

$$\min_{U_1,\cdots,U_k} \sum_{k=1}^{K} \lambda_1(U_k)$$

where $U = \bigcup_{k=1}^{K} U_k$ with $U_1, \cdots, U_K$ being pairwise disjoint and λ_1 is the first eigenvalue of the negative Laplace operator, $-\Delta$, with the zero Dirichlet boundary condition imposed on the boundary of each subdomain ∂U_k. This is a very nonlinear shape optimization problem since one has to determine the shape of some subdomains and then to find the corresponding eigenvalues ($\lambda \neq 0$) and eigenfunctions ($u \neq 0$) satisfying the partial differential equation $-\Delta u = \lambda u$ in U_k with the boundary condition $u = 0$ on ∂U_k.

This problem is interesting theoretically and is also important in many fields of applied sciences. Some theoretical studies on the existence of such a partition can be found in references such as [16,18, 17], while a weaker version of the existence statement has also been proved in [11,13]. The regularity of the partition has been studied in [25]. We refer interested readers to [24,12] for some surveys on the problem. In terms of applications and motivations of the problem, the partition problem can be found in various fields of science including those from studying the ground state energy of Bose-Einstein condensates [4,3], understanding the limiting case behavior of a chemical reaction system [19,20], modeling of the

Kwunlun Chu
Department of Mathematics, Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong.
E-mail: klchu@ust.hk

Shingyu Leung
Department of Mathematics, Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong.
Tel.: +852-23587414
Fax.: +852-23581643
E-mail: masyleung@ust.hk

self-organized structures formed by soap froths and liquid crystalline micelles [41,38], and biological segregation phenomena in population dynamics with diffusion and large interaction [15,16].

Numerically, there are different computational algorithms to search for the optimal partition for a given K. One of the earliest numerical studies has been developed in [22] which has formulated the problem as a minimization problem of a vector-valued function on U representing both the partition and the eigenfunction simultaneously. The resulting parabolic system is then discretized and is solved by the Gauss-Seidel method. A recent improvement of the method can be found in [9,10]. Another recent improvement has been developed in [39] which has incorporated with the so-called MBO method [29,30] to improve the overall computational efficiency using ideas from diffusion generated motions. A different class of numerical procedures has been proposed in [8] which relies on a Schrödinger operator relaxation. Due to the relaxation of the problem, the work can also investigate the asymptotic behavior of the partition as $K \to \infty$. A recent development based on the same relaxation has also been proposed in [7] for a related optimal partition problem which aims to minimize not the sum of the ground state eigenvalues but the p-norm of eigenvalues of the Dirichlet Laplacian. Another recent numerical procedure has been proposed in [6] based on a representation using density functions so that the overall algorithm can be done locally near the neighborhood of the region boundaries. Extended to non-Cartesian spaces, [23] has developed a finite element procedure to construct the Dirichlet k-partition for two-dimensional surfaces in three dimensions. In all numerical algorithms we mentioned here, one has to discretize the partition problem, solve the discretized system, and then try to observe the convergence to the solution of the continuum problem. A careful study on the convergence of this discretized graph partitioning problem to the continuum Dirichlet k-partition formulation can be found in [42,34]. A recent approach developed in [2] can also solve the partition on graphs.

In this work, we propose a level set method [33,36,32] for the Dirichlet k-partitions problem in arbitrary Lipschitz domains U. The level set method was originally developed to simulate the evolution of the interface by solving equations on a fixed Cartesian mesh rather than having an explicit parametrization of the moving surface. Because of the flexibility in modeling the topological change in the interface evolution, it has become a very popular tool in fields including multiphase flow simulation, shape optimization, computational geometry, and computer graphics. In many of the numerical approaches developed, the boundary of each partition evolves in iterations. Since there is no *a priori* information on the topology of these boundaries, surfaces might merge or split depending on the initial guess as they involve to lower the overall energy of the system. This makes the level set representation perfect for the representation of the partition because interfaces are represented implicitly. There is no need to worry about the connectivity of the discrete parametrization when there is any topological change in the partition. The level set method has also been applied to a similar spectral partition problem called bandgap maximization [27]. But it is a different shape optimization problem based on the eigenvalues of the Maxwell operator. We refer interested readers to the reference thereafter for more details on how the implicit representation can help in solving these optimal partition problems.

In this paper, we first introduce level set functions ϕ_k to implicitly represent the boundary of each partition and then we formulate the partition problem as a nested minimization problem of a functional consisting of the level set functions ϕ_k and the eigenfunctions u_k defined in each subdomain U_k. To improve the computational efficiency, we propose to relax the problem by approximating the eigenfunction u_k directly using the level set function ϕ_k. With this approximation, the nested minimization can be converted into a minimization problem of the level set functions which can be easily optimized using the standard gradient descent method. Although such approximation will not lead to the same minimizer as in the original problem in general, we are going to consider some simple configurations and provide some intuitions of the approximation. We will also demonstrate that the proposed approximation does give some reasonable minimizers corresponding to the original problem.

The paper is organized as follows. In Section 2, we will review the methods for computing k-partition problem proposed by [8,5,22,39]. Our proposed level set method for computing the Dirichlet k-partition problem will be proposed in Section 3. In Section 4, we will relate our proposed approach to the so-called variational level set method for multiphase flow simulation in [40] and will give a physical interpretation of our model. One crucial step in our approach is to approximate the eigenfunction within each subdomain by the corresponding level set function. We will discuss the error introduced in this step in Section 5. In Section 7, we give the details of the numerical implementation and present some numerical results. A conclusion will be given in Section 8.

2 Background

To express the Dirichlet k-partition problem more explicitly in terms of the first eigenfunction within each subdomain, we first assume that a subdomain U_k is known and, therefore, the first eigenvalue λ_1 can be easily found by solving

$$\lambda_1(U_k) = \min_{|u|^2_{H_0^1(U_k)}=1} \langle -\Delta u, u\rangle_{H_0^1(U_k)} = \min_{\substack{u\in H_0^1(U_k)\\ \|u\|_{L^2}=1}} \int_{U_k} |\nabla u|^2 \, d\mathbf{x}$$

where $\langle u, v\rangle_{H_0^1(U_k)} = \int_{U_k} uv \, d\mathbf{x}$. This implies that the Dirichlet k-partition problem can be expressed as a nested minimization problem by

$$\min_{U_1,\cdots,U_K} \sum_{k=1}^{K} \lambda_1(U_k) = \min_{U_1,\cdots,U_K} \sum_{k=1}^{K} \min_{\substack{u\in H_0^1(U_k)\\ \|u\|_{L^2}=1}} \int_{U_k} |\nabla u|^2 \, d\mathbf{x} \tag{1}$$

with $U = \bigcup_{k=1}^{K} U_k$ and $U_1, \cdots, U_K$ being pairwise disjoint. To solve this problem, [22] has proposed to introduce a mapping formulation to represent the eigenfunctions and the partition simultaneously using a vector-valued function $\mathbf{u}$. In particular, one defines the set of coordinate axes Σ_K by

$$\Sigma_K = \left\{\mathbf{u} = (u_1, \cdots, u_K) \in \mathbb{R}^K \text{ such that at most one of } u_j\text{'s is nonzero}\right\}.$$

Then, the Sobolev space associated with the set Σ_k, denoted by $H_0^1(U; \Sigma_K)$, is defined by

$$H_0^1(U, \Sigma_K) := \left\{\mathbf{u} = (u_1, \cdots, u_K) : u_k \in H_0^1(U) \text{ for all } k, \mathbf{u} \in \Sigma_K\right\} = H_0^1(U) \cap \Sigma_k$$

where $H_0^1(U) = W_0^{1,2}(U)$ is the standard Sobolev space with a compact support in U. For any $\mathbf{u} \in H_0^1(U, \Sigma_K)$, we have $\mathbf{u} \in \Sigma_k$ where at most one of $u_k(\mathbf{x})$ is non-zero for any $\mathbf{x}$. This indicates that the sets $U_k = u_k^{-1}(\mathbb{R}\backslash\{0\})$ are pairwise disjoint for $1 \le k \le K$. Therefore, each vector-valued function $\mathbf{u}$ in $H_0^1(U, \Sigma_K)$ corresponds to a possible configuration of K disjoint subsets of U.

With this mapping formulation, the Dirichlet k-partition problem is equivalent to the minimization problem given by

$$\min_{\mathbf{u}\in H_0^1(U,\Sigma_K)} \sum_{k=1}^{K} \int_U |\nabla u_k|^2 \, d\mathbf{x}$$

subject to $\int_U u_k^2 \, d\mathbf{x} = 1$ for each $k = 1, \cdots, K$. Numerically, on the other hand, it is more desirable to relax the constraint $\mathbf{u} \in \Sigma_k$ during the computation. [22] has also proposed to replace the condition $\mathbf{u} \in H_0^1(U, \mathbb{R}^k)$ by an extra penalty term, $\int_U \sum_{i\neq j} u_i^2(\mathbf{x}) u_j^2(\mathbf{x}) \, d\mathbf{x}$ which penalizes the overlapping of the sets $u_i^{-1}(\mathbb{R}\backslash\{0\})$ and $u_j^{-1}(\mathbb{R}\backslash\{0\})$ where $i \neq j$. For any small $\epsilon > 0$, the new formulation is

$$\min_{\mathbf{u}\in H_0^1(U,\mathbb{R}^K)} \sum_{k=1}^{K} \int_U |\nabla u_k|^2 \, d\mathbf{x} + \frac{1}{2\epsilon^2} \int_U \sum_{i\neq j} u_i^2 u_j^2 \, d\mathbf{x} \tag{2}$$

subject to $\int_U u_k^2 \, d\mathbf{x} = 1$ for each $k = 1, \cdots, K$. Standard gradient descent method then finally leads to the following system of parabolic partial differential equations

$$\partial_t u_k = \Delta u_k - \frac{1}{\epsilon^2} \left(\sum_{i\neq k} u_i^2 \right) u_k + \lambda_k u_k \tag{3}$$

for each $k = 1, \cdots, K$. Finally, to determine a steady-state solution of this coupled nonlinear system, [22] has developed the following three-step splitting scheme consisting of (**Step One**) a diffusion step, (**Step Two**) a reaction step and (**Step Three**) a normalization step. Starting with some initial condition $\mathbf{u}^0$, the algorithm iteratively updates the solution from $\mathbf{u}^{(n)}$ to $\mathbf{u}^{(n+1)}$ until it converges.

Step One: A diffusion step. One solves the heat equation $\partial_t u_k = \Delta u_k$ for a small timestep $t \in (0, \tau]$ and for $k = 1, \cdots, K$ with initial condition $u_k(\mathbf{x}, 0) = u_k^{(n)}(\mathbf{x})$. Denote the solution at time τ by $\tilde{u}_k(\mathbf{x}) = u_k(\mathbf{x}, \tau)$.

Step Two: A reaction step. The second operator in the equation is a reaction operator. One solves the system of ordinary differential equations given by $\partial_t u_k = -\frac{1}{\epsilon^2}\left(\sum_{i\neq k} u_i^2\right) u_k$ for $t \in (0,\tau]$ and for $k = 1,\cdots,K$ with the initial condition $u_k(\mathbf{x},0) = \tilde{u}_k(\mathbf{x})$. Denote the solution at time τ to be $\hat{u}_k(\mathbf{x}) = u_k(\mathbf{x},\tau)$. This step turns out to be the most challenging stage of the overall algorithm since it has to deal with a nonlinear system. It does not seem to be easy to determine an efficient numerical algorithm for the system. In particular, a simple semi-implicit scheme by linearizing the system might lead to a stiff system that will require a very small timestep in the ODE integrator.
A possible solution as suggested by [22] is to apply the following Gauss-Seidel idea. In particular, when seeking the solution $\hat{u}_k(\mathbf{x})$, the functions $\hat{u}_j(\mathbf{x})$ for $1 \leq j \leq k-1$ are already known. Therefore, we can replace the system by $\partial_t u_k = -\frac{1}{\epsilon^2}\left(\sum_{i<k} \hat{u}_i^2 + \sum_{i>k} \tilde{u}_i^2\right) u_k$ with the initial condition $u_k(\mathbf{x},0) = \tilde{u}_k(\mathbf{x})$. The exact solution for this ODE is known and is given by

$$\hat{u}_k(\mathbf{x}) = u_k(\mathbf{x},\tau) = u_k(\mathbf{x},0)\exp\left[-\frac{\tau}{\epsilon^2}\left(\sum_{i<k}\hat{u}_i^2 + \sum_{i>k}\tilde{u}_i^2\right)\right].$$

Step Three: A normalization step. This normalization step takes care of the last term in the original nonlinear equation corresponding to the integral constraint $\int_U u_k^2\, d\mathbf{x} = 1$. Instead of determining the Lagrangian multiplier explicitly in each iteration, the method projects the intermediate solution back to the required space by $\hat{u}_k$ by rescaling each component of the mapping function $u_k^{(n+1)}(\mathbf{x}) = \hat{u}_k(\mathbf{x})/|\hat{u}_k(\mathbf{x})|$ for $t \in (0,\tau]$ and for $k = 1,\cdots,K$.
Step Four: Check for the convergence. There are various possible strategies to determine if the algorithm convergence. One approach is to check if the sets U_k converges in the iterations. Since the nonzero element in $\mathbf{u}$ classifies the location into one of the subdomains, one can defines

$$\sum_{k=1}^{K} \left| \mathbb{1}_{\{u^{(n+1)}>0\}} - \mathbb{1}_{\{u^{(n)}>0\}} \right|$$

where $\mathbb{1}_\Omega$ is the characteristic function so that $\mathbb{1}_\Omega(\mathbf{x})$ equals to 1 if $\mathbf{x} \in \Omega$ and 0 otherwise. If this quantity remains unchanged in the iterations, the partition converges and the algorithm stops.

Step Two has a flavor of the phase-field method [1,14,21] for modeling dynamic interface for multiphase flow where a similar Lagrange multiplier ϵ controls the *width* of the interface. In the limit of ϵ approaches to zero, the phase-field model reduces to a sharp interface model as in the level set method. As $\epsilon \to 0$, the nonlinear ODE system becomes stiffer while the dynamic takes the solution $\mathbf{u}$ back to the constraint set Σ_K at a faster pace. Motivated by the idea in the diffusion generated motions or the so-called MBO scheme [29,30], a recent work [39] has proposed to replace **Step Two** by a simple thresholding step

$$\hat{u}_k(\mathbf{x}) = \begin{cases} \tilde{u}_k(\mathbf{x}) \text{ if } \tilde{u}_k(\mathbf{x}) = \max\limits_{1\leq j\leq K} \tilde{u}_j(\mathbf{x}) \\ 0 \qquad\qquad \text{otherwise.} \end{cases}$$

This step assumes $\epsilon \to 0$ and simply projects the solution $\mathbf{u}$ back to the constraint set Σ_K. This thresholding step significantly reduces the amount of computational time since it does not require to solve any differential equation. Because of overall algorithm is simple, it is computationally very efficient and cheap which makes 3D computations possible.

To have an overall algorithm involves only explicit operations, [22,39] solve the diffusion equation in **Step One** using Fast Fourier Transform (FFT). This, however, works only if the domain is rectangular and requires the periodic boundary condition on ∂U. For general Lipschitz domains, it is not easy how to replace the FFT by some simple efficient numerical solvers. Moreover, in all discussions above, one considers only the periodic boundary condition. It might not be straightforward to extend these methods to handle the Dirichlet k-partition problem on a bounded domain where a set of partition constraints on ∂U are already imposed. Finally, although [39] has also extended the method to a unit sphere by solving the diffusion equation using the Spherical Harmonic Transform (SHT), the method has difficulty solving the Dirichlet k-partition problem on general manifolds.

3 Our Proposed Method

In this section, we will give the details of our proposed method for the Dirichlet k-partition problem on a domain with Lipschitz boundary based on the level set method [33,36,32]. Following [22,39], we start with the nested minimization problem based on the subdomain U_k and the associating first eigenfunction u_k. Our proposed approach first introduces a level set function ϕ_k for each subdomain U_k. In the case when the level set functions are signed distance functions, we noticed that they give a good approximation to their associating first eigenfunctions u_k's. Therefore, as an approximation, we propose to replace the eigenfunction u_k by simply the level set function ϕ_k. This effectively converts the nested minimization to a single minimization problem.

3.1 A Level-Set Formulation for the Dirichlet k-partition Problem

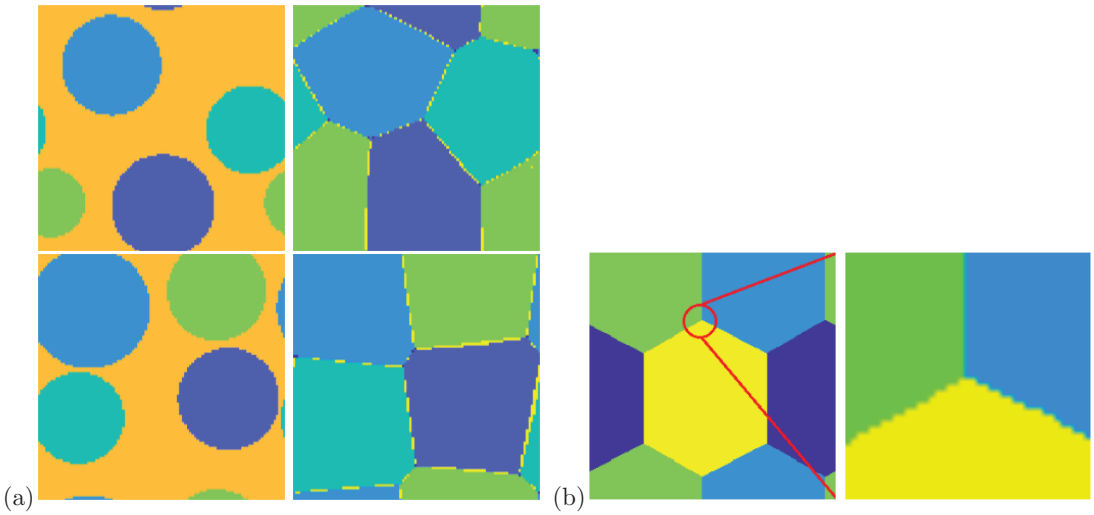


Fig. 1: (Section 3.1) Solutions to the k-partition problem using different initial conditions based on the formulation (5) and (6). (a) The first column is the initial condition. The second column is the corresponding steady-state solution. steady-state solution might not give the global minimum. (b) The correct global minimum gives 120-degree triple junction structures.

Following the previous discussion, we express the smallest eigenvalue λ_1 of the negative Laplace operator, $-\Delta$, in the subdomain U_k with zero Dirichlet boundary as the minimization problem given by

$$\lambda_1(U_k) = \min_{\substack{u\in H_0^1(U_k)\\ \|u\|_{L^2(U_k)}=1}} \int_{U_k} |\nabla u|^2 \, d\mathbf{x} = \min_{\substack{u\in H_0^1(U_k)\\ \|u\|_{L^2(U_k)}\neq 0}} \frac{\displaystyle\int_{U_k} |\nabla u|^2 \, d\mathbf{x}}{\displaystyle\int_{U_k} u^2 \, d\mathbf{x}} \, .$$

Our first step is to introduce the level set method [33,36,32], and rewrite the formulation using the level set function. Mathematically, for a given open set $U_k \in \mathbb{R}^d$ with smooth boundary ∂U_k, we define a Lipschitz level set function ϕ_k satisfying

$$\begin{cases} \phi(\mathbf{x}) > 0 & \text{for} \quad \mathbf{x} \in U_k \\ \phi(\mathbf{x}) = 0 & \text{for} \quad \mathbf{x} \in \partial U_k \\ \phi(\mathbf{x}) < 0 & \text{for} \quad \mathbf{x} \in \bar{U}_k^c \, . \end{cases}$$

Then we could re-expressed the smallest eigenvalue λ_1 by

$$\lambda_1(\phi_k) = \min_{\substack{u\in H_0^1(U_k)\\ \|u\|_{L^2(U_k)}\neq 0}} \frac{\displaystyle\int |\nabla u|^2 H(\phi_k) \, d\mathbf{x}}{\displaystyle\int u^2 H(\phi_k) \, d\mathbf{x}} \tag{4}$$

where $H(\cdot)$ is the Heaviside function so that it gives 1 if the argument is positive and 0 otherwise. Similar to the original formulation, the Dirichlet k-partition problem can be expressed as a nested minimization problem given by

$$\min_{\phi_1,\cdots,\phi_K} \sum_{k=1}^{K} \lambda_1(\phi_k) = \min_{\phi_1,\cdots,\phi_K} \sum_{k=1}^{K} \min_{\substack{u\in H_0^1(U_k)\\ \|u\|_{L^2(U_k)}\neq 0}} \frac{\int |\nabla u|^2 H(\phi_k)\, d\mathbf{x}}{\int u^2 H(\phi_k)\, d\mathbf{x}}$$

subjected to the non-overlapping constraint $\sum_{k=1}^{K} H(\phi_k) = 1$ for each $\mathbf{x}$ in the domain. Once we have the optimizer to this minimization problem, the final k-partition can be determined from the level set functions given by $U_k = \{\mathbf{x} : \phi_k > 0\}$. This nested minimization problem, unfortunately, is not easy to be solved. One has to vary both u and ϕ_k together to determine an optimizer of the above functional. One possible approach is to introduce an alternative minimization strategy to perturb each of them in turn. In particular, we might define a functional

$$E[u, \phi_1, \cdots, \phi_K] = \sum_{k=1}^{K} \left[\frac{\int |\nabla u|^2 H(\phi_k)\, d\mathbf{x}}{\int u^2 H(\phi_k)\, d\mathbf{x}} \right] = \sum_{k=1}^{K} \left[\frac{\int_{U_k} |\nabla u|^2\, d\mathbf{x}}{\int_{U_k} u^2\, d\mathbf{x}} \right]$$

and minimize it over u and ϕ_k's alternatively. Minimizing over u with fixed ϕ_k's, we obtain the Euler-Lagrange equation

$$-\Delta u - u \left[\frac{\int_{U_k} |\nabla u|^2\, d\mathbf{x}}{\int_{U_k} u^2\, d\mathbf{x}} \right] = 0 \text{ inside each } U_k\,. \tag{5}$$

The solution to this equation is simply the eigenfunctions of the Laplace operator over each region U_k. In the next iteration, one minimizes each ϕ_k's with a fixed u which gives

$$-\delta(\phi_k) \left[|\nabla u|^2 - u^2 \left(\int_{U_k} |\nabla u|^2\, d\mathbf{x} \right) \right] = 0$$

with a Lagrange multiplier $\mu(x)$ chosen to enforce the constraint $\sum H(\phi_k) = 1$. Then the gradient descent method with time rescaling would yield

$$\frac{\partial \phi_k}{\partial t} = \left[|\nabla u|^2 - u^2 \left(\int_{U_k} |\nabla u|^2\, d\mathbf{x} \right) + \mu(x) \right]. \tag{6}$$

We have implemented this alternating minimization approach but we have noticed that the corresponding steady-state solution of the gradient descent depends heavily on the initial condition and might easily get stuck at a local minimizer. Figure 1 shows the steady-state solution for solving (5) and (6) with two different initial conditions. As we can see, none of both solutions agrees with the correct hexagonal structure (with 120-degree triple junctions). Intuitively, each of the regions will expand during the evolution with the rate given by $|\nabla u|^2$ (as $u \approx 0$ around the zero level set of ϕ_k) until the vacuum region is filled up by a certain phase of the partition. Once a vacuum region is eliminated, the motion will stop immediately and the boundary segments will not further to re-adjust themselves to give the correct angle configuration. This motivates us to incorporate the diffusion term in equation (5) to the motion-governing equation (6).

We observed that the level set function ϕ_k itself is in $H_0^1(U_k)$ and $|\phi_k|_{L^2(U_k)} \neq 0$ when the set U_k is non-empty. Therefore, we propose to replace the eigenfunction u in the minimization problem directly by the level set function ϕ. Indeed, replacing the eigenfunction u_k by the corresponding level set function ϕ_k clearly would not lead to the same k-partition. We are not claiming that this approximation will give the same solution to the original problem. However, as we will demonstrate in Section 5, our approach will give a reasonable match partition for general cases and will provide the same optimizer under some simple configuration of the final partition.

If we define a functional $\lambda_1' : H^1(\Omega) \to \mathbb{R}$ by

$$\lambda_1'(\phi_k) := \frac{\int_{\Omega} |\nabla \phi_k|^2 H(\phi_k)\, d\mathbf{x}}{\int_{\Omega} \phi_k^2 H(\phi_k)\, d\mathbf{x}}$$

which approximates the smallest eigenvalue $\lambda_1(\phi_k)$ defined in equation (4), the Dirichlet k-partition can be rewritten as a single minimization problem as

$$\min_{\phi_1,\cdots,\phi_K} \sum_{k=1}^{K} \lambda_1'(\phi_k) = \min_{\phi_1,\cdots,\phi_K} \sum_{k=1}^{K} \left[\frac{\int |\nabla \phi_k|^2 H(\phi_k)\, d\mathbf{x}}{\int \phi_k^2 H(\phi_k)\, d\mathbf{x}} \right]$$

subjected to the non-overlapping constraint $\sum_{k=1}^{K} H(\phi_k) = 1$. To replace this constrained optimization problem by an unconstrained problem, we rewrite the constraint as

$$\frac{1}{2} \int \left(\sum_{k=1}^{K} H(\phi_k) - 1 \right)^2 d\mathbf{x} = 0$$

and incorporate it into the optimization problem as an extra penalty term. This finally leads to

$$\min_{\phi_1,\cdots,\phi_K} \sum_{k=1}^{K} \lambda_1'(\phi_k) + \frac{\mu}{2} \int \left(\sum_{k=1}^{K} H(\phi_k) - 1 \right)^2 d\mathbf{x}$$

for some constant $\mu > 0$ to be determined according to the mesh size. Intuitively, since this term has to balance the effect from the first term which involves the normed-square of the level set gradient, we recommend choosing $\mu = O\left(\Delta x^{-2}\right)$.

3.2 The Gradient of the Functional

To minimize the functional, we compute the Fréchet derivative of the functional with respect to the level set function ϕ. We first concentrate on the derivative of the approximated first eigenfunction. To simplify the notation, we introduce $F[\phi] = \int |\nabla \phi|^2 H(\phi)\, d\mathbf{x}$ and $G[\phi] = \int \phi^2 H(\phi)\, d\mathbf{x}$. Since the Fréchet derivative of F and G with respect to ϕ are given by $\frac{\delta F}{\delta \phi} = -2H(\phi)\Delta\phi - \delta(\phi)|\nabla\phi|^2$ and $\frac{\delta G}{\delta \phi} = 2\phi H(\phi) + \phi^2 \delta(\phi)$, we apply the quotient rule and obtain

$$\begin{aligned} \frac{\delta \lambda_1'}{\delta \phi} &= \frac{1}{G[\phi]^2} \left[\frac{\delta F}{\delta \phi} G - \frac{\delta G}{\delta \phi} F \right] = \frac{1}{G[\phi]} \left[\frac{\delta F}{\delta \phi} - \frac{\delta G}{\delta \phi} \lambda_1' \right] \\ &= - \left[\int \phi^2 H(\phi)\, d\mathbf{x} \right]^{-1} \left[2H(\phi) \left(\Delta\phi + \phi \lambda_1' \right) + \delta(\phi) \left(|\nabla \phi|^2 + \phi^2 \lambda_1' \right) \right]. \end{aligned}$$

Now, if we have $\phi\delta(\phi) = 0$, we can drop the last term in the expression which leads to

$$\frac{\delta \lambda_1'}{\delta \phi} = - \left[\int \phi^2 H(\phi)\, d\mathbf{x} \right]^{-1} \left[2H(\phi) \left(\Delta\phi + \phi \lambda_1' \right) + \delta(\phi) |\nabla \phi|^2 \right].$$

Together with the penalty term for the non-overlapping constraint, we have the following gradient descent equation for minimizing λ_1'

$$\frac{\partial \phi_k}{\partial t} = \gamma_k \left[2H(\phi_k)(\Delta\phi_k + \phi_k \lambda_1') + \delta(\phi_k) |\nabla \phi_k|^2 \right] - \mu \delta(\phi_k) \left[\sum_{k=1}^{K} H(\phi_k) - 1 \right].$$

where $\gamma_k = \left[\int \phi_k^2 H(\phi_k)\, d\mathbf{x} \right]^{-1}$.

Since we are are only interested in the evolution of the zero level set, we follow similar approaches in the level set community and make the following modifications to simplify the computations. First, we introduce the regularized Heaviside function $H_\alpha(x)$ and regularized Dirac's delta function $H_\alpha(\cdot)$, $\delta_\alpha(\cdot) : \mathbb{R} \to \mathbb{R}$ by

$$H_\alpha(x) := \begin{cases} 0 & \text{if } x \le -\alpha \\ \frac{1}{2} + \frac{x}{2\alpha} + \frac{1}{2\pi} \sin\left(\frac{\pi x}{\alpha} \right) & \text{if } -\alpha < x < \alpha \\ 1 & \text{if } x \ge \alpha \end{cases}$$

and $\delta_\alpha(x) := H_\alpha'(x)$. The first term $2H_\alpha(\phi_k)$ can be approximated by one as $2H_\alpha(0) = 1$, and hence we can replace the coefficient of $\Delta\phi_k$ by $\delta_\alpha(\phi_k)/\delta_\alpha(0)$. The second term is removed from the equation since

$\phi_k \approx 0$ near the zero level set. If we project the level set function in the evolution so that it is a signed distance function, we have $|\nabla\phi_k| = 1$ and so the third term on the right hand side contains only $\delta(\phi_k)$. Now, we have

$$\frac{\partial\phi_k}{\partial t} = \delta_\alpha(\phi_k)\left[\frac{\gamma_k}{\delta_\alpha(0)}\Delta\phi_k + \gamma_k - \mu\left(\sum_{k=1}^{K} H_\alpha(\phi_k) - 1\right)\right] \tag{7}$$

where $\gamma_k = \left[\int \phi_k^2 H_\alpha(\phi_k)\, d\mathbf{x}\right]^{-1}$ for some small $\alpha > 0$. Finally, we rescale the time so that the term $\delta_\alpha(\phi_k)$ is replaced by $|\nabla\phi_k|$ and it leads to

$$\frac{\partial\phi_k}{\partial t} + v_n|\nabla\phi_k| = \frac{\gamma_k}{\delta_\alpha(0)}|\nabla\phi_k|\Delta\phi_k$$

with $v_n = \mu\left[\sum_{k=1}^{K} H_\alpha(\phi_k) - 1\right] - \gamma_k$ and $\gamma_k = \left[\int \phi_k^2 H_\alpha(\phi_k)\, d\mathbf{x}\right]^{-1}$. The left-hand side of the equation is now a standard first order nonlinear Hamilton-Jacobi equation as in the typical level set method. It models the normal motion of the zero level set of ϕ_k with the given normal velocity v_n. The right hand side of the equation is a diffusion term. When the level set function is close to a signed distance function, this term corresponds to the motion by mean curvature so that the zero level set is regularized in the evolution.

To initialize the evolution, we initialize $\phi_k(t = 0)$ by assigning K random non-overlapping circles. There are multiple ways to check this condition. One is to use the level set representation by ensuring that $\sum_{k=0}^{K} H(\phi_k(\mathbf{x})) \leq 1$ at each point $\mathbf{x} \in U$. If the quantity is larger than 1, it means there exists a point $\mathbf{x}$ lies inside of more than one zero level set. This implies that at least two-level set functions intersect at that point. Another approach is to check explicitly that $|\mathbf{x} - \mathbf{x}_k| < r_k$ for at most 1 such k for k from 1 to K where $\mathbf{x}_k$ and r_k are the center and the radius of the initial disk.

3.3 A Dirichlet k-partition Problem on an Arbitrary Domain

In this section, we generalize the formulation above and discuss a related Dirichlet k-partition problem. In all the above discussion, we have assumed that the domain U is rectangular where the periodic boundary condition is imposed on ∂U for simplicity. Instead, we now consider the case when the domain U is arbitrary and, therefore, in general, we cannot impose the periodic boundary condition on ∂U. Since it is not straight-forward how to apply the FFT on such domains, typical numerical approaches might not be easy to solve such a generalized Dirichlet k-partition problem. On the other hand, it is rather simple in our level set formulation. The only modification is on the penalty term where we impose the non-overlapping constraint.

Assume that the boundary ∂U is represented by a level set function ψ so that $\partial U = \psi^{-1}(0)$. We can simply modify the penalty term to

$$\frac{\mu}{2}\int\left[\sum_{k=1}^{K} H(\phi_k) + H(\psi) - 1\right]^2 d\mathbf{x}$$

which leads to the following new normal velocity in the Hamilton-Jacobi equation

$$v_n = \mu\left[\sum_{k=1}^{K} H_\alpha(\phi_k) + H_\alpha(\psi) - 1\right] - \gamma_k\,.$$

The initial condition for the iteration is similar to the previous case. We only need to check with one more condition on ψ. The rest follows.

3.4 The Level Set Regularization

In the previous derivation, we have assumed that the level set function $\phi(\mathbf{x})$ is a signed distance function such that $|\nabla\phi| = 1$ for all intermediate iterations. This property, however, cannot be satisfied automatically in the gradient flow. This implies that ϕ may develop steep and flat gradients at or near the zero level set as it evolves, making the computed location of the boundary ∂U_k and further computations inaccurate. Therefore, we use reinitialization to regularize the level set evolution. To restore the distance

property for the level set functions, the usual way is to make ϕ a signed distance function without moving its zero level set appreciably. This can be done by the so-called reinitialization. There are also many well-developed numerical methods to efficiently obtain their numerical solutions. We refer all interested readers to [32] and thereafter for a more detailed discussion. Following the typical level set approach [32], we reinitialize the level set functions by introducing an artificial time τ and solving the following reinitialization equation

$$\phi_\tau + \text{sign}(\phi^n)(|\nabla\phi| - 1) = 0$$

with the initial condition $\phi(\tau = 0)$ being the intermediate state given by ϕ^n at the n-th iteration step. There are standard numerical implementations and packages available for solving this first-order nonlinear hyperbolic Hamilton-Jacobi equation. For example, one can apply the TVDRK3 [37] in the time direction and the WENO5 [26] for the spatial derivative. This could give a high order accurate solution to the τ-direction evolution. The CFL condition for solving this equation is given by $\Delta\tau = O(\Delta x)$. If only low order solution is needed, one may simply replace the TVDRK3 by a forward Euler step and the WENO5 by the simple forward/backward differencing. In practice, however, since we are interested in the solution only near the zero level set, there is no need to obtain a steady-state solution. Instead, we simply solve the system for several $\Delta\tau$ steps, and the intermediate solution ϕ is used to replace the original level set function.

3.5 The Stability Condition

When solving the evolution equation governing the motion of the zero level set using an explicit scheme, we need to have a stable numerical scheme by constraining the size of the timestep. The equation consists a diffusion term $\Delta\phi$ with the diffusion coefficient given by $\gamma_k|\nabla\phi_k|/\delta_\alpha(0)$ and also a nonlinear hyperbolic part $|\nabla\phi|$ with the normal velocity $v_n = \mu\left[\sum_{k=1}^K H_\alpha(\phi_k) - 1\right] - \gamma_k$.

If we take $\alpha = \Delta x$, the diffusion coefficient $\gamma_k|\nabla\phi_k|/\delta_\alpha(0)$ is roughly $\gamma_k\Delta x$. This implies that the stability condition for the diffusion part is given by $\Delta t \le \Delta x^2/(2d\Delta x \max_k \gamma_k) = \Delta x/(2d\max_k \gamma_k)$, where d is the dimension. Since both γ_k and μ are both positive, the magnitude of the convection coefficient v_n is bounded by $\max_k \gamma_k^n + \mu$. This gives the stability condition $\Delta t \le \Delta x/(\max_k \gamma_k + \mu)$. Combining these two constraints, we have the time step requirement for Δt by

$$\Delta t \le \min\left(\frac{\Delta x}{2d\max_k \gamma_k}, \frac{\Delta x}{\max_k \gamma_k + \mu}\right) = \frac{\Delta x}{\max_k \gamma_k + \max\left[(2d-1)\max_k \gamma_k, \mu\right]}.$$

Since the parameter μ is of order $O\left(\Delta x^{-2}\right)$ as discussed in Section 3.1, the overall stability condition of the time dependent equation is in general given by $\Delta t = O\left(\Delta x^3\right)$.

As we will see in Section 7, the initial state will be set by disjoint circles. These circles will expand to fill the vacuum region. In general, a larger region yields a smaller γ_k which makes the above stability condition less restrictive. Hence, in the numerical experiments, we check the timestep requirement only in the initial state.

3.6 A Stopping Criterion

Finally, we provide a convergence criterion for the iterations. In this work, we define the following relative change to measure the convergence of the partition based on the level set function,

$$E = \frac{\sum_{k=1}^K \sum_{i,j} \left|\phi_{k,ij}^{n+1} - \phi_{k,ij}^n\right| \mathbb{1}_{\left\{|\phi_{k,ij}^n| < \Delta x\right\}}}{\sum_{k=1}^K \sum_{i,j} \mathbb{1}_{\left\{|\phi_{k,ij}^n| < \Delta x\right\}}}$$

where $\mathbb{1}_\Omega$ is the characteristic function of the set Ω so that it gives 1 if $\mathbf{x} \in \Omega$ and 0 otherwise. Since the partition is given by only the location of the zero level set, we can simply consider the change of the level set function within a small neighborhood of the zero level set. This explains why we have $\mathbb{1}_{\left\{|\phi_{k,ij}^n| < \Delta x\right\}}$.

4 Relationships to the Variational Level Set Method for Multiphase Flow Simulations

In this section, we are going to compare our method with the variational level set method [40] originally developed for multiphase flow simulation and to show a physical interpretation of our model. Consider the following multiphase flow model which tries to capture the motion of a material interface (such as the solid-liquid, grain, or antiphase boundaries) by associating an energy functional involving the surface tension (surface area) and the bulk energy (volume of each sub-region). When the initial state is not in the energy equilibrium, interfaces will evolve to reduce the energy function.

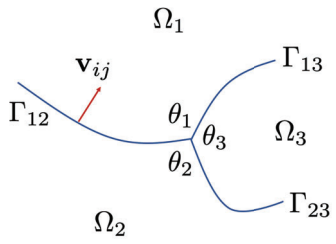


Fig. 2: The setup for a three-phase flow problem as discussed in [40].

Mathematically, an n-phase flow problem can be formulated as follows. Let Ω_i for $1 \le i \le n$ be a set of disjoint regions in $\mathbb{R}^2$. The energy functional can be defined as $E = E_1 + E_2$ with

$$E_1 = \sum_{1 \le i < j \le n} f_{ij} \text{Length}(\Gamma_{ij}) \quad \text{and} \quad E_2 = \sum_{i=1}^{n} e_i \text{Area}(\Omega_i)$$

where f_{ij} is the coefficient to the surface energy and e_i is the coefficient for the bulk energies for region Ω_i. A typical setup for a three-phase flow problem is shown in Figure 2. When the level set method is used [40], we introduce ϕ_i for $1 \le i \le n$ to represent Ω_i. One can then express the energy functional using $E = E_1 + E_2$ with

$$E_1 = \sum_{i=1}^{n} \gamma_i \int \delta(\phi_i) |\nabla \phi_i| \, d\mathbf{x} \quad \text{and} \quad E_2 = \sum_{i=1}^{n} e_i \int H(\phi_i) \, d\mathbf{x}$$

where γ_i's are coefficients satisfying $f_{ij} = \gamma_i + \gamma_j$. Hence, the problem becomes to minimize E subject to the constraint $\frac{1}{2} \int \left(\sum_{i=1}^{n} H(\phi_i) - 1 \right)^2 d\mathbf{x} = 0$. Introducing a regularization parameter $\mu > 0$ and replacing the problem with an unconstrained optimization problem, one obtains

$$E[\phi_1, \cdots, \phi_n, \mu] = E_1 + E_2 + \frac{\mu}{2} \int \left[\sum_{i=1}^{n} H(\phi_i) - 1 \right]^2 d\mathbf{x} \, .$$

The minimizer of this functional can be determined by the gradient descent method. We introduce an artificial time t and solve the following evolution equation for the steady-state solution

$$\frac{\partial \phi_i}{\partial t} = |\nabla \phi_i| \left[\gamma_i \nabla \cdot \left(\frac{\nabla \phi_i}{|\nabla \phi_i|} \right) - e_i - \mu \left(\sum_{i=1}^{n} H(\phi_i) - 1 \right) \right] ,$$

with the boundary conditions $\frac{\delta(\phi_i)}{|\nabla \phi_i|} \frac{\partial \phi_i}{\partial \mathbf{n}} = 0$ for $1 \le i \le n$. Finally, if we project the level set function so that it is a signed distance function in the evolution, we have

$$\frac{\partial \phi_i}{\partial t} = \delta(\phi_i) \left[\gamma_i \Delta \phi_i - e_i - \mu \left(\sum_{i=1}^{n} H(\phi_i) - 1 \right) \right] .$$

We have observed that this equation has the same form as our evolution equation (7) with

$$\gamma_i \Longleftrightarrow \frac{\gamma_k}{\delta_\alpha(0)} \quad \text{and} \quad e_i \Longleftrightarrow -\gamma_k \, .$$

Because of this similarity, we can provide a physical understanding of our model. Following the physical interpretation as in the multiphase flow simulation, our proposed method aims to maximize the area of each subdomain with the *bulk energy coefficient* γ_k, at the same time tries to minimize the arc length of the boundary with the coefficient $\gamma_k/\delta_\alpha(0)$ relating to the *surface energy*.

5 Approximating the Eigenfunctions by the Level Set Function

The crucial step of our proposed algorithm is to replace the first eigenfunction within the subdomain U_k by the associated level set function ϕ_k. To motivate the discussion, we first consider the one-dimensional case and then will provide a slightly more general analysis in the two-dimensional case. We are going to demonstrate that such approximation will lead to the same partition in the first case, while our approximation λ_1' provides a reasonably well estimate to λ_1 for some regular partitions.

For the one dimensional case, we first consider an interval $U = [0, L]$ where the first eigenvalue and the corresponding eigenfunction are given by $\lambda_1 = \pi^2/L^2$ and $\sin(\pi x/L)$, respectively. The signed distance function representation of the level set function is given by $\phi = \min(x, L-x)$ while the approximate $\lambda_1'(\phi)$ over the domain U is given by $\int_U |\nabla\phi|^2 H(\phi)\,dx / \int_U \phi^2 H(\phi)\,dx = 12/L^2 = (12/\pi^2)\lambda_1$. Now, considering the k-partition on the one dimensional case where the whole domain $U = [0, 1]$ and we would like to segment it into K subintervals. For each subinterval, we have $\lambda_1(U_k) = \pi^2/L_k$ where L_k is the length of the interval U_k. Therefore, the overall problem to solve is given by

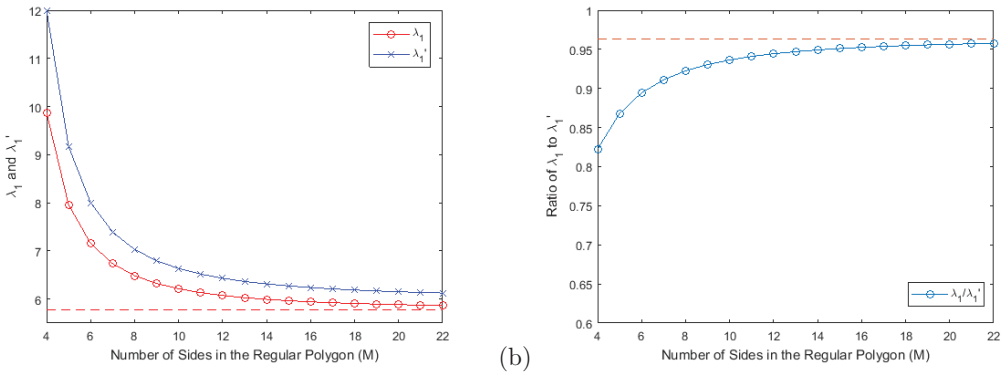
$$\min_{U_1,\cdots,U_K} \sum_{k=1}^{K} \lambda_1(U_k) = \pi^2 \min_{U_1,\cdots,U_K} \sum_{k=1}^{K} \frac{1}{L_k^2}\,.$$

For the approximated problem, we have

$$\min_{U_1,\cdots,U_K} \sum_{k=1}^{K} \lambda_1'(\phi_k) = 12 \min_{U_1,\cdots,U_K} \sum_{k=1}^{K} \frac{1}{L_k^2} = \frac{12}{\pi^2}\left[\min_{U_1,\cdots,U_K} \sum_{k=1}^{K} \lambda_1(U_k)\right]\,.$$

Although the minimum value of the sum of the first eigenvalues over-estimates the actual minimum, we notice that both minimization problems essentially lead to the same *minimizer* given by the uniform partition where $L_k = 1/K$.

For the two dimensional cases, the corresponding situation is not as clear as the one-dimensional case. Yet, we would be able to look at some simple configuration and show that the approximated problem gives a minimum value closes to the original one. We will consider the following three configurations where the partition itself gives (i) circles, (ii) rectangles, and (iii) some regular M-polygons. For the first two cases, we can express the solution analytically. For the last case, we numerically compute the first eigenvalue of the partition.



(a) (b)

Fig. 3: (Section 5) λ_1 and λ_1' and their ratio for different regular M-polygons. (a) The value of λ_1 will converges to 5.7830 as the number of sides increases. (b) The ratio of λ_1 and λ_1', denoted by r_M, will approach to $r_\infty = j_{0,1}^2/6$ as M increases.

(i) Circle. Consider a domain Ω to be a circle with radius r_0 center at the origin. Since $\sqrt{\lambda_1} r_0$ equals to the first root of the zeroth order Bessel's function of the first kind (denoted by $j_{0,1}$), the first eigenvalue λ_1 of this domain is approximately given by $(j_{0,1}/r_0)^2 = 5.7830/r_0^2$. On the other hand, the signed distance function representing the domain Ω is given by $\phi(x, y) = r_0 - \sqrt{x^2 + y^2}$. This implies that the value of λ_1' is given by $\lambda_1' = 6/r_0^2$. Therefore, we notice that $\lambda_1 = r_\infty \lambda_1' \simeq 0.9638\lambda_1'$ where $r_\infty = j_{0,1}^2/6$. Similar to the previous one dimensional case, our approximation for λ_1' actually slightly over-estimates the actual minimizer for λ_1 in case if the partition itself is a circle.

(ii) Rectangle. Assume that the partition gives a rectangular subdomain with the length and the width given by L_1 and L_2, respectively. It is well-known that the first eigenvalue is given by

$$\lambda_1 = \pi^2 \left(\frac{1}{L_1^2} + \frac{1}{L_2^2} \right) = \frac{\pi^2}{L_1^2} \left(1 + \frac{1}{\theta^2} \right)$$

if $L_2 = \theta L_1$. Without lost of generality, we assume that $L_2 \leq L_1$ so that $0 < \theta \leq 1$. On the other hand, the estimated eigenvalue λ_1' is given by $\lambda_1' = 24/[L_1^2(2-\theta)\theta^2]$. Hence the ratio

$$\frac{\lambda_1}{\lambda_1'} = \frac{\pi^2}{24}(\theta^2+1)(2-\theta)\,.$$

For $0 < \theta \leq 1$, the expression on the right hand side is bounded above and below by $\pi^2/12 \approx 0.8224$ and $25\pi^2/324 \approx 0.7615$ when $\theta = 1$ and $\theta = 1/3$, respectively.

(iii) Regular M-polygon. Assume the partition gives a regular M-polygon with the vertices lying on the unit circle, i.e. the vertices of the polygon are given by

$$\left\{ \left(\cos \frac{2i\pi}{M}, \sin \frac{2i\pi}{M} \right) \right\}_{i=0}^{M-1} .$$

In the interior of this polygon, the signed distance representation of the level set function can be expressed by a piecewise linear function

$$\phi(x,y) = \cos \frac{\pi}{M} - x \cos \left[\frac{(2j+1)\pi}{M} \right] - y \sin \left[\frac{(2j+1)\pi}{M} \right]$$

where j is the greatest integer less than $M\theta/2\pi$ and θ is the polar angle of the point (x,y) from the origin. Based on this piecewise linear representation of the level set function, one can analytically determine λ_1'. There is, unfortunately, no exact formula for the first eigenfunction within the M-polygon (except the square and the circle cases as discussed above). Therefore, we can only numerically approximate the eigenfunction to determine the corresponding first eigenvalue λ_1. Figure 3 shows both λ_1 and our estimate λ_1', as well as the ratio between two quantities (denoted by r_M). For $M = 4$, i.e. the square case, the discussion reduces back to the previous case when $\theta = 1$. We have $\lambda_1 = r_4 \lambda_1' \approx 0.8224 \lambda_1'$. We see that as M gets bigger, the ratio approaches to the limiting case when the partition is a circle giving $\lambda_1 = r_\infty \lambda_1' \approx 0.9638 \lambda_1'$ as shown in case (i) above.

(iv) A mixture of regular polygons. In the case when the optimal configuration consists of a mixture of regular M-polygons ($M \geq 4$), we have

$$\sum_{k=1}^{K} \lambda_1(U_k) = \sum_{k=1}^{K} \left[\frac{\lambda_1(U_k)}{\lambda_1'(U_k)} \right] \lambda_1'(U_k) = \sum_{k=1}^{K} r_{M(k)} \lambda_1'(U_k)$$

where the constant $r_{M(k)}$ is the ratio of λ_1 and λ_1' in the $M(k)$-polygonal subdomain U_k. Therefore, how well our approach approximates the exact minimizer depends on the distribution of $r_{M(k)}$'s given by

$$\sum_{k=1}^{K} \lambda_1(U_k) \simeq \left[\frac{1}{K} \sum_{k=1}^{K} r_{M(k)} \right] \left[\sum_{k=1}^{K} \lambda_1'(U_k) \right] .$$

Since r_M is bounded below and above by r_4 and r_∞, respectively, the error we have made in each term of the sum is bounded by constants $r_\infty - r_4 = j_{0,1}^2/6 - \pi^2/12 \simeq 0.1414$.

6 Comparison with Existing Methods

In this section, we compare our proposed method with the approach developed in [8] based on the Schrödinger relaxation operator and a related approach in [5], and also the diffusion generated motion approach in [39].

Interface Representation. Different approaches use a different interface representation for the partition boundaries. In particular, [8] and [39] are developed based on a labeling representation and a phase-field type representation. Since our approach uses the level set method, we can obtain a continuous representation from the interpolation and provide the so-called subcell resolution. More discussion on this will be demonstrated in Section 7.1.

Irregular Domains. Both our approach and the one in [8] can easily handle different types of domains by simply introducing an extra level set function Ψ to the penalty term or an extra density function ϕ_0 in the implementation. Since [39] relies heavily on the FFT or SHT, it might require extra care in designing an efficient numerical approach in the diffusion stage.

Computational Complexity in Terms of K. For large-K problems, [8] seems to perform better in terms of computational complexity. While most computational efforts are spent on determining the eigenvectors, this process can be done in parallel first followed by a normalization step to capture the interaction between different phases. Together with the grid restriction procedure [5], the overall computational effort in each iteration is almost independent of K. In particular, [8] is able to obtain a solution for K as large as 512. For both our approach and the one developed in [39], however, the computational complexity is linear in K. One possible way to lower the complexity in our approach is to follow the multiphase level set framework [28] which is able to reduce the number of level set functions from K to $\log_2 K$.

Computational Speed. Like the typical level set method, our PDE based approach seems to be the least efficient in terms of computational speed. Due to the stability condition in the time evolution, one has to iterate the algorithm for a large number of times to reach a steady-state solution. In the current implementation, we simply use the typical full level set implementation. To speed up the computations and improve the resolutions in the final partition, one can implement the local level set approach [35] or some adaptive level set method [31]. They can both significantly improve the computational efficiency of the PDE-based approach. Also, since one evolves a level set function for each partition, we can apply an operator splitting method and develop a parallelize version for the overall algorithm.

Generalization to Related Partition Problems. The approach in [8] can be easily generalized to solve some related partition problems related to other eigenvalues of the Laplacian. However, both our method or [39] will need further exploration.

7 Numerical Examples

In this section, we will test our algorithm on several numerical examples to demonstrate the effectiveness and the performance of the proposed level set method for the Dirichlet k-partition problem. All results are obtained on a Linux computer with a 3.40GHz Intel Core i7-4770 and a 16GB RAM. Section 7.1 shows a 1D example. We compare our solutions with other methods as developed in [8] and [39]. In Section 7.2, we will repeat some of the 2D examples from [39] and compare our solutions with their method based on the MBO approach. Since these problems have no theoretical solutions, we are not able to compare quantitatively and demonstrate the convergence of the method but will be able to look at the solution qualitatively by checking the shape of the partitions. We will also look at the sum of the first eigenvalue of each partition and will show that we can provide some new configurations leading to some possible new insight into the problem. In Section 7.3, we will consider some different boundary conditions and domain U. Finally, we test our approach in 3D in Section 7.4.

7.1 Partitions on an One-dimensional Interval

The first numerical test is a simple 1D example. As proved in Section 5, our level set formulation is equivalent to the original eigenfunction formulation. Here, we consider the 3-partition problem (i.e. $K = 3$) on the $[0, 1]$ intervals with the periodic boundary condition. We have also implemented the methods in [8] and [39] for comparisons. The mesh size is $\Delta x = 1/512$. The exact solution is given by 3 disjoint intervals with a length of $1/3$. We start with three disjoint random intervals in $[0, 1]$. To initialize the approach in [8], we first generate three random vectors of length 512 and then normalize them so

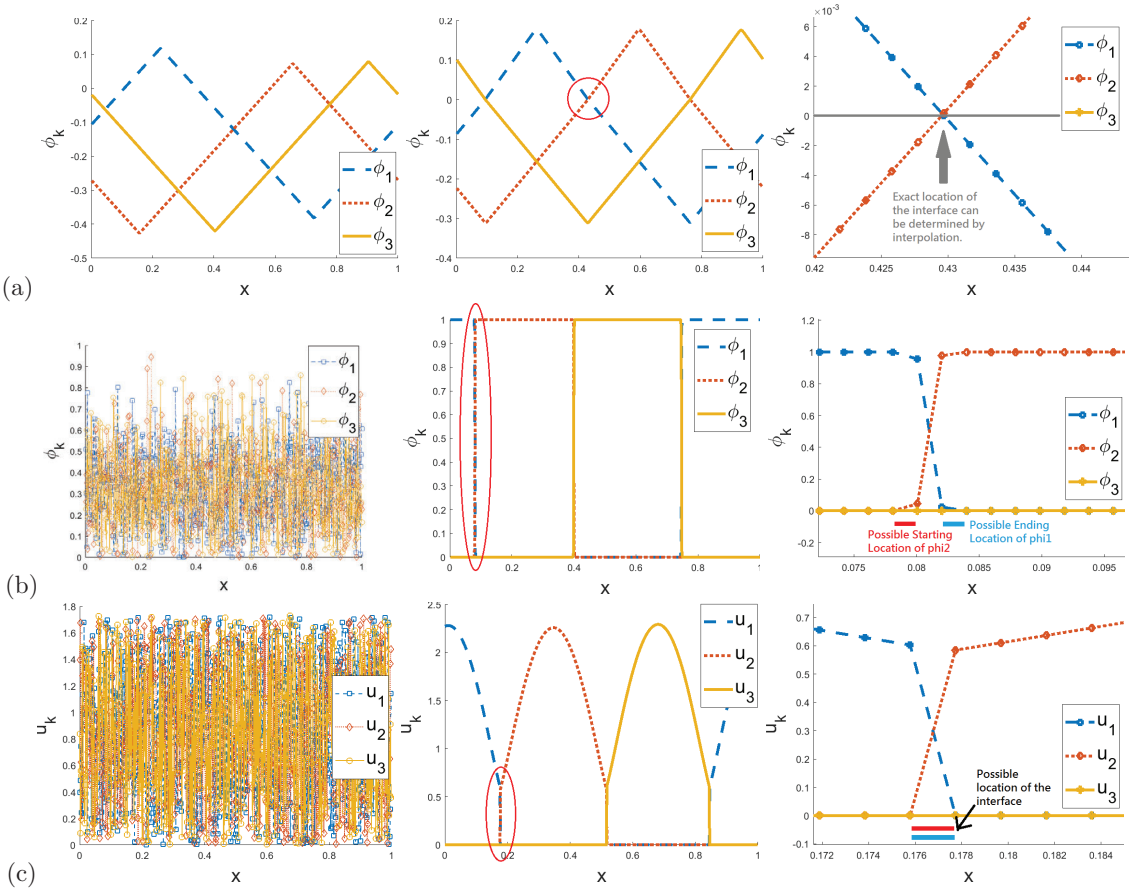


Fig. 4: (Section 7.1) The initial condition, the final solution and a zoom-in of the final solution near the interface using (a) our method, (b) [8] and (c) [39].

	Our Method	Method in [8]	Method in [39]
Iteration (N)	10,000	500	1,000
Time (in seconds)	7.49	4.72	0.12
$\epsilon^{(N)}$	1.3904×10^{-5}	7.2091×10^{-10}	8.8818×10^{-16}
Length of the partitions	{0.3350,0.3331,0.3324}	{0.3184,0.3476,0.3340}	{0.3398,0.3282,0.3320}
Max error	1.7×10^{-3}	1.49×10^{-2}	6.5×10^{-3}

Table 1: (Section 7.1) The comparison of the 3-partition problem in 1D using our method, [8] and [39].

that the pointwise sum is one at each entry across the three vectors. For the method as described in [39], we generate 3 random vectors of length 512 then normalize them so that the L2-norm is one for each of these vectors. For the parameters, we take $\mu = 10^4$ and $\Delta t = 5 \times 10^{-8}$ in our method. For the method of [8], we take $C = 10^6$, $\Delta t = 5 \times 10^{-2}$ and $\alpha = 1$. For the method in [39], we take $\tau = 0.001$.

Figure 4 shows the initial and final conditions in each of these methods. We see that all three methods give accurate approximations to the correct configuration (3 disjoint intervals with an equal length of 1/3). However, due to the difference in the interface/boundary representation, there is still a small difference in accuracy. The method in [8] uses a labeling representation that tries to label those grid points which lie inside a certain partition. This implies that there is no resolution between grids. Similarly, the thresholding in [39] also destroys the sub-cell resolution in the interface after the thresholding. In our representation, however, since the interface of each partition is represented implicitly by the level set function, one can obtain a continuous representation from the interpolation.

Table 1 shows some statistics on all numerical solutions. In terms of computational time, the proposed method indeed takes the longest time. The difference in computational efficiency will be more significant, especially in higher dimensions. This is explained in Section 6. Our method also requires the largest

number of iterations to reach a steady-state solution. In particular, we define the difference in the numerical solution between two consecutive iterations in both [8] and our approach to be $\epsilon^{(n)} = \|\phi^{(n)} - \phi^{(n-1)}\|_\infty$, and define $\epsilon^{(n)} = \|u^{(n)} - u^{(n-1)}\|_\infty$ to be the corresponding difference in [39]. We note that the value of $\epsilon^{(N)}$ in our approach is significantly larger than that by the other two methods. This indicates that our time iteration converges much slower than others. This is mainly because the gradient descent approach only allows the level set functions to gradually adjust its value in the time evolution. However, because the signed distance property of the level set function, we are able to provide a better approximation of the interface locations from interpolation.

7.2 Partitions on a Two-dimensional Flat Torus

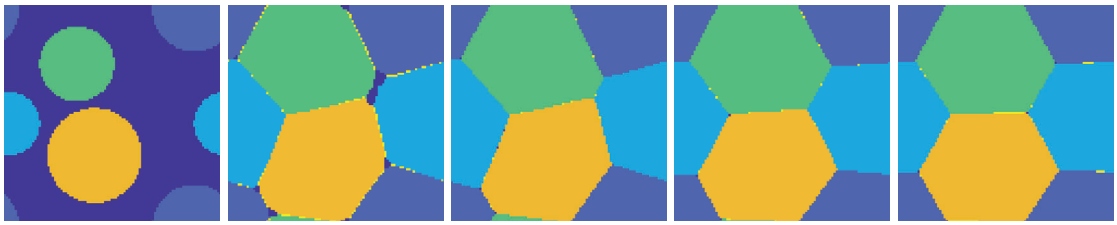


Fig. 5: (Section 7.2) Partition at different times for $K = 4$. (From left to right) The initial condition and the intermediate solution after 10^3, 2×10^3, 10^4 and 2×10^4 iterations.

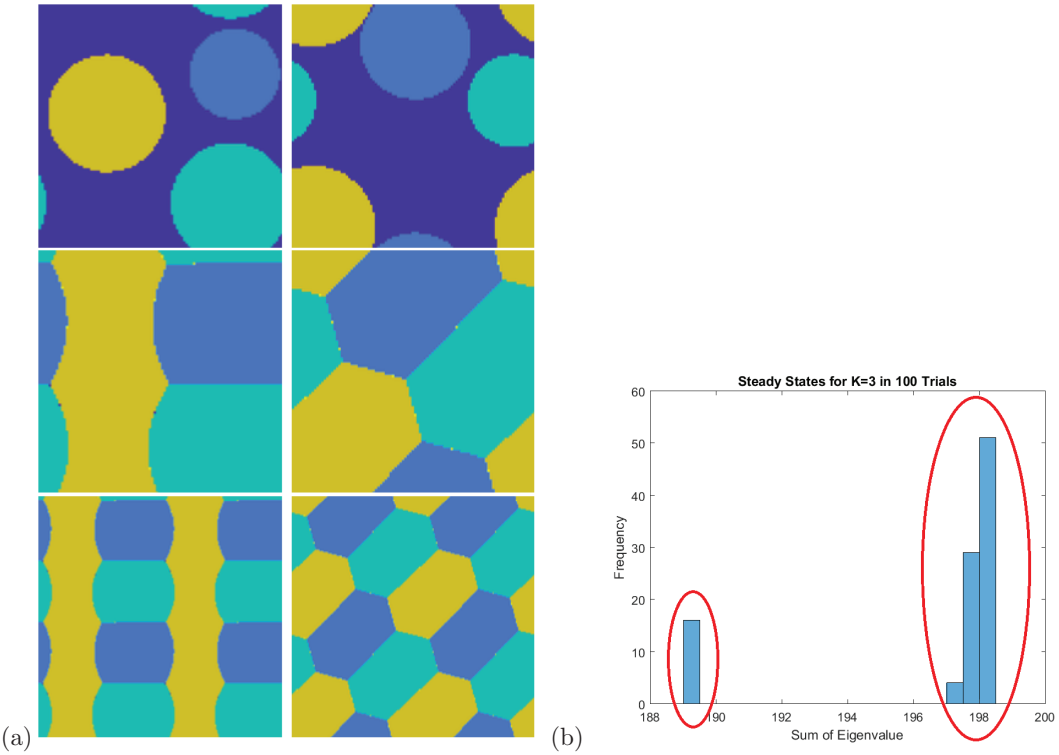


Fig. 6: (Section 7.2) Results for $K = 3$ with the periodic boundary condition on ∂U. (a) The first row shows the initial conditions. The second row shows the corresponding steady-state solution. The third row shows the steady-state with the periodic extension. The sum of λ_1's are given by (from left to right): 197.2 and **189.2565**, respectively. The global minimum is bolded. (b) Histogram showing the sum of λ_1's of the steady-state solution in 100 trials.

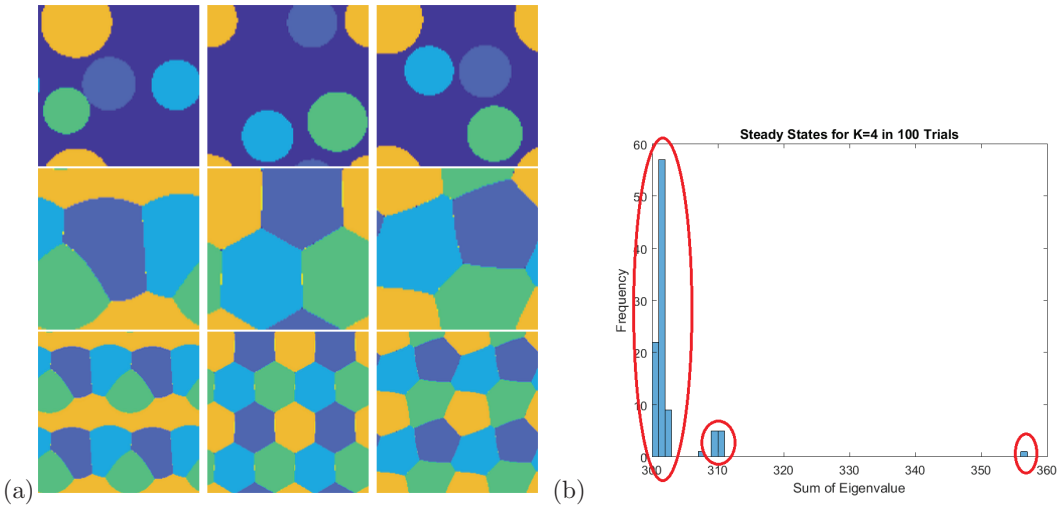


Fig. 7: (Section 7.2) Results for $K = 4$ with the periodic boundary condition on ∂U. (a) The first row shows the initial conditions. The second row shows the corresponding steady-state solution. The third row shows the steady-state with the periodic extension. The sum of λ_1's are given by (from left to right): 356.4, **300.1** and 309.8, respectively. The global minimum is bolded. (b) Histogram showing the sum of λ_1's of the steady-state solution in 100 trials.

In this example, we will find out the k-partition (for K ranging from 3 to 10) inside a unit square $U = [0, 1]^2$ with the periodic boundary, i.e. the so-called flat torus. In the numerical simulations, we use a mesh size $\Delta x = \Delta y = 0.01$ and the regularization parameter $\mu = 10^4$ for the non-overlapping constraint. The initial condition is constructed by assigning K random non-overlapping circles. We have observed that we have to use a smaller time step Δt as we increase the value of K in the simulations. In particular, we take $\Delta t = 2 \times 10^{-7}$ for K less than or equal to 6, $\Delta t = 10^{-7}$ for $K = 7, 8$ and $\Delta t = 5 \times 10^{-8}$ for $K = 9$ and 10. We stop the iteration if the difference E defined in Section 3.6 is less than $\epsilon = 10^{-6}$, or if the number of iterations exceeds 10^6 for $K \leq 6$, 2×10^6 for $K = 7, 8$, or 4×10^6 for $K = 9, 10$ (where the corresponding terminal time is $T = 0.2$). For the regularization of the level set function, we reinitialize the level set function by updating the corresponding partial differential equations 20 steps for every 10^3 steps in the main iterations.

As a simple demonstration of how our approach behaves, we first consider a relatively simple case where $K = 4$. Figure 5 shows the evolution of the level set functions at different times. We first initialize the computation with K (i.e. 4 in this case) disks with a random radius corresponding to K different partition of the domain. The region outside these disjoint disks is the vacuum region which will be automatically eliminated by our approach. As we can see from the simulation, such a vacuum region is almost filled after 2×10^3 iterations, as shown in the second subfigure. When the iteration converges, the hexagonal structure obtained in the steady-state matches very well with the those reported in [22,39].

For the rest of this section, we will consider the partition solutions for $K = 3$ to $K = 10$ in detail. We will also provide some comparisons with the solutions obtained by [39].

3-*Partition.* The case with $K = 3$ is an interesting one. With the same setting, we repeat the experiment 100 times with different non-overlapping circles as the initial conditions. The results are shown in Figure 6. We notice that there are two possible steady-state solutions as shown in Figure 6(a). Except for the one commonly reported in previous literature showing on the left-hand side, we discover another periodic structure as shown on the right-hand side in Figure 6(a). To the best of our understanding, this is a new configuration that has never been reported elsewhere. More importantly, this structure provides a *smaller* value of $\sum_{k=1}^{K} \lambda_1(U_k)$ (not of our proposed $\lambda_1'(\phi_k)$) which makes us believe that this gives the global minimum of the k-partition problem. We have shown the histogram of the sum of the first eigenvalues in Figure 6(b). Among all 100 random initial conditions, we have found that 16 of them have led to this new minimizer while the other 84 trials felt into the local minimum reported in [39] or other related papers.

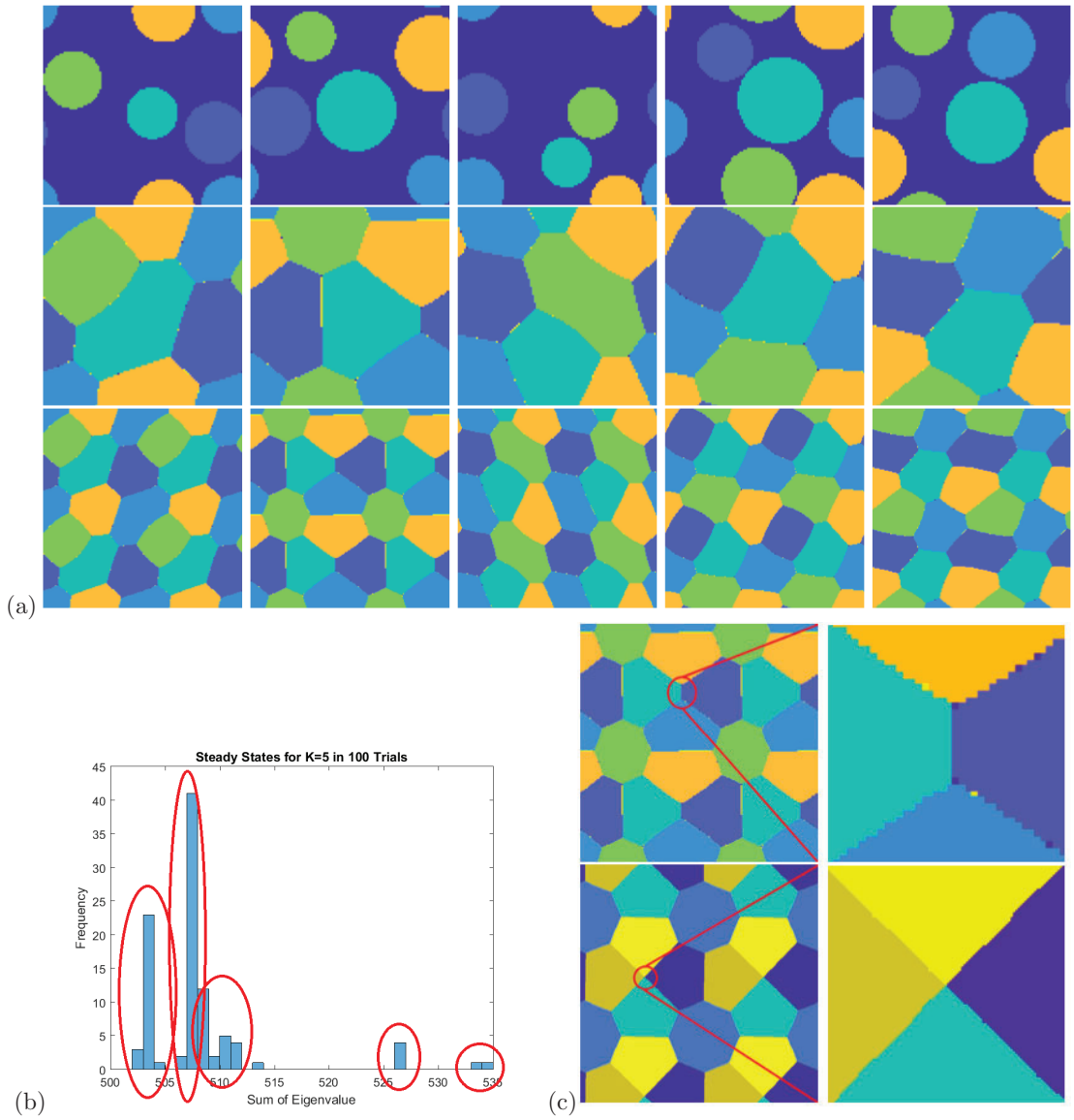


Fig. 8: (Section 7.2) Results for $K = 5$ with the periodic boundary condition on ∂U. (a) The first row shows the initial conditions. The second row shows the corresponding steady-state solution. The third row shows the steady-state with the periodic extension. The sum of λ_1's are given by (from left to right): 526.1, 507.0, 534.0, **502.7** and 508.1, respectively. The global minimum is bolded. (b) Histogram showing the sum of λ_1's of the steady-state solution in 100 trials. (c) The splitting of a quadruple junction into 2 triple junctions as computed using (top) our method and (bottom) the method in [39].

4-*Partition.* Figure 7 shows the solutions to the case where $K = 4$. We have observed that the majority of the initial conditions will lead to the same minimizer consisting of four hexagonal structures.

5-*Partition.* The case $K = 5$ has multiple steady-state solutions as shown in Figure 8(a). If we take a closer look at the second structure on the left of Figure 8(a), it is similar to the global minimum as reported in [39]. As shown in Figure 8(c), we have also observed that in some evolutions, a quadruple junction would split into two triple functions. This splitting was mentioned in [40] and is due to the instability of quadruple junction to the two triple functions. Unfortunately, such a splitting increases the sum of the first Laplace-eigenvalue which makes itself not being the global minimum.

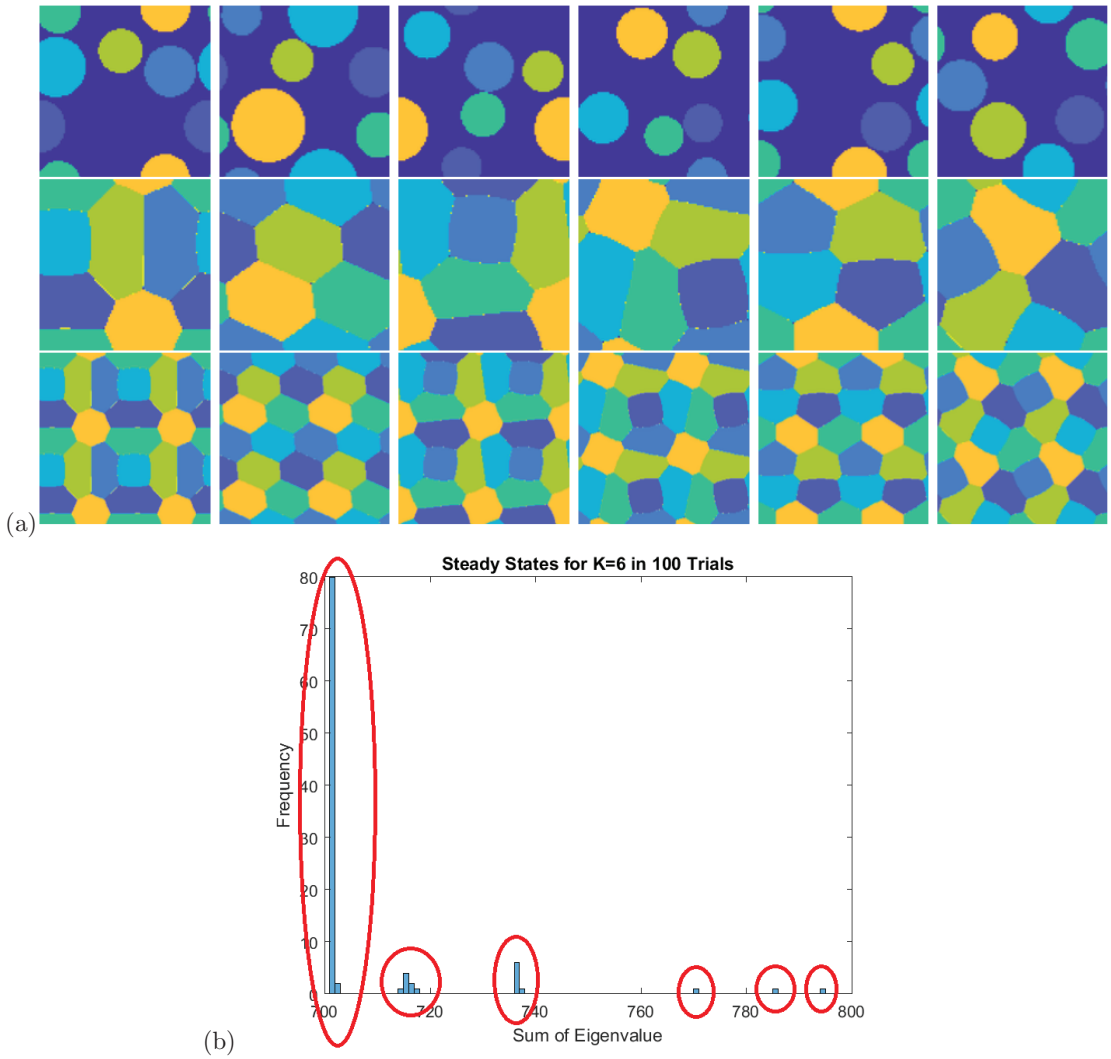


Fig. 9: (Section 7.2) Results for $K = 6$ with the periodic boundary condition on ∂U. (a) The first row shows the initial conditions. The second row shows the corresponding steady-state solution. The third row shows the steady-state with the periodic extension. The sum of λ_1's are given by (from left to right): 794.7, **701.3**, 785.0, 770.8, 714.9 and 736.4, respectively. The global minimum is bolded. (b) Histogram showing the sum of λ_1's of the steady-state solution in 100 trials.

6-*Partition.* For the case $K = 6$, the optimal packing is back to the hexagonal structure. Different initial conditions may result in some minor local minimum as shown in Figure 9. However, as shown in Figure 9(b), the chance for falling into an incorrect minimum is relatively small.

A Comparison with [39]. As a comparison to the method proposed by [39], we have plotted some of our partition results in Figure 10. They match generally well with those as demonstrated in [39]. When computing the discretized Laplace-eigenvalues numerically, we interpolate the level set function linearly to calculate the exact boundary. However, method in [39] only provides a classification if a grid point is in a certain subdomain U_k. It does not have the so-called subcell resolution. We therefore simply treat the middle of the grid point to be the location of the partition boundary. The result shows that our method aligns with the method proposed by [39]. Except for $K = 3$ and $K = 5$, the global minimum proposed by our method has the same structure compared with [39].

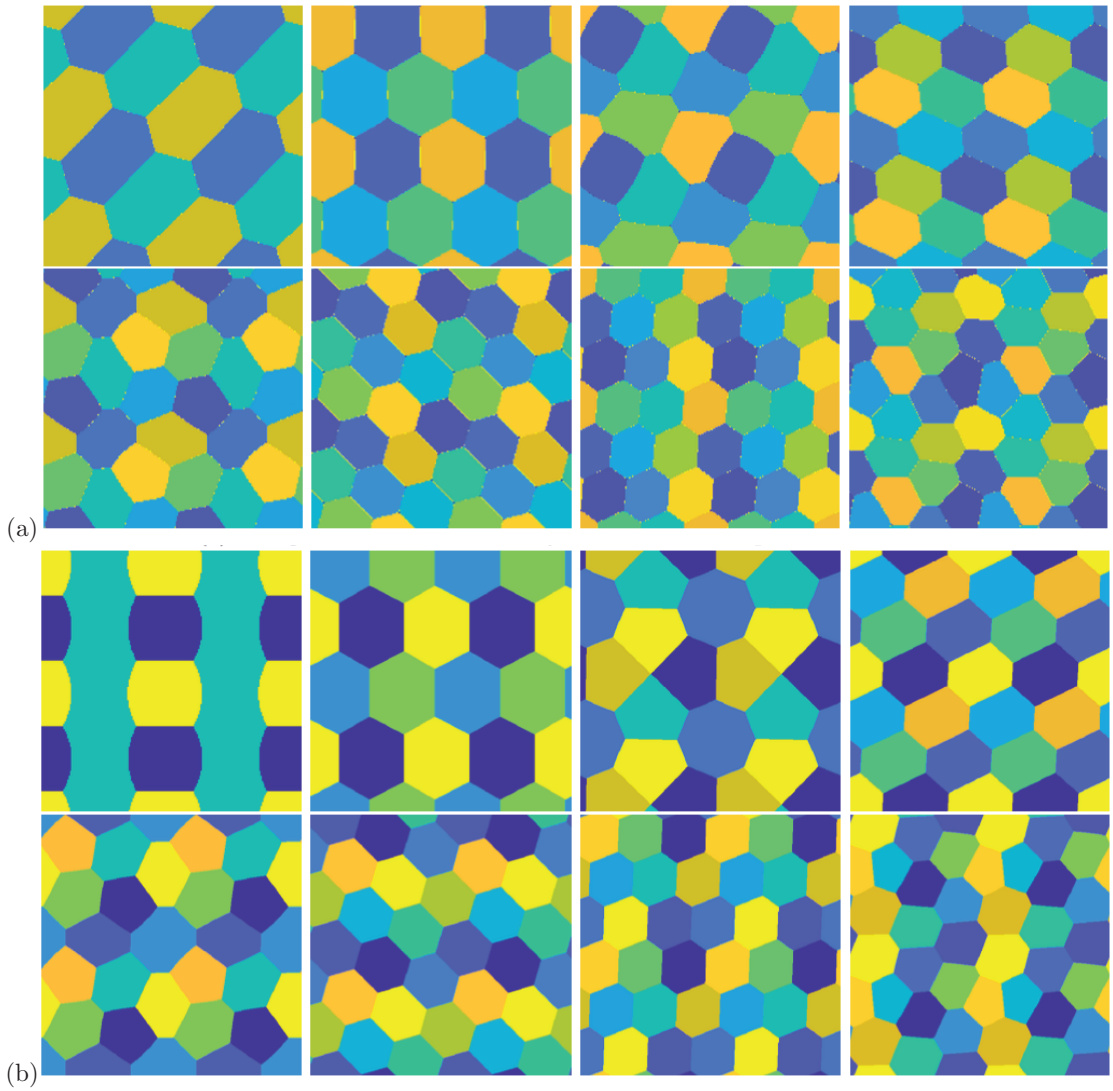


Fig. 10: (Section 7.2) Comparison of the global minimum in a 2D-torus by our method and the approach in [39] with K between 3 and 10. (a)Our proposed method (The sums of λ_1 are 189.2565, 300.0990, 502.7314, 701.2853. 963.1235, 1195.5875 and 1529.6110) (b) the approach in [39] (The sums of λ_1 are 190.6018, 300.6115, 495.5503, 702.6459, 958.6348, 1204.0997, 1540.6496 and 1960.5951) The computational time for our method is approximately 25, 11, 16, 15, 22, 26, 23 and 34 minutes for $K = 3, \ldots, 10$, respectively.

7.3 Partitions on a Bounded Domain

As discussed, the level set method developed here in this work can be easily applied to a Dirichlet k-partition problem on an arbitrary domain. In the following numerical experiments, we consider several arbitrary domains $U \subset [-0.5, 0.5]^2$ and determine the optimal k-partition with Dirichlet boundary condition on ∂U. The domains we considered are (i) the unit square, (ii) the unit circle, (iii) a five-folded star-shape domain, (iv) an annulus, (v) a regular triangle and (vi) a 3-Hexagon. The first four domains are constructed analytically by the level set function ψ of U given by

$$\psi_1(x, y) = \max(|x - 0.5|, |y - 0.5|) - 0.5 \quad \text{(Unit Square)}$$

$$\psi_2(x, y) = \sqrt{(x - 0.5)^2 + (y - 0.5)^2} - 0.5 \quad \text{(Unit Circle)}$$

$$\psi_3(r, \theta) = r - 0.1\cos(5\theta) - 0.4 \quad \text{(Five-folded Star)}$$

$$\psi_4(r,\theta) = |r - 0.375| - 0.125 \quad \text{(Annulus)}\,.$$

In the fifth case, the triangular domain is formed by connecting the three vertices $\left\{\cos\frac{2i\pi}{3}, \sin\frac{2i\pi}{3}\right\}_{i=1,2,3}$. In the sixth case, the 3-Hexagon is formed by merging 3 regular hexagons with the length of each side 0.25. For these two domains, the analytical form of the level set functions are not that simple and hence they are constructed on a point-by-point basis.

We carried out the computations on the mesh with $\Delta x = \Delta y = 0.01$. The regularization parameter is chosen to be $\mu = 10^4$. Like the previous computations, the initial condition is constructed by K random non-overlapping circles inside U. We take $\Delta t = 2 \times 10^{-7}$ for K less than or equal to 6, $\Delta t = 1 \times 10^{-7}$ for $K = 7$ and 8, and $\Delta t = 5 \times 10^{-8}$ for $K = 9$ and 10. We reinitialize the level set function ϕ by 20 steps for each 100 steps in the main iterations. We stop the evolution if the difference E is less than $\epsilon = 10^{-6}$, or if the number of iterations exceeds 10^6 for $K \leq 6$, 2×10^6 for $K = 7, 8$, or 4×10^6 for $K = 9, 10$ (where the corresponding terminal time is $T = 0.2$.)

Fig. 11: (Section 7.3) The k-partition problem on a unit square for $K = 3, \cdots, 10$. The sum of λ_1's are given by 200.9, 317.3, 520.1, 756.8, 1015.1, 1267.2, 1573.0 and 1971.5, respectively. The computational time is approximately 25, 28, 32, 56, 65, 71, 49 and 85 minutes for $K = 3, \ldots, 10$, respectively.

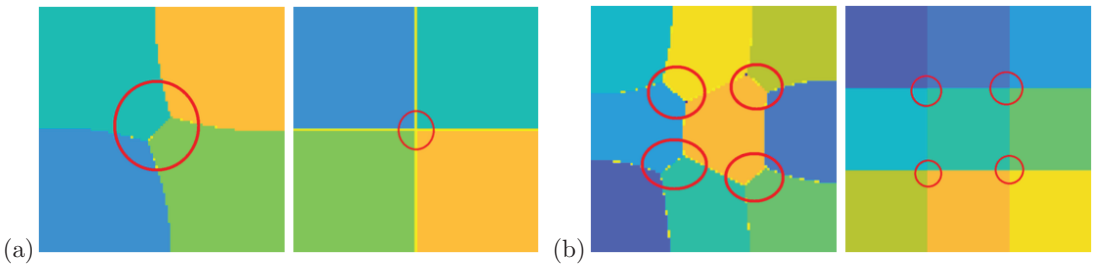


Fig. 12: (Section 7.3) (a) The k-partition problem on a unit square for $K = 4$. Our solution is plotted on the left and the uniform division is shown on the right. The sum of λ_1's are given by 317.3 and **315.8**. (b) The k-partition problem on a unit square for $K = 9$. The sum of λ_1's are given by **1573.0** and 1598.9.

A Unit Square. For the square domain with the signed distance function ψ_1, the global minimum found by our method is shown in Figure 11. We notice that the global minimum for $K = 3, 5, 6, 8$ and 9 are symmetric vertically or horizontally, while the global minimum for $K = 4, 5, 7$ and 10 are symmetric in diagonal. For the cases when K being a squared number, i.e. $K = 4$ and $K = 9$, we compare our partition

result with the uniform division as shown in Figure 12 and notice that the behavior in these two cases is quite different. For $K = 4$, the unstable quadruple junctions are split into two triple junctions in our method. This splitting slightly increases the sum of the eigenvalues from 315.8 to 317.3. Conversely, when $K = 9$, we have checked that the 9-partition we obtained has a significantly smaller sum of the eigenvalue (with a value 1573.0) than the value in the uniform partition (with a value 1598.9). This shows that the uniform partition *might not* necessarily be the optimal partition for a square number K.

There are reference solutions for both the 3-partition and the 5-partition where the sum of all three and five first eigenvalues are given by 199.74 and 521.47, respectively [9,24]. Comparing to our computed values 200.9 and 520.1, our solutions match reasonably well with these reference solutions.

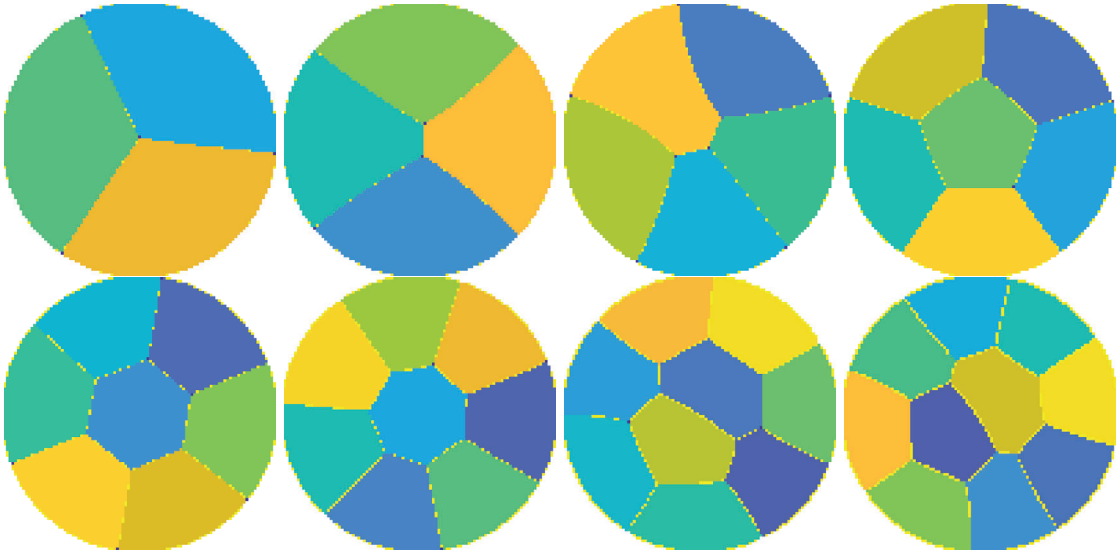


Fig. 13: (Section 7.3) The k-partition problem on a unit disk for $K = 3, \cdots, 10$. The sum of λ_1's are given by 242.1, 422.3, 664.4, 922.4, 1204.2, 1557.9, 2000.7 and 2395.4, respectively. The computational time is approximately 24, 27, 33, 37, 41, 46, 50 and 55 minutes for $K = 3, \ldots, 10$, respectively.

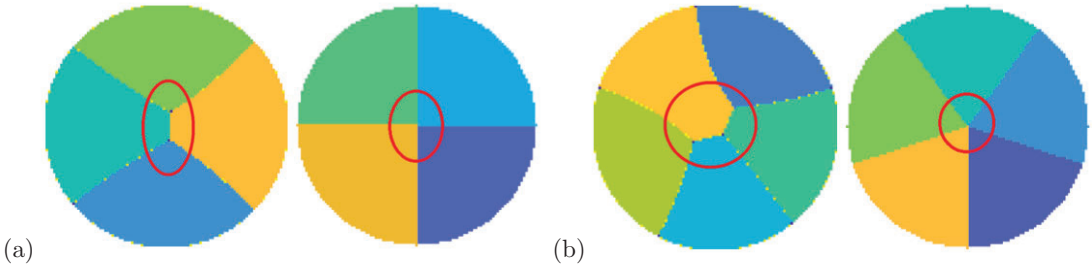


Fig. 14: (Section 7.3) (a) The k-partition problem on a unit disk for $K = 4$. Our solution is plotted on the left and the uniform division is shown on the right. The sum of λ_1's are given by 422.3 and **421.8**. (b) The k-partition problem on a unit disk for $K = 5$. The sum of λ_1's are given by 664.4 and **663.8**.

A Unit Disk. For the circle domain with the signed distance function ψ_2, the global minimum we obtained is shown in Figure 13. For $K = 3$, the circle is divided into 3 equal 120-degree sectors. For $K = 4$ and $K = 5$, it looks like that the global minimum should be a uniform division of a circle into different sectors. The instability of quadruple and quintuple junctions causes them to deviate from the uniform division. Therefore, we compare our solution with the uniform division as shown in Figure 14. The sum of the eigenvalues in the uniform division is 421.8 and 663.8 for $K = 4$ and $K = 5$ respectively. They are slightly lower than our computed values of 422.3 and 664.4. In fact, [24] has shown that the optimal

partition for $K = 5$ is the uniform division. After re-scaling the domain to have a unit area, our computed sum of the eigenvalues is $664.4 \times 0.5^2\pi \approx 521.8$ which is the same as the computed values in [24]. From $K = 6$ to $K = 8$, one partition is centered at the middle and the others are just uniformly around it. For $K = 9$ onward, more partitions will fall into the middle. There is a reference solution in [9] for the 3-partition on a unit disk where the sum of all three first eigenvalues is given by 190.37 . This matches reasonably well with our computed value $242.1 \times 0.5^2\pi \approx 190.15$ after rescaling.

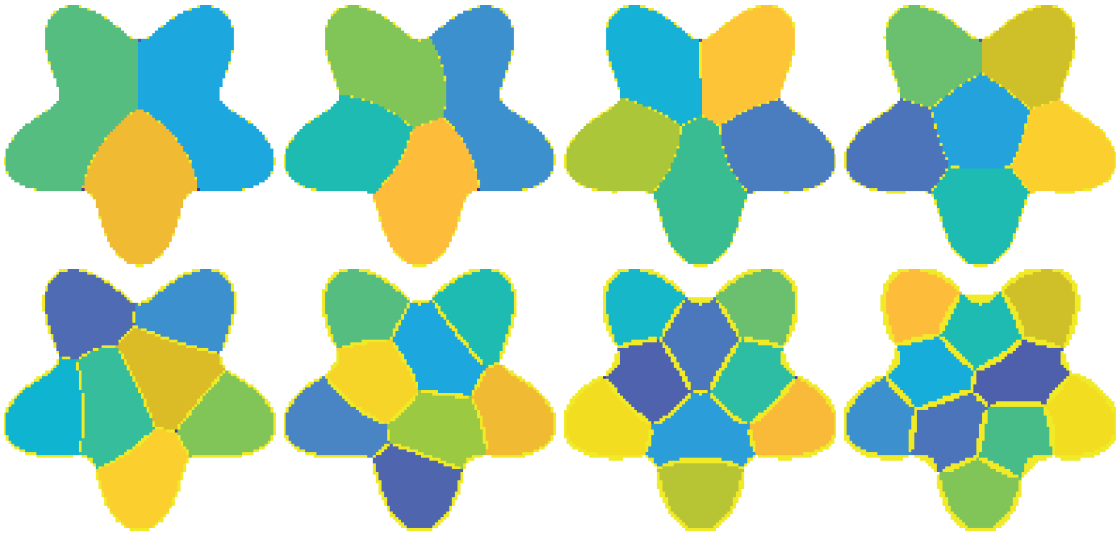


Fig. 15: (Section 7.3) The k-partition problem of a five-folded Star for $K = 3, \cdots, 10$. The sum of λ_1's are given by 454.8, 733.7, 936.7, 1267.7, 1770.9, 2191.5, 2626.8 and 3041.4, respectively. The computational time is approximately 23, 27, 31, 35, 40, 44, 49 and 54 minutes for $K = 3, \ldots, 10$, respectively.

A Five-folded Star-shape Domain. For the five-folded star domain with the signed distance function ψ_3, the global minimum generated by our method is reported in Figure 15. Not surprisingly, the instability of the quintuple junctions causes the global minimum in $K = 5$ to deviate from the 5 equal division of the five-folded star. From $K = 6$ onward, there is one partition at each wing of the star, and the other partition will fall into the middle part of the star. Besides, we notice that the thickness of the edge increases with the value of K. In general, when K increases, the area of each partition will drop. This will increase the size of γ. Hence, we should have a larger penalty coefficient μ to balance the effect of the increase of the coefficient in the diffusion and convection term.

An Annulus. For the annulus domain with the signed distance function ψ_4, it is constructed by 2 concentric circles with radius 0.25 and 0.5. The global minimum we obtained is shown in Figure 16. We notice that the result is simply a K equal division of the annulus. Similar to the case in a five-folded star, we can observe an increase in the thickness when K gets larger. If we gradually increase the penalty coefficient μ, the thickness will be reduced significantly as shown in Figure 17.

A Regular Triangle. For the case of a regular triangle, the global minimum we obtained is shown in Figure 18. To maintain a thin edge or vacuum region, we have taken $\mu = 10^4$ when $K = 3, 4$ and $\mu = 10^5$ when $K \geq 5$. Some interesting examples are the cases where K is a triangle number (i.e. $K = 3, 6, 10, \ldots$). The optimal partition is aligned rows by rows from the top down to the bottom of the triangle. In each row, there is exactly one extra partition compared with the previous row. In Figure 19, we compared our 3-partition solution with a reference solution found in [9]. The sum of the first three eigenvalues in the reference solution is 185.6, which is comparable to our computed value $552.1 \times \frac{3\sqrt{3}}{16} \approx 179.3$ after re-scaling the domain to have a unit area. This shows that our method provides a reasonable approximation to the original Dirichlet k-partition problem.

Fig. 16: (Section 7.3) The k-partition problem in an annulus for $K = 3, \cdots, 10$. The sum of λ_1's for $K = 3, \cdots, 10$ are given by 506.4, 718.6, 970.6, 1265.8, 1609.9, 2001.2, 2424.8 and 2853.4, respectively. The computational time is approximately 24, 27, 33, 37, 41, 46, 50 and 55 minutes for $K = 3, \ldots, 10$, respectively.

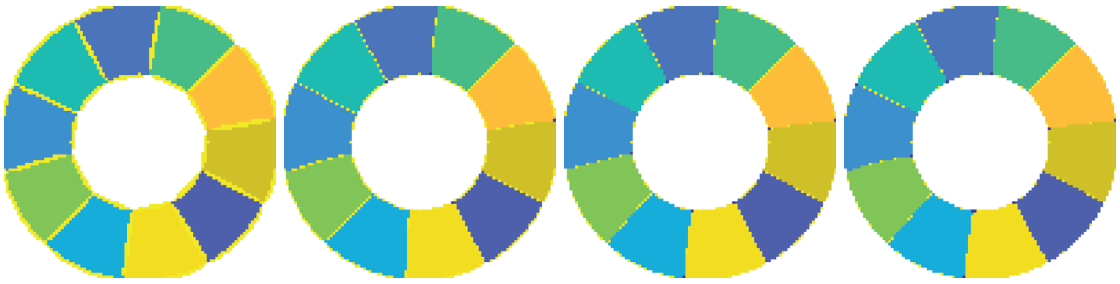


Fig. 17: (Section 7.3) The k-partition problem in an annulus for $K = 10$ for $\mu = 10^4, 2{\times}10^4, 4{\times}10^4, 8{\times}10^4$.

Fig. 18: (Section 7.3) The k-partition problem in a regular triangle for $K = 3, \cdots, 10$. The sum of λ_1's for $K = 3, \cdots, 10$ are given by 552.1, 1095.3, 1677.0, 2182.0, 3135.5, 4039.7, 4926.0 and 5760.6, respectively.

A 3-Hexagon For the case of a 3-hexagon, the domain is constructed by merging three regular hexagons with each side 0.25 in length. The global minimum partition we obtained is shown in Figure 20 for

Fig. 19: (Section 7.3) The 3-partition problem on a regular triangle. (a) The initial condition, (b) our solution and (c) The reference solution from [9]. The sum of λ_1's are (b)179.3 and (c)185.6, respectively (after re-scaling the domain to have the unit area).

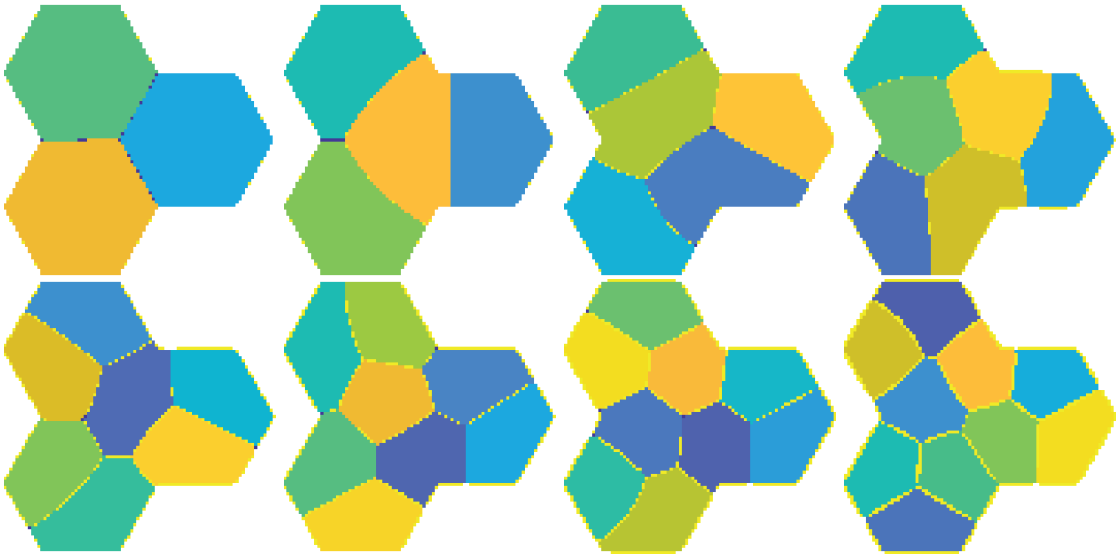


Fig. 20: (Section 7.3) The k-partition problem in a 3-Hexagon for $K = 3, \cdots, 10$. The sum of λ_1's for $K = 3, \cdots, 10$ are given by 343.1, 680.8, 1125.0, 1536.2, 2044.4, 2547.1, 3074.2 and 3718.2, respectively.

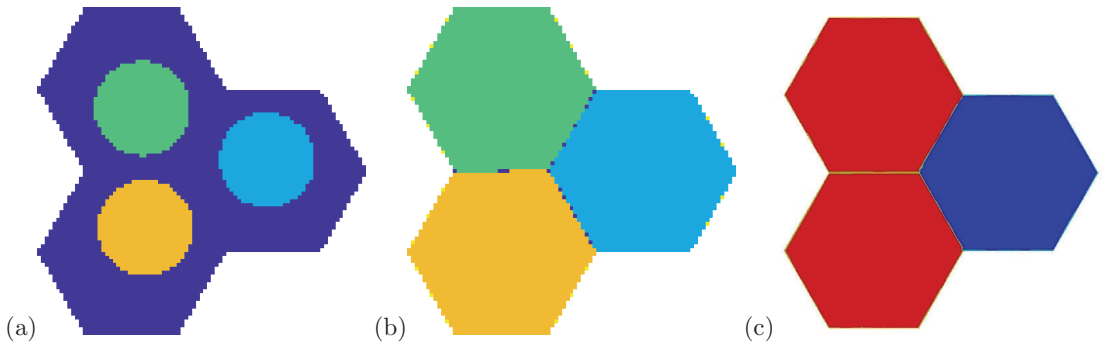


Fig. 21: (Section 7.3) The 3-partition problem on a 3-Hexagon. (a) The initial condition, (b) our solution and (c) The reference solution from [9]. The sum of λ_1's are (b)167.1353 and (c)167.3109, respectively (after re-scaling the domain to have the unit area).

$K = 3$ to $K = 10$. A reference solution from [9] for $K = 3$ simply divides the whole domain back to three separated hexagons, as shown in Figure 21(c). Using our method, we can also obtain the same partition as shown in Figure 21(b). The sum of the eigenvalues in our method after re-scaling to unit area is $343.0955 \times \frac{9\sqrt{3}}{32} \approx 167.1353$, which is comparable to their computed values 167.3109.

7.4 Partitions on a Three-dimensional Domain

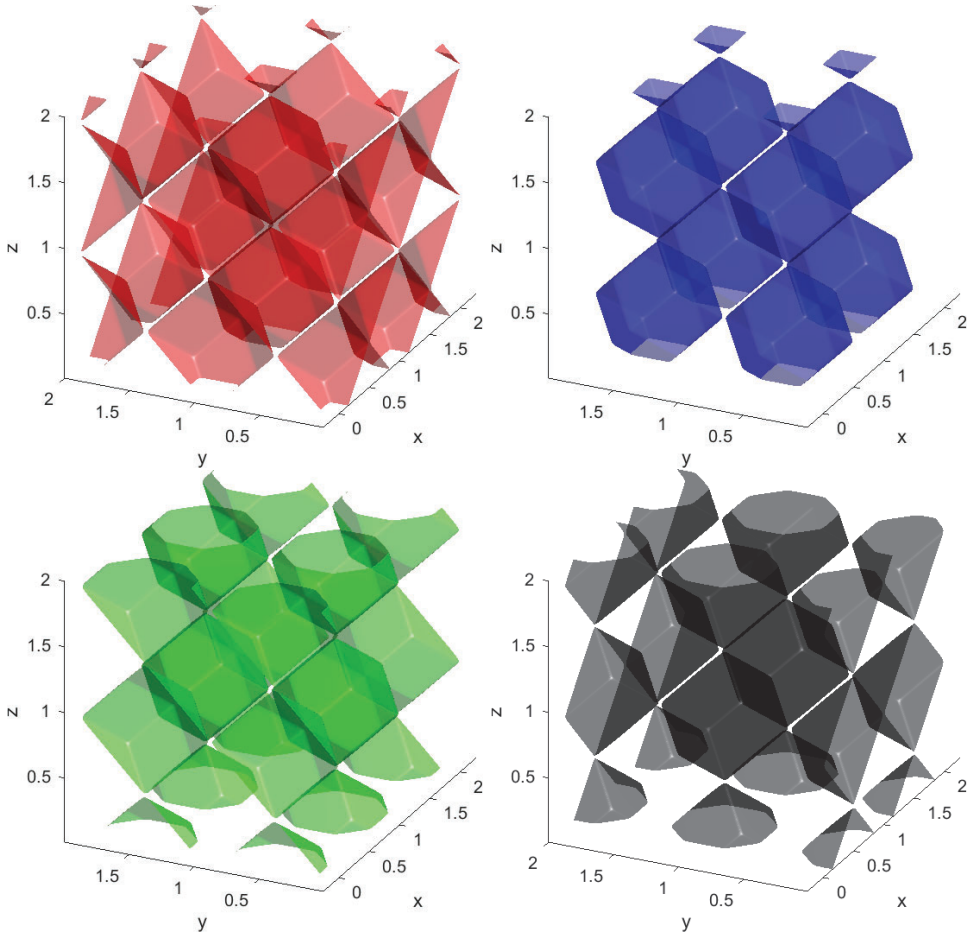
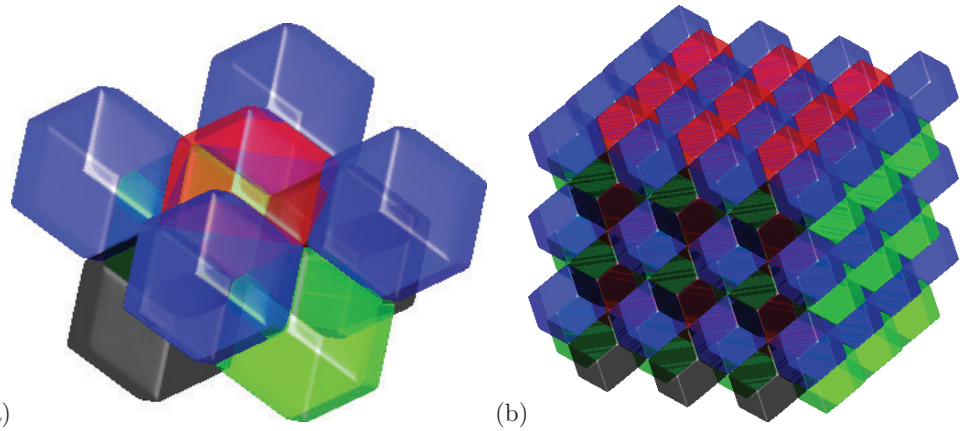
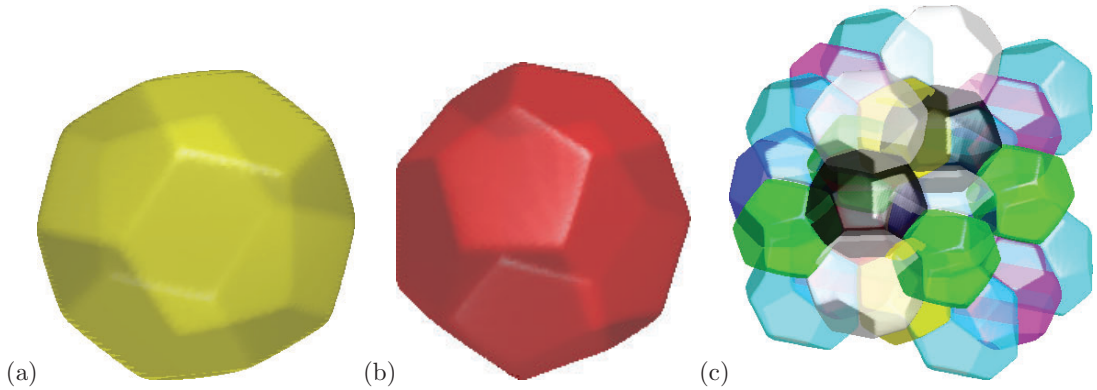


Fig. 22: (Section 7.4) The zero level set of ϕ_k, $k = 1, 2, 3, 4$ for the 4-partition problem on a periodic unit cube with the periodic extension.



(a) (b)

Fig. 23: (Section 7.4) The 4-partition problem on a periodic unit cube. (a) All rhombic dodecahedra inside the periodic extension of the unit cube. (b) A macroscopic view showing the packing structure obtained in the 4-partition. The computational time is approximately 588 minutes.



(a) (b) (c)

Fig. 24: (Section 7.4) The 8-partition problem on a periodic unit cube. (a-b) Two different types of shape obtained. The first shape has 12 pentagonal faces while the second one has 12 pentagonal faces plus 2 hexagonal faces. (c) The overall packing structure for the 8-partition problem. The computational time is approximately 1246 minutes.

In this 3D example, we compute the 4-partition and 8-partition inside a unit cube $U = [0,1]^3$ with the periodic boundary. The mesh size is $\Delta x = \Delta y = \Delta z = 0.01$ and the regularization parameter is chosen to be $\mu = 10^5$. We iterate 20000 steps with the timestep $\Delta t = 5 \times 10^{-8}$. The final partitions are shown in Figure 22-24. For $K = 4$, all four partitions are of the same shape under the periodic extended domain $[0,2]^3$ given by the so-called rhombic dodecahedra comprising of 12 identical rhombic faces. Figure 23(a) shows all the complete rhombic dodecahedra under the periodic extension of the unit cube. Figure 23(b) shows a more complete packing structure. For $K = 8$, we obtain two different types of shape and they are shown in Figure 24(a) and (b). Among eight partitions, two regions have the first type of shape, while the other six regions belong to the second type. The overall packing structure is shown in Figure 24(c). All the above results are comparable with the solution obtained in [8,39].

8 Conclusion

In this paper, we have developed a level set method for the Dirichlet k-partition problem. This provides a possible alternative to some popular numerical approaches such as [22,8,5,39]. By replacing the eigenfunction by the signed distance function in the level set formulation, we can reduce the nested minimization problem to a single minimization problem so that it can be easily solved by the gradient descent method. Following the idea in the variational level set formulation [40], we have realized that the resulting dynamics near the zero level set are similar to multiphase flow modeling. Even though both governing PDEs consist of a diffusion mechanism characterized by the $|\nabla\phi|$ term, the diffusion coefficient in our method is not predefined before the computation but has to be updated adaptively according to the level set function. To handle problems of a bounded domain, we can simply introduce another signed distance function ψ to the domain U. An extra penalty term involving ψ is added to the evolution equation to restrict the partition to stay inside the domain.

To improve the computational efficiency of the full level set approach, one can implement the PDE-based local level set method to concentrate the computational effort near the zero level set or can incorporate with a grid adaptive approach to put more grid points near the boundary of the partition. One might also develop a level set approach to these optimal partition problems on surfaces. We will leave these possible improvements and extensions to future investigations.

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

Acknowledgment

The work of Leung was supported in part by the Hong Kong RGC grant 16302819.

References

1. S.M. Allen and J.W. Cahn. A microscope theory for antiphase boundary motion and its application to antiphase domain coarsening. Acta Metall., 27:1085–1095, 1979.
2. E. Oudet B. Osting, C. White. Minimal Dirichlet energy partitions for graphs. SIAM Journal on Scientific Computing, 36(4):A1635–A1651, 2014.
3. W. Bao. Ground states and dynamics of rotating Bose-Einstein condensates. Modeling and Simulation in Science, Engineering and Technology, 2(9780817644895):215–255, 2007.
4. W. Bao and Q. Du. Computing the Ground State Solution of Bose–Einstein Condensates by a Normalized Gradient Flow. SIAM J. Sci. Comput., 25(5):1674–1697, 2004.
5. B. Bogosel. Efficient algorithm for large spectral partitions. arXiv:1705.08739, 2017.
6. B. Bogosel. Efficient algorithm for optimizing spectral partition. Applied Mathematics and Computation, 333:61–75, 2018.
7. B. Bogosel and V. Bonnaillie-Noel. Minimal partitions for p-norms of eigenvalues. Interfaces and Free Boundaries, 20(1):129, 2018.
8. B. Bourdin, D. Bucur, and D. Oudet. Optimal Partitions for Eigenvalues. SIAM J. Sci. Comput., 31(6):4100–4114, 2010.
9. F. Bozorgnia. Numerical algorithm for spatial segregation of competitive systems. SIAM J. Sci. Comput., 31(5):3946–3958, 2009.
10. F. Bozorgnia. Optimal partitions for first eigenvalues of the Laplace operator. Numerical Methods for Partial Differential Equations, 31(3):923–949, 2015.
11. D. Bucur, G. Buttazzo, and A. Henrot. Existence results for some optimal partition problems. Adv. Math. Sci. Appl., 8:571–579, 1998.
12. G. Buttazzo. Spectral optimization problems. Rev. Mat. Complut., 24:277–322, 2011.
13. L.A. Cafferelli and F.H. Lin. An optimal partition problem for eigenvalues. J. Sci. Comput., 31(1-2), 2007.
14. J.W. Cahn and J.E. Hilliard. Free energy of a nonuniform system. i. interfacial free energy. J. Chem. Phys., 28:258–267, 1958.
15. M. Conti, S. Terracini, and G. Verzini. Nehari's problem and competing species systems. Ann. I. H. Poincare Anal. Non Lineaire, 19(6):871–888, 2002.
16. M. Conti, S. Terracini, and G. Verzini. An optimal partition problem related to nonlinear eigenvalues. Journal of Functional Analysis, 198:160–196, 2003.
17. M. Conti, S. Terracini, and G. Verzini. On a class of optimal partition problems related to the fucik spectrum and to the monotonicity formula. Calc. Var., 22:45–72, 2005.
18. M. Conti, S. Terracini, and G. Verzini. A variational problem for the spatial segregation of rection-diffusion systems. Indiana Univ. Math. J., 54:779–815, 2005.
19. O. Cybulski, V. Babin, and R. Holyst. Minimization of the Renyi entropy production in the space-partitioning process. Phys. Rev. E., 71:046130, 2005.
20. O. Cybulski and R. Holyst. Three-dimensional space partition based on the first Laplacian eigenvalues in cells. Phys. Rev. E., 77:056101, 2008.
21. Q. Du and X. Feng. The phase field method for geometric moving interfaces and their numerical approximations. Geometric Partial Differential Equations, Handbook of Numerical Analysis, 21(I), 2020.
22. Q. Du and F. Lin. Numerical approximations of a norm-preserving gradient flow and applications to an optimal partition problem. Nonlinearity, 22(1):67–83, 2009.
23. C.M. Elliott and T. Ranner. A computational approach to an optimal partition problem on surfaces. Interfaces and Free Boundaries, 17(3):353, 2015.
24. B. Helffer. On Spectral Minimal Partitions: A Survey. Milan J. Math., 78:575–590, 2010.
25. B. Helffer, T. Hoffmann-Ostenhof, and S. Terracini. Nodel domains and spectral minimal partition. Ann. Inst. H. Poincare (Anal. Non Lineaire), 26:101–138, 2009.
26. G. S. Jiang and D. Peng. Weighted ENO schemes for Hamilton-Jacobi equations. SIAM J. Sci. Comput., 21:2126–2143, 2000.
27. C.-Y. Kao, S. Osher, and E. Yablonovitch. Maximizing band gaps in two dimensional photonic crystals by using level set methods. Applied Physics B: Lasers and Optics, 81:235–244, 2005.
28. A.V. Luminita and T.F. Chan. A new multiphase level set framework for image segmentation via the Mumford and Shah model. International Journal of Computer Vision, 50:271–293, 2002.
29. B. Merriman, J.K. Bence, and S. Osher. Diffusion generated motion by mean curvature. UCLA CAM Report, 92(18), 1992.
30. B. Merriman, J.K. Bence, and S. Osher. Motion of Multiple Junctions: A Level Set Approach. J. Comput. Phys., 112(2):334–363, 1994.
31. C. Min and F. Gibou. A second order accurate level set method on non-graded adaptive cartesian grids. J. Comput. Phys., 225:300–321, 2007.
32. S. J. Osher and R. P. Fedkiw. Level Set Methods and Dynamic Implicit Surfaces. Springer-Verlag, New York, 2003.
33. S. J. Osher and J. A. Sethian. Fronts propagating with curvature dependent speed: algorithms based on Hamilton-Jacobi formulations. J. Comput. Phys., 79:12–49, 1988.
34. B. Osting and Todd H. Reeb. Consistency of Dirichlet Partitions. SIAM J. Math. Anal., 49(5):4251–4274, 2017.
35. D. Peng, B. Merriman, S. Osher, H. K. Zhao, and M. Kang. A PDE-based fast local level set method. J. Comput. Phys., 155:410–438, 1999.
36. J. A. Sethian. Level set methods. Cambridge Univ. Press, second edition, 1999.
37. C. W. Shu and S. J. Osher. Efficient implementation of essentially non-oscillatory shock capturing schemes. J. Comput. Phys., 77:439–471, 1988.
38. G. Ungar, Y. Liu, X. Zheng, V. Percec, and WD Cho. Giant supermolecular liquid crystal lattice. Science, 299:1208–11, 2003.
39. D. Wang and B. Osting. A diffusion generated method for computing Dirichlet partitions. J. Comput. Appl. Math., 351:302–316, 2019.

40. H. K. Zhao, T. Chan, B. Merriman, and S. Osher. A variational level set approach to multiphase motion. J. Comput. Phys., 127(1):179–195, 1996.
41. P. Ziherl and R.D. Kamien. Soup froths and crystal structures. Phys. Rev. Lett., 85(16):3528, 2000.
42. D. Zosso and B. Osting. A minimal surface criterion for graph partitioning. Inverse Problems and Imaging, 10(4):1149–1180, 2016.