

Generalized MLS and LS-SVR for Allen–Cahn Equations on Spheres

Vahid Mohammadi^a, Mehdi Dehghan^b, Stefano De Marchi^{c*}

^a*Department of Mathematics, Faculty of Science,
Shahid Rajaee Teacher Training University, Tehran, 16785-163, Iran*

^b*Department of Applied Mathematics, Faculty of Mathematics and Computer Sciences,
Amirkabir University of Technology (Tehran Polytechnic), No. 424, Hafez Ave., 15914, Tehran, Iran*

^c*Department of Medicine and Padova Neuroscience Center, University of Padua, Italy
October 12, 2026*

Abstract. This work extends least-squares support vector regression (LS-SVR) based on the generalized moving least squares (GMLS) approximation to the numerical solution of partial differential equations on spheres. We show that the proposed approach constructs a kernel-based approximant in its dual form and establishes its connection with the associated reproducing kernel Hilbert space (RKHS). Furthermore, solving the PDE through the proposed LS-SVR framework is shown to be equivalent to solving a smoothing problem or a kernel ridge regression problem in the corresponding RKHS.

As applications, the method is employed to solve the Allen–Cahn equation and its conservative counterpart on spheres. An implicit first-order operator-splitting scheme is used to separate each model into linear and nonlinear subproblems. The linear subproblem is discretized by the backward differentiation formula and approximated at each time step using LS-SVR derived from a GMLS-based collocation formulation, while the nonlinear part is treated analytically. The resulting scheme is shown to admit a unique numerical solution at every time step. For the conservative Allen–Cahn model, a correction procedure is incorporated to preserve mass. Numerical experiments demonstrate the accuracy and effectiveness of the proposed framework.

Keywords: Generalized moving least squares approximation; Least-squares support vector regression; Reproducing Kernel Hilbert space and its smoothing form; Kernel ridge regression; Allen-Cahn equation; Conservative Allen-Cahn equation; PDEs on spheres, Time-splitting technique.

Mathematics Subject Classification (2020): 62J02, 33C55, 65Nxx, 41Axx.

1 Introduction

In this paper, we conduct a numerical investigation of two partial differential equations (PDEs) defined on the sphere, i.e., the Allen-Cahn (AC) equation and the conservative Allen-Cahn (CAC) equation, using the new approach.

*Corresponding author. *E-mail addresses:* vahid.mohammadi@sru.ac.ir (V. Mohammadi), mdehghan@aut.ac.ir, mdehghan.aut@gmail.com (M. Dehghan), stefano.demarchi@unipd.it (S. De Marchi).

The AC equation on sphere $\mathbb{S}^2$ reads as follows [7, 10]:

$$\frac{\partial u(\mathbf{x}, t)}{\partial t} = \Delta_{\mathbb{S}^2} u(\mathbf{x}, t) - \frac{F'(u(\mathbf{x}, t))}{\epsilon^2}, \quad (1.1)$$

in which $\mathbf{x} \in \mathbb{S}^2$, $t \in (0, T]$ and $\Delta_{\mathbb{S}^2}$ shows the Laplace-Beltrami operator, and $u(\mathbf{x}, t)$ is the phase-field variable (order parameter). Also, $F(u(\mathbf{x}, t)) = 0.25(u(\mathbf{x}, t)^2 - 1)^2$ shows the free-energy function. ϵ shows the gradient energy coefficient corresponding to the interfacial energy.

Besides, the CAC equation on sphere $\mathbb{S}^2$ is formulated by [32]:

$$\frac{\partial u(\mathbf{x}, t)}{\partial t} = \Delta_{\mathbb{S}^2} u(\mathbf{x}, t) - \frac{F'(u(\mathbf{x}, t))}{\epsilon^2} + \beta(t) \sqrt{F(u(\mathbf{x}, t))}, \quad (1.2)$$

in which $\sqrt{F(u(\mathbf{x}, t))} = 0.5|u(\mathbf{x}, t)^2 - 1|$, and

$$\beta(t) = \frac{\int_{\mathbb{S}^2} F'(u(\mathbf{x}, t)) \, d\mathbf{x}}{\epsilon^2 \int_{\mathbb{S}^2} \sqrt{F(u(\mathbf{x}, t))} \, d\mathbf{x}}. \quad (1.3)$$

The Lagrange multiplier $\beta(t)$ is used to satisfy the mass conservation property for variable $u(\mathbf{x}, t) = u$ in time [31, 32]. As a note, the denominator of $\beta(t)$ is positive. Indeed, $\sqrt{F(u)} = 0.5|u^2 - 1|$ is nonnegative for $u \in [-1, 1]$, and it vanishes only at phases $u = -1$ or $u = 1$. In the proposed diffuse-interface model, the transition layer between phases has finite thickness ϵ . Hence, the region where $-1 < u < 1$ has positive measure, making the integral strictly positive. The only case where this term could be zero occurs at the trivial homogeneous state, i.e., $u = -1$ or $u = 1$ everywhere, where the diffusion term is absent. Using an asymptotic analysis by considering the diffusion term gives a tanh profile as the solution showing a diffuse-interface of positive width always exists (cf. [42]).

In Appendix A, we show that the phase-field variable u of model (1.2) preserves mass in time.

Until now, numerous numerical methods have been developed for solving PDEs on surfaces (to predict natural phenomena, such as image inpainting [15, 35], pattern formations [20], and tumor growth [60]) such as (1.1) and (1.2), as the availability of exact solutions is impossible in most cases. In previous research works related to solving Eqs. (1.1) and (1.2) several numerical approaches were developed, including the classical finite difference (FD) method on the narrow band domain boundary, combined with an explicit time-splitting scheme of order one for solving the AC equation on surfaces [10]. Moreover, a CAC equation on surfaces was numerically investigated by utilizing the standard FD scheme in space and a first-order time-splitting idea in time [31]. Authors of [41] used an explicit time-splitting method of order one, combined with the restricted RBFs interpolant to solve the AC equation on 2-dimensional surfaces. Recently, a new numerical solution to the CAC equation on the cubic surface has been found in [29].

Overall, the numerical techniques applied to the surface PDE problems fall into two groups, i.e., the intrinsic and embedded narrow band methods (cf. [10, 17, 18, 20, 22, 31, 33, 34, 39, 41, 44, 53, 61]). The first group of these methods attempts to find an approximation to the differential operator in the original dimension. Of course, this approach faces difficulty with the inherent coordinate distortions or singularities arising in the metric terms of the differential operators (cf. [20]). In the second group, the differential operators are extended to the extrinsic coordinates, which avoids all coordinate singularities. Accordingly, any standard numerical methods such as the finite difference (FD) schemes on 3D regular nodes and the finite element method (FEM) on unstructured meshes can be implemented. Of course, these methods

increase the computational costs because they solve the problem in a dimension at least one greater than the original dimension of the main equation. Additionally, it needs to impose artificial boundary conditions, which may lead to the low accuracy of the applied numerical method. Hence, to avoid all the disadvantages of the mentioned approaches, a high-order numerical method applied to PDE problems on surfaces has been introduced in [20] that is based on the radial basis functions (RBFs) interpolation, and it integrates the benefits of both intrinsic and embedded narrow-band techniques (see also [41]). Besides, its recent developments, i.e., the RBF partition of unity (RBF-PU) method [40] and the RBF finite difference (RBF-FD) scheme [8, 53] have successfully been applied to the surface PDEs. In addition, the spherical basis functions (SBFs) have been applied to find the numerical solution of the PDEs on spheres (cf. [25, 26, 27, 39, 46, 62]).

Another possible numerical method on the spheres is to use zonal basis functions, which were investigated in [28, 45]. A zonal kernel is constructed by the restricted original positive definite radial kernel, say $\Phi = \phi(\|\cdot\|_2) : \mathbb{R}^{d+1} \rightarrow \mathbb{R}$ to $\mathbb{S}^d = \{\mathbf{x} \in \mathbb{R}^{d+1} : \|\mathbf{x}\|_2 = 1\}$ (d -dimensional sphere) such that for any $\mathbf{x}, \mathbf{y} \in \mathbb{S}^d$, we have $\|\mathbf{x} - \mathbf{y}\|_2 = \sqrt{2 - 2\mathbf{x}^T \mathbf{y}}$, and thus $\Phi(\mathbf{x}, \mathbf{y}) = \phi(\sqrt{2 - 2\mathbf{x}^T \mathbf{y}}) := \psi(\mathbf{x}^T \mathbf{y})$. Also, these basis functions have been used in the Petrov-Galerkin form of the diffusion equations on the unit sphere as the space of trial functions (cf. [39, 6]). In addition, in [30], by applying the local coordinates of the tangent plane, the differential operators on the two-dimensional surfaces have been approximated by the RBF-FD and the GMLS procedures.

On the other hand, one of the famous meshfree approaches is the generalized moving least squares (GMLS) approximation on spheres introduced in [38], known as a direct approximation. Comparing this method with the original moving least squares (MLS) technique presented in [56], the GMLS technique does not need to compute the derivatives (in many PDE problems on spheres, the surface gradient, and the Laplace-Beltrami operators should be computed) of the complicated shape functions. As will be observed, the approximate solution is found in the space spanned by the spherical harmonics of degree less than or equal to m (the maximum degree of spherical harmonics used here), and the Laplace-Beltrami operator is approximated easily due to the main property of the spherical harmonics (see Section 2). Besides, the proposed method does not depend on any mesh or background cell for the approximation, similar to the meshfree methods.

In this paper, we propose a new numerical procedure based on a supervised learning algorithm, i.e., the least-squares support vector regression (LS-SVR), considering the GMLS approximation [38] to solve both AC and CAC equations on $\mathbb{S}^2$. Moreover, we show that the constructed kernel function on the sphere induces a reproducing kernel Hilbert space (RKHS) in the dual representation of the approximate solution proposed in LS-SVR. Furthermore, we prove that the original minimization problem in the LS-SVR is equivalent to a smoothing problem or kernel ridge regression (KRR) in its corresponding RKHS. As far as we are aware, in the literature, this point of view related to the proposed supervised learning method is not considered at all. Hence, this is one of the main reasons for writing this research work. In what follows, we only mention some recent research works on this topic, where the proposed supervised learning algorithm has been applied to solve partial (ordinary) differential equations or integral equations.

The LS-SVR is a type of the least-squares support vector machine (LS-SVM), which is applied for function estimation [36, 50]. This approach is known as one of the main categories of supervised learning [55]. In [37, 50], this idea is successfully applied to solve both ordinary and partial differential equations. Also, the authors of [47] have extended the LS-SVR to solve the Volterra integral equations. In [48], this procedure has been developed to find the numerical solutions to Fredholm integral equations. The same approach, combined with spectral methods, is employed for solving numerically differential equations on unbounded domains [49]. In [1], a new algorithm based on a combination of the reduced-order LS-SVR with the isogeometric collocation method has been applied to solve the Cahn-Hilliard-Navier-Stokes (CHNS)

equations. See also [2], where the LS-SVR combined with a meshfree approximation has been developed to find the numerical solution of the fractional PDEs on diverse complex geometries.

Main contributions of this paper are itemized as follows:

- We show that the proposed supervised learning algorithm on spheres can be interpreted as a smoothing problem or a KRR in its corresponding RKHS.
- For the first time, a supervised learning algorithm, i.e., the LS-SVR based on the GMLS approximation is developed to solve numerically two nonlinear phase-field problems on spheres. The applied GMLS technique allows us to approximate the Laplace-Beltrami operator more easily than a mesh-dependent method, for example, methods developed in [10, 31].
- An implicit time-splitting approach combined with the proposed learning algorithm makes a simple fully discrete scheme for each phase-field model. Indeed, instead of solving a nonlinear system per time step, an implicit linear and a nonlinear problems are solved, where the nonlinear part is treated analytically. As a result, no nonlinear iterative solvers, such as the Newton-Raphson algorithm, are required to obtain the approximate solution for the fully implicit schemes of the studied mathematical models.
- By using the proposed supervised learning algorithm, the fully discrete problem related to the linear part of each equation is a minimization problem with constraints obtained from the collocation technique of the GMLS method at each time step. Accordingly, a dual form of this regression problem is equivalent to solving a linear system of algebraic equations with a positive definite coefficient matrix, resulting in a unique approximate intermediate solution.

Additionally, unlike the methods presented in [10, 31], the developed numerical technique does not require defining any quasi-Neumann boundary conditions on the narrow band domain boundary. On the other hand, the presented LS-SVR is meshfree because no triangulation or background cell is required for constructing the trial approximation via the GMLS procedure. Instead, it only depends on the number of nodes and their locations scattered on spheres. This makes it easy to execute on the PDE problems.

The other parts of the paper are organized as follows. In Section 2, the GMLS technique on spheres is explained briefly. Section 3 proposes the implementation of the LS-SVR to solve the PDE problem on spheres. In the next section, we first prove that the space of real-valued functions belonging to an approximation space is an RKHS. Then, we show how the proposed LS-SVR can be interpreted in the sense of a smoothing problem or a KRR in its associated RKHS. In Section 5, we first apply an implicit time-splitting scheme of order one to Eqs. (1.1) and (1.2) for time discretization. Then, we develop the proposed LS-SVR based on the GMLS approximation to derive the fully discrete schemes for both phase-field models. In the next section, the ability of the presented numerical procedure is demonstrated through the investigation of some simulation results. Finally, we give a conclusion in Section 7.

2 The GMLS procedure on spheres

In this section, a brief description of the GMLS technique on a 2D sphere, proposed in [38], is presented. As is already mentioned, the method is meshfree. This means that any background cell or triangulation on spheres is not necessary for the approximation. Instead, the scattered points, like $X = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$, on spheres are used to construct the trial approximation. However, a sequence of such sets should be quasi-uniform, meaning there exists a constant $c_{qu} > 1$ independent of X for all sets of this sequence, such that

the mesh-ratio $\rho_X := h_{X, \mathbb{S}^2}/q_X \leq c_{qu}$, where $h_{X, \mathbb{S}^2}$ and q_X are the fill distance and separation distance of points X , respectively (cf. [38]).

We are interested in finding an approximation to the Laplace-Beltrami operator $\Delta_{\mathbb{S}^2} := \mathbf{L}$, applied to the function $u \in C^{m+1}(\mathbb{S}^2)$ for some $m \geq 0$. To this end, the GMLS approximation to $\mathbf{L}u(\mathbf{x})$ based on the points X represented by $s_X^{\mathbf{L}}(\mathbf{x})$, where $\mathbf{x} \in \mathbb{S}^2$, is given by [38]

$$s_X^{\mathbf{L}}(\mathbf{x}) = \sum_{j \in I(\mathbf{x})} a_j^{\mathbf{L}}(\mathbf{x}) u_j = \mathbf{a}^{\mathbf{L}}(\mathbf{x})^T \mathbf{u}_X, \quad (2.1)$$

in which the coefficients $\{a_j^{\mathbf{L}}(\mathbf{x}) : j \in I(\mathbf{x})\}$ are computed through solving the weighted minimization problem over the spherical harmonics space with degree m or less, say $\pi_m(\mathbb{S}^2)$ with dimension $N(3, m)$ (cf. [38, Theorem 4.1]). Note that

$$N(3, 0) = 1, \quad N(3, m) = \frac{(2m+2)\Gamma(m+2)}{\Gamma(m+1)\Gamma(3)}, \quad m \geq 1.$$

Besides, $\mathbf{u}_X = [u(\mathbf{x}_i)]_{1 \leq i \leq N}$ denotes the value of $u(\mathbf{x})$ at points X . Besides, a set of indices, i.e., $I(\mathbf{x})$ is defined by

$$\{1 \leq j \leq N : \text{dist}(\mathbf{x}, \mathbf{x}_j) \leq \delta\},$$

where $\text{dist}(\mathbf{x}, \mathbf{y}) = \arccos(\mathbf{x}^T \mathbf{y})$ denotes the geodesic distance between two points $\mathbf{x}, \mathbf{y}$ on $\mathbb{S}^2$, and $0 < \delta < \pi$ is the radius of the spherical cap [38, 56]. By assuming the unisolvency of X on each spherical cap, we have

$$\mathbf{a}^{\mathbf{L}}(\mathbf{x}) = W(\mathbf{x})P \left(P^T W(\mathbf{x}) P \right)^{-1} \mathbf{L}Y(\mathbf{x}), \quad \mathbf{x} \in \mathbb{S}^2, \quad (2.2)$$

where P is a $|I(\mathbf{x})| \times N(3, m)$ matrix with elements $Y_{\ell k}(\mathbf{x}_j)$. $W(\mathbf{x})$ is a diagonal matrix of size $|I(\mathbf{x})|$ carrying the weights $w(\mathbf{x}, \mathbf{y}) := \phi\left(\frac{\text{dist}(\mathbf{x}, \mathbf{y})}{\delta}\right)$ on its diagonal, where $w : \mathbb{S}^2 \times \mathbb{S}^2 \rightarrow [0, \infty)$ is a continuous function. This function is typically compactly supported radial, i.e., $\phi : [0, \pi) \rightarrow [0, \infty)$ such that for $0 \leq r \leq 1$, $\phi(r) > 0$ and for $r > 1$, $\phi(r) = 0$ (of course, it is possible to utilize the weight function such that when $r \rightarrow 0$, $\phi \rightarrow \infty$) [38, 56]. Note that, in (2.2), the matrix $A(\mathbf{x}) := P^T W(\mathbf{x}) P$ is invertible because it is positive definite (cf. [38]).

Besides, in relation (2.2), $[\mathbf{L}Y_{\ell' k'}(\mathbf{x})]_{0 \leq \ell' \leq m, 1 \leq k' \leq N(2, \ell')} = \mathbf{L}Y(\mathbf{x}) \in \mathbb{R}^{N(3, m)}$ is the vector of the spherical harmonic functions with degree less than or equal m , where $\mathbf{L}$ acts on the vector $Y(\mathbf{x})$ that can be computed throughout the following relation (cf. [5, 11, 12]).

$$\mathbf{L}Y(\mathbf{x}) = -\ell(\ell+1)Y(\mathbf{x}). \quad (2.3)$$

Finally, an approximation to $\mathbf{L}u(\mathbf{x})$ at points X via the GMLS approximation is

$$\mathbf{L}u(\mathbf{x})|_X \approx \mathbf{a}^{\mathbf{L}^T}(\mathbf{x})|_X \mathbf{u}_X = L_X \mathbf{u}_X, \quad (2.4)$$

in which $L_X = [a_j^{\mathbf{L}}(\mathbf{x}_i)]_{1 \leq i, j \leq N}$ is a sparse differentiation matrix, and its elements are computed according to relation (2.2).

On the other hand, if we consider the point-evaluation functional, $\delta_{\mathbf{x}}$ instead of $\mathbf{L}$, then the GMLS

approximation to the function u represented by $s_X(\mathbf{x})$, where $\mathbf{x} \in \mathbb{S}^2$, is given by

$$s_X(\mathbf{x}) = \sum_{j \in I(\mathbf{x})} a_j(\mathbf{x}) u_j = \mathbf{a}^T(\mathbf{x}) \mathbf{u}_X, \quad (2.5)$$

which the coefficients $a_j(\mathbf{x})$ satisfy the delta-kronecker property, i.e., if $i = j$, $a_j(\mathbf{x}_i) = 1$, else-if $i \neq j$, $a_j(\mathbf{x}_i) = 0$, where $i, j = 1, \dots, N$. This means that the approximation (2.5) is an interpolant form. Also,

$$\mathbf{a}(\mathbf{x}) = W(\mathbf{x}) P \left(P^T W(\mathbf{x}) P \right)^{-1} Y(\mathbf{x}), \quad \mathbf{x} \in \mathbb{S}^2. \quad (2.6)$$

Also, $u(\mathbf{x})$ can be approximated at points X using the proposed GMLS idea as follows:

$$u(\mathbf{x})|_X \approx \mathbf{a}^T(\mathbf{x})|_X \mathbf{u}_X = A_X \mathbf{u}_X, \quad (2.7)$$

where $A_X = [a_j(\mathbf{x}_i)]_{1 \leq i, j \leq N}$ is a sparse matrix.

As a note, this idea is referred to as the direct approximation due to acting the linear functional L (or any other linear functionals) only on spherical harmonics, as claimed in [38] and seen here. The interested reader can also refer to [38] for finding most materials corresponding to the theoretical aspects of this approximation.

3 The LS-SVR on spheres

In this section, we extend the LS-SVR [36] to spheres for solving the PDE problems (or in the special case, the approximation problem) in the sense of the collocation form.

To start, assume that the linear differential equation is given by

$$\mathcal{L}u(\mathbf{x}) = f(\mathbf{x}), \quad \mathbf{x} \in \mathbb{S}^2, \quad (3.1)$$

in which $\mathcal{L}$ denotes a differential operator defined on $\mathbb{S}^2$, and u and f are sufficiently smooth functions. Now, consider the training (trial) data $\{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ on $\mathbb{S}^2$. Also, the solution to the regression problem is considered as follows:

$$\hat{u}(\mathbf{x}) = \mathbf{a}^T(\mathbf{x}) \mathbf{w} = \sum_{j=1}^N a_j(\mathbf{x}) w_j, \quad (3.2)$$

in which $\mathbf{w} = [w_1, \dots, w_N]^T$ and $\mathbf{a}^T(\mathbf{x}) = [a_1(\mathbf{x}), \dots, a_N(\mathbf{x})]$ contain the weights and the basis functions, respectively. To find the weights, we should solve the following minimization problem (cf. [36, 50]).

$$\min_{\mathbf{w}, \mathbf{e} \in \mathbb{R}^N} \frac{1}{2} \mathbf{w}^T \mathbf{w} + \frac{\gamma}{2} \mathbf{e}^T \mathbf{e}, \quad (3.3)$$

$$\mathcal{L}\hat{u}(\mathbf{x}_i) - f(\mathbf{x}_i) = e_i, \quad i = 1, \dots, N, \quad (3.4)$$

where $\mathbf{e} = [e_1, \dots, e_N]^T$ is the vector containing error value at each point $\mathbf{x}_i$ due to the proposed constraints in (3.4). Additionally, γ represents the positive constant, which is usually chosen arbitrarily by the user. However, in the next section, we show how this parameter can be chosen based on the theory of this approach. According to [50, Theorem 7.1] (see also [36]), where the Karush-Kuhn-Tucker (KKT) optimality conditions

are applied, we find the solution of minimization problem (3.3)-(3.3) as follows:

$$\mathbf{w} = A^T \boldsymbol{\alpha}, \quad \mathbf{e} = -\frac{1}{\gamma} \boldsymbol{\alpha}, \quad (3.5)$$

where $A =: A_L = [\mathcal{L}a_j(\mathbf{x}_i)]_{1 \leq i, j \leq N}$ thanks to the GMLS approximation given in the previous section. Also, $\boldsymbol{\alpha} =: \boldsymbol{\alpha}(\gamma)$ (i.e., the vector $\boldsymbol{\alpha}$ depends on parameter γ) is the vector of Lagrangian coefficients in the dual form of the minimization problem (3.3)-(3.4), which is computed by solving the following linear algebraic equations [50].

$$(AA^T + \frac{1}{\gamma} I_N) \boldsymbol{\alpha} = \mathbf{b}, \quad (3.6)$$

in which $\mathbf{b} = [f(\mathbf{x}_i)]_{1 \leq i \leq N}$, and it is known as a dual form of problem (3.3)-(3.4). Note also that, the matrix $B =: AA^T + \frac{1}{\gamma} I_N$ is clearly symmetric, i.e., $B^T = B$.

Theorem 3.1. *The problem (3.3)-(3.4) has a unique solution (3.5).*

Proof. It is sufficient to show that the coefficient matrix B in (3.6) is invertible. To do so, we show the matrix B is positive definite. Indeed, for any nonzero $\mathbf{x} \in \mathbb{R}^N$,

$$\mathbf{x}^T B \mathbf{x} = \mathbf{x}^T (AA^T + \frac{1}{\gamma} I_N) \mathbf{x} = \|A^T \mathbf{x}\|_2^2 + \frac{1}{\gamma} \|\mathbf{x}\|_2^2 > 0.$$

Hence, the matrix B is invertible, which gives a unique solution $\boldsymbol{\alpha}$ for any $\gamma > 0$ and also problem (3.3)-(3.4) according to (3.5). $\square$

The following lemma shows the formulation of approximate solution (3.2) in the dual form.

Lemma 3.2. *For any $\gamma > 0$, the dual representation of the approximate solution (3.2) is given by:*

$$\hat{u}(\mathbf{x}) = \sum_{j=1}^N K(\mathbf{x}, \mathbf{x}_j) \alpha_j = \mathbf{k}^T(\mathbf{x}) \boldsymbol{\alpha} := \hat{u}_\gamma(\mathbf{x}), \quad \mathbf{x} \in \mathbb{S}^2, \quad (3.7)$$

where $K(\cdot, \cdot)$ is the kernel function defined by $K(\mathbf{x}, \mathbf{x}_j) =: \langle \mathbf{a}(\mathbf{x}), \mathcal{L}\mathbf{a}(\mathbf{x}_j) \rangle$ for $j = 1, \dots, N$, and $\langle \cdot, \cdot \rangle$ denotes the usual inner-product between two vectors. Besides, $\mathbf{k}^T(\mathbf{x}) = [K(\mathbf{x}, \mathbf{x}_1) \dots K(\mathbf{x}, \mathbf{x}_N)]$ and $\boldsymbol{\alpha} = [\alpha_1 \dots \alpha_N]$.

Proof. According to the first relation in (3.5), the approximate solution (3.2) can be written as follows:

$$\begin{aligned} \hat{u}(\mathbf{x}) &= \mathbf{a}^T(\mathbf{x}) \mathbf{w} \\ &= \mathbf{a}^T(\mathbf{x}) A^T \boldsymbol{\alpha} \\ &= \mathbf{a}^T(\mathbf{x}) [\mathcal{L}\mathbf{a}(\mathbf{x}_1) \dots \mathcal{L}\mathbf{a}(\mathbf{x}_N)] \boldsymbol{\alpha} \\ &= [\mathbf{a}^T(\mathbf{x}) \mathcal{L}\mathbf{a}(\mathbf{x}_1) \dots \mathbf{a}^T(\mathbf{x}) \mathcal{L}\mathbf{a}(\mathbf{x}_N)] \boldsymbol{\alpha} \\ &= [\langle \mathbf{a}(\mathbf{x}), \mathcal{L}\mathbf{a}(\mathbf{x}_1) \rangle \dots \langle \mathbf{a}(\mathbf{x}), \mathcal{L}\mathbf{a}(\mathbf{x}_N) \rangle] \boldsymbol{\alpha}. \end{aligned} \quad (3.8)$$

Now, by defining $\langle \mathbf{a}(\mathbf{x}), \mathcal{L}\mathbf{a}(\mathbf{x}_j) \rangle := K(\mathbf{x}, \mathbf{x}_j)$, $j = 1, \dots, N$, the result is concluded. $\square$

4 An RKHS

Suppose in (3.1) $\mathcal{L} = \text{id}$, that is the identity operator. So, the constructed kernel in the previous section allows us to define a new Hilbert space with reproducing kernel. Accordingly, the new function, i.e.,

$$K(\mathbf{x}, \mathbf{y}) := \sum_{i=1}^N a_i(\mathbf{x})a_i(\mathbf{y}) = \langle \mathbf{a}(\mathbf{x}), \mathbf{a}(\mathbf{y}) \rangle_{\mathbb{R}^N}, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{S}^2, \quad (4.1)$$

can be constructed.

Now, we assume that for any distinct points $\mathbf{x}_1, \dots, \mathbf{x}_N$ on $\mathbb{S}^2$, the matrix A has full rank. Accordingly, the matrix $K := AA^T$ is positive definite. Hence, based on [3], there exists a reproducing kernel Hilbert space (RKHS), where K is reproducing kernel.

To show this fact, let $\mathbf{a} : \mathbb{S}^2 \rightarrow \mathbb{R}^N$ be a feature map into a Hilbert space $(\mathbb{R}^N, \langle \cdot, \cdot \rangle_{\mathbb{R}^N})$. We consider the following Hilbert space.

$$H =: H(\mathbb{S}^2) := \{f_{\mathbf{w}} : \mathbb{S}^2 \rightarrow \mathbb{R} : \exists \mathbf{w} \in \mathbb{R}^N \text{ with } f_{\mathbf{w}}(\mathbf{x}) = \langle \mathbf{w}, \mathbf{a}(\mathbf{x}) \rangle_{\mathbb{R}^N}, \forall \mathbf{x} \in \mathbb{S}^2\}, \quad (4.2)$$

with the inner product defined by:

$$\forall f_{\mathbf{w}}, f_{\mathbf{v}} \in H, \quad \langle f_{\mathbf{w}}, f_{\mathbf{v}} \rangle_H := \langle \mathbf{w}, \mathbf{v} \rangle_{\mathbb{R}^N}. \quad (4.3)$$

Besides, its associated norm is

$$\|f_{\mathbf{w}}\|_H = \|\mathbf{w}\|_{\mathbb{R}^N}. \quad (4.4)$$

Based on the above assumption and definition, the map $T : \mathbb{R}^N \rightarrow H$ with $T(\mathbf{w}) = f_{\mathbf{w}}$ for all $\mathbf{w} \in \mathbb{R}^N$, where $f_{\mathbf{w}}(\mathbf{x}) = \langle \mathbf{w}, \mathbf{a}(\mathbf{x}) \rangle_{\mathbb{R}^N}$, $\forall \mathbf{x} \in \mathbb{S}^2$, is an isometric isomorphism. This confirms the well-definedness of RKHS inner product.

We are now interested to show H is a reproducing kernel Hilbert space (RKHS), where its kernel is given by (4.1).

Theorem 4.1. *The function (4.1) is reproducing kernel for H .*

Proof. Based on [13, Definition 13.1], we show $K(\cdot, \mathbf{x}) : \mathbb{S}^2 \rightarrow \mathbb{R}$ belongs to H for all $\mathbf{x} \in \mathbb{S}^2$, and $f_{\mathbf{w}}(\mathbf{x}) = \langle f_{\mathbf{w}}, K(\cdot, \mathbf{x}) \rangle_H$ for all $f_{\mathbf{w}} \in H$ and all $\mathbf{x} \in \mathbb{S}^2$.

For the first claim, assume that $\mathbf{x} \in \mathbb{S}^2$ is arbitrary and fixed. For any $\mathbf{x}' \in \mathbb{S}^2$, $K(\mathbf{x}', \mathbf{x}) = \langle \mathbf{a}(\mathbf{x}'), \mathbf{a}(\mathbf{x}) \rangle_{\mathbb{R}^N}$. On the other hand, if we consider $\mathbf{w} = \mathbf{a}(\mathbf{x})$, then for all $\mathbf{x}' \in \mathbb{S}^2$

$$\begin{aligned} f_{\mathbf{w}}(\mathbf{x}') &= \langle \mathbf{w}, \mathbf{a}(\mathbf{x}') \rangle_{\mathbb{R}^N} \\ &= \langle \mathbf{a}(\mathbf{x}), \mathbf{a}(\mathbf{x}') \rangle_{\mathbb{R}^N} \\ &= K(\mathbf{x}', \mathbf{x}), \end{aligned} \quad (4.5)$$

which shows $k(\cdot, \mathbf{x}) \in H$.

For the second claim, from the above, we know that $f_{\mathbf{a}(\mathbf{x})}(\cdot) = K(\cdot, \mathbf{x})$. Hence, for any $f_{\mathbf{w}} \in H$

$$\begin{aligned}\langle f_{\mathbf{w}}, K(\cdot, \mathbf{x}) \rangle_H &= \langle f_{\mathbf{w}}, f_{\mathbf{a}(\mathbf{x})} \rangle_H \\ &= \langle \mathbf{w}, \mathbf{a}(\mathbf{x}) \rangle_{\mathbb{R}^N} \\ &= f_{\mathbf{w}}(\mathbf{x}),\end{aligned}\tag{4.6}$$

which completes the proof. $\square$

There are other properties related to reproducing kernel which are given in the following proposition.

Proposition 4.2. *For a Hilbert space of functions with reproducing kernel K such as H , we have*

- $\forall \mathbf{x}, \mathbf{y} \in \mathbb{S}^2, K(\mathbf{x}, \mathbf{y}) = \langle K(\cdot, \mathbf{x}), K(\cdot, \mathbf{y}) \rangle_H$.
- $\forall \mathbf{x}, \mathbf{y} \in \mathbb{S}^2, K(\mathbf{x}, \mathbf{y}) = K(\mathbf{y}, \mathbf{x})$.
- if $\lim_{n \rightarrow \infty} \|f - f_n\|_H = 0$, then $\lim_{n \rightarrow \infty} |f(\mathbf{x}) - f_n(\mathbf{x})| = 0$, for all $\mathbf{x} \in \mathbb{S}^2$.

Proof. The first property can be obtained directly from Theorem 4.1. The second property is concluded from (4.1). To prove the last property, assume that $\mathbf{x} \in \mathbb{S}^2$ is arbitrary. Using the famous Cauchy-Schwarz inequality, we can write

$$\begin{aligned}|f(\mathbf{x}) - f_n(\mathbf{x})| &= |\langle f - f_n, K(\cdot, \mathbf{x}) \rangle_H| \\ &\leq \|f - f_n\|_H \|K(\cdot, \mathbf{x})\|_H \\ &= \|f - f_n\|_H \|\mathbf{a}(\mathbf{x})\|_{\mathbb{R}^N}.\end{aligned}\tag{4.7}$$

Accordingly, if $n \rightarrow \infty$, then the desired result can be obtained, and it completes the proof. $\square$

We now consider the following pre-Hilbert space, namely H_{pre} .

$$H_{pre} := \left\{ \sum_{i=1}^N \alpha_i K(\cdot, \mathbf{x}_i) : N \in \mathbb{N}, \alpha_1, \dots, \alpha_N \in \mathbb{R}, \mathbf{x}_1, \dots, \mathbf{x}_N \in \mathbb{S}^2 \right\}.\tag{4.8}$$

Proposition 4.3. *Let $\mathbf{a} : \mathbb{S}^2 \rightarrow \mathbb{R}^N$ be a feature map into a Hilbert space $(\mathbb{R}^N, \langle \cdot, \cdot \rangle_{\mathbb{R}^N})$. Then, we have*

- The pre-Hilbert space H_{pre} is dense in H .
- $\forall f, g \in H_{pre}, \langle f, g \rangle_H = \sum_{j=1}^N \sum_{m=1}^N \alpha_j \beta_m k(\mathbf{x}_j, \mathbf{x}_m) = \boldsymbol{\alpha}^T K \boldsymbol{\beta}$.
- $\forall f \in H_{pre}, \|f\|_H^2 = \sum_{j=1}^N \sum_{m=1}^N \alpha_j \alpha_m k(\mathbf{x}_j, \mathbf{x}_m) = \boldsymbol{\alpha}^T K \boldsymbol{\alpha}$.

Proof. The first part can be proved in the same manner presented in [54, Theorem 4.21]. For the second part, for any $f, g \in H_{pre}$, we can write

$$f(\cdot) = \sum_{j=1}^N \alpha_j K(\cdot, \mathbf{x}_j), \quad g(\cdot) = \sum_{m=1}^N \beta_m K(\cdot, \mathbf{x}_m).$$

From the reproducing property of H , we can write

$$\begin{aligned}
\langle f, g \rangle_H &= \left\langle \sum_{j=1}^N \alpha_j K(\cdot, \mathbf{x}_j), \sum_{m=1}^N \beta_m K(\cdot, \mathbf{x}_m) \right\rangle_H \\
&= \sum_{j=1}^N \sum_{m=1}^N \alpha_j \beta_m K(\mathbf{x}_j, \mathbf{x}_m) \\
&= \boldsymbol{\alpha}^T K \boldsymbol{\beta}.
\end{aligned} \tag{4.9}$$

Finally, the last part is obtained from the second part, where g is replaced by f . $\square$

Remark 4.4. *It should be noted that if we want to have an infinite-dimensional space, the Hilbert space ℓ_2 is used instead of $\mathbb{R}^N$, and so $N = \infty$.*

We now show the proposed kernel-based approximant, i.e.,

$$\hat{f}(\mathbf{x}) = \sum_{j=1}^N K(\mathbf{x}, \mathbf{x}_j) \alpha_j, \quad \mathbf{x} \in \mathbb{S}^2, \tag{4.10}$$

is an interpolant to a function f , where $X = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ on $\mathbb{S}^2$ and the values $f(\mathbf{x}_1), \dots, f(\mathbf{x}_N)$ are given.

Proposition 4.5. *Assume that for any distinct points $\mathbf{x}_1, \dots, \mathbf{x}_N$ on $\mathbb{S}^2$, the matrix A has no loss rank. The approximation (4.10) is interpolant for given data $\{(\mathbf{x}_i, f(\mathbf{x}_i))\}_{i=1}^N$.*

Proof. According to (4.1), for any $\mathbf{x} \in \mathbb{S}^2$, we can write

$$\begin{aligned}
\hat{f}(\mathbf{x}) &= \mathbf{k}^T(\mathbf{x}) \boldsymbol{\alpha} \\
&= \mathbf{k}^T(\mathbf{x}) K^{-1} \mathbf{f} \\
&= \boldsymbol{\ell}^T(\mathbf{x}) \mathbf{f},
\end{aligned} \tag{4.11}$$

where in the second line of this equality, we have used $K\boldsymbol{\alpha} = \mathbf{f}$, and the fact that A has no loss rank. Due to the definition of $\mathbf{k}^T(\mathbf{x})$ and $\mathbf{k}^T(\mathbf{x}) K^{-1} := \boldsymbol{\ell}^T(\mathbf{x}) = [\ell_1(\mathbf{x}) \dots \ell_N(\mathbf{x})]$, it is clear that $\ell_j(\mathbf{x}_i) = 1$ if $i = j$, and $\ell_j(\mathbf{x}_i) = 0$ if $i \neq j$, which gives the desired results. $\square$

Theorem 4.6. *The proposed kernel based interpolant $\hat{f}$ to a function f in H is a minimum norm interpolant. Indeed*

$$\|\hat{f}\|_H = \min\{\|s\|_H : s \in H, \text{ with } s(\mathbf{x}_i) = f(\mathbf{x}_i), i = 1, \dots, N\}. \tag{4.12}$$

Proof. The proof follows from the proof of [13, Theorem 18.1]. $\square$

4.1 Smoothing in RKHS

We start with the following definition similar to [58].

Definition 4.7. Suppose that the RKHS (4.2) is given. For given data $X = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ on $\mathbb{S}^2$ and function values $\mathbf{f} = \{f(\mathbf{x}_1), \dots, f(\mathbf{x}_N)\}$, where $f(\mathbf{x}_i) \in \mathbb{R}$, $i = 1, \dots, N$, we define

$$F(\hat{f}) := \sum_{j=1}^N (f(\mathbf{x}_j) - \hat{f}(\mathbf{x}_j))^2 + \frac{1}{\gamma} \|\hat{f}\|_H^2, \quad (4.13)$$

where $\frac{1}{\gamma}$ is smoothing parameter. Then, we find the solutions $\hat{f}_\gamma \in H$ to

$$\min_{\hat{f} \in H} F(\hat{f}). \quad (4.14)$$

The following lemma shows that the unique solution of (4.14) is the same with problem (3.3)-(3.4).

Lemma 4.8. The unique solution of problem (4.14) with target function $f \in H$ is $\hat{f}_\gamma$ given by

$$\hat{f}_\gamma = \sum_{j=1}^N K(\cdot, \mathbf{x}_j) \alpha_j, \quad (4.15)$$

where α can be computed by solving the following linear system.

$$(K + \frac{1}{\gamma} I_N) \alpha = \mathbf{f}. \quad (4.16)$$

Proof. First, it should be noted that K is assumed to be positive definite. So, the proposed functional (4.13), which can be written with respect to the vector α , i.e., $F(\alpha) =: \frac{1}{\gamma} \alpha^T K \alpha + (K \alpha - \mathbf{f})^T (K \alpha - \mathbf{f})$, is convex. Hence, it has a unique minimum. By differentiating the functional with respect to α , we have

$$\nabla_\alpha F(\alpha) = \frac{2}{\gamma} K \alpha + 2K^T (K \alpha - \mathbf{f}). \quad (4.17)$$

Considering $\nabla_\alpha F(\alpha) = \mathbf{0}$ (the necessary and sufficient condition for finding a unique minimum), we get

$$(\frac{2}{\gamma} K + 2K^T K) \alpha = 2K^T \mathbf{f}. \quad (4.18)$$

Since K is a symmetric positive definite matrix, multiplying both sides of (4.18) by the left inverse K and dividing them to factor 2, we achieve

$$(K + \frac{1}{\gamma} I_N) \alpha = \mathbf{f}, \quad (4.19)$$

which completes the proof. $\square$

We now give the following proposition showing an upper bound for the approximation $\hat{f}_\gamma$ at points X .

Proposition 4.9. Suppose $\hat{f}_\gamma \in H$ is a solution of (4.14). Also, assume that $f(\mathbf{x}_i)$ for $i = 1, \dots, N$ are given, where $f \in H$. Besides, suppose that there exists a function $\hat{f}_\infty \in H$ with $\sum_{j=1}^N (f(\mathbf{x}_j) - \hat{f}_\infty(\mathbf{x}_j))^2 = 0$

and $\|\hat{f}_\infty\|_H^2 \leq \|f\|_H^2$. Then, we have

$$\begin{aligned} \|\hat{f}_\gamma\|_H^2 &\leq \|f\|_H^2 \\ |f(\mathbf{x}_i) - \hat{f}_\gamma(\mathbf{x}_i)| &\leq \gamma^{-1/2} \|f\|_H, \quad i = 1, \dots, N. \end{aligned} \quad (4.20)$$

Proof. For the proof, see [58, Proposition 3.1]. $\square$

In kernel-based methods, it is most common to represent the above upper bounds in terms of the fill distance of X . Hence, if we set $\gamma \geq h_{X, \mathbb{S}^2}^{-\beta}$, where β can be chosen to deal with a desired convergence rate, then the above error bounds at points X obey

$$|f(\mathbf{x}_i) - \hat{f}_\gamma(\mathbf{x}_i)| \leq h_{X, \mathbb{S}^2}^{\beta/2} \|f\|_H, \quad i = 1, \dots, N. \quad (4.21)$$

Note that, to deal with a proper rate of convergence, it is recommended to set $\beta \geq 2$ that is an appropriate way to choose parameter γ . Of course, there are other algorithms to select this parameter such as generalized cross validation (cf. [14]).

4.2 Connection to the KRR

We can utilize the kernel-based approximant (3.7) to solve the main problem (3.1) via the collocation technique directly. Hence, we have

$$\sum_{j=1}^N \mathcal{L}k(\mathbf{x}_i, \mathbf{x}_j) \alpha_j = f(\mathbf{x}_i), \quad i = 1, \dots, N, \quad (4.22)$$

in which $\mathcal{L}k(\mathbf{x}_i, \mathbf{x}_j) = \langle \mathcal{L}\mathbf{a}(\mathbf{x}_i), \mathcal{L}\mathbf{a}(\mathbf{x}_j) \rangle$, $i, j = 1, \dots, N$. So, Eq. (4.22) can be written as:

$$AA^T \boldsymbol{\alpha} = \mathbf{b}. \quad (4.23)$$

Now, assume that A has full rank. Let $\mathbf{w}^*$ be the exact solution to the linear system $A\mathbf{z} = \mathbf{b}$. So, the linear system (4.23) is known as the normal equations of the following least squares problem (cf. [51, Chapter 8]):

$$\min \|\mathbf{w}^* - A^T \boldsymbol{\alpha}\|_2. \quad (4.24)$$

By defining $A^T \boldsymbol{\alpha} = \mathbf{z}$, problem (4.24) finds the solution vector $\mathbf{z}$ that is closest to $\mathbf{w}^*$ in two-norm sense.

Lemma 4.10. *Assume that A has full rank. The solution of problem (4.24) is given by:*

$$\mathbf{w} = A^T \boldsymbol{\alpha}^*, \quad (4.25)$$

where

$$\boldsymbol{\alpha}^* = \sum_{i=1}^N \frac{\mathbf{v}_i^T \mathbf{b}}{\sigma_i^2} \mathbf{v}_i. \quad (4.26)$$

in which $\mathbf{v}_i$ is the i -th column of the orthogonal matrix V from the singular value decomposition (SVD) of matrix A^T . Besides, for $i = 1, \dots, N$, σ_i are the singular values of A^T .

Proof. As mentioned earlier, assume that $\mathbf{w}^*$ is the exact solution to the linear system $A\mathbf{z} = \mathbf{b}$. Using the SVD of matrix A^T , we can write

$$\begin{aligned}
\|\mathbf{w}^* - A^T \boldsymbol{\alpha}\|_2^2 &= \left\| \mathbf{w}^* - \left(U \sum V^T \right) \boldsymbol{\alpha} \right\|_2^2 \\
&= \left\| U^T \left(\mathbf{w}^* - \left(U \sum V^T \right) \boldsymbol{\alpha} \right) \right\|_2^2 \\
&= \left\| U^T \mathbf{w}^* - \underbrace{\sum (V^T \boldsymbol{\alpha})}_{:= \mathbf{y}} \right\|_2^2 \\
&= \sum_{i=1}^N (\sigma_i y_i - (\mathbf{u}_i^T \mathbf{w}^*))^2.
\end{aligned} \tag{4.27}$$

Accordingly, to deal with the minimum of problem (4.24), we should have

$$y_i = \frac{\mathbf{u}_i^T \mathbf{w}^*}{\sigma_i}, \quad i = 1, \dots, N. \tag{4.28}$$

Since $\mathbf{y} = V^T \boldsymbol{\alpha}$, we get $\boldsymbol{\alpha}^* = V \mathbf{y}$. And, considering relation (4.27) gives

$$\boldsymbol{\alpha}^* = \sum_{i=1}^N \frac{\mathbf{u}_i^T \mathbf{w}^*}{\sigma_i} \mathbf{v}_i. \tag{4.29}$$

On the other hand, applying the SVD to A , we have $V \Sigma^T U^T \mathbf{w}^* = \mathbf{b}$, which yields $\Sigma^T U^T \mathbf{w}^* = V^T \mathbf{b}$. Finally, we get $U^T \mathbf{w}^* = (\Sigma^T)^{-1} V^T \mathbf{b}$, and replacing it into (4.29), the desired result is concluded. $\square$

On the other hand, it is possible to find an upper bound for $\|\mathbf{w}^* - \mathbf{w}\|_2$ in terms of fill distance of points X based on Schaback's theory [52] and some details in [38]. To do so, assume that $\mathbf{w}^*$ is the vector of exact values so that $A\mathbf{w}^* = \mathbf{b}$. If $\mathbf{w}^*$ is replaced by the vector of approximate values, $\tilde{\mathbf{w}}$, then according to problem (4.24), we have

$$\|\mathbf{w}^* - \mathbf{w}\|_2 = \min \|\mathbf{w}^* - \mathbf{z}\|_2 \leq \|\mathbf{w}^* - \tilde{\mathbf{w}}\|_2. \tag{4.30}$$

Now, we can find the upper bound for $\|\mathbf{w}^* - \mathbf{w}\|_2$ based on [38], yielding

$$\|\mathbf{w}^* - \tilde{\mathbf{w}}\|_2 \leq CC_s(A)(1 + K(A))h_{X, \mathbb{S}^2}^{m+1} \|\mathbf{w}^*\|_{C^{m+1}(\mathbb{S}^2)}, \tag{4.31}$$

in which C is a positive constant, $C_s(A)$ is the stability constant, $K(A)$ is equal to 1 if $\tilde{\mathbf{w}}$ is calculated via the minimization of the residual $\|A\mathbf{w}^* - \mathbf{b}\|_2$, or to 0 if $\mathbf{b}$ is in the range of A .

Hence, we can finally find the following upper bound.

$$\|\mathbf{w}^* - \mathbf{w}\|_2 \leq CC_s(A)(1 + K(A))h_{X, \mathbb{S}^2}^{m+1} \|\mathbf{w}^*\|_{C^{m+1}(\mathbb{S}^2)}. \tag{4.32}$$

Theorem 4.11. *Under the above assumptions for matrix K , the regularized least squares approximation problem, i.e.,*

$$\min_{\boldsymbol{\alpha}} \frac{1}{2} \boldsymbol{\alpha}^T K \boldsymbol{\alpha} + \frac{\gamma}{2} (K \boldsymbol{\alpha} - \mathbf{b})^T (K \boldsymbol{\alpha} - \mathbf{b}). \tag{4.33}$$

has the same solution with the problem (3.3)-(3.4).

Proof. The proof can be done in the same manner presented in Lemma 4.8. $\square$

Corollary 1. Assume that α is the solution of linear system (4.19). If $\gamma \rightarrow +\infty$, then $\alpha \rightarrow \alpha^*$, where α^* is obtained as (4.26).

Proof. If $A^T = U\Sigma V^T$ is the SVD of A^T , then (4.19) changes to

$$\begin{aligned} \left(V\Sigma^T U^T U \Sigma V^T + \frac{1}{\gamma} I_N \right) \alpha &= \mathbf{b} \xRightarrow{U^T U = I_N} \left(V\Sigma^T \Sigma V^T + \frac{1}{\gamma} I_N \right) \alpha = \mathbf{b} \\ \Rightarrow \left(V^T V \Sigma^T \Sigma V^T + \frac{1}{\gamma} V^T I_N \right) \alpha &= V^T \mathbf{b} \xRightarrow{V^T V = I_N} \left(\Sigma^T \Sigma + \frac{1}{\gamma} I_N \right) (V^T \alpha) = V^T \mathbf{b}. \end{aligned} \quad (4.34)$$

Hence, we have

$$\alpha = \sum_{i=1}^N \frac{\mathbf{v}_i^T \mathbf{b}}{\sigma_i^2 + \frac{1}{\gamma}} \mathbf{v}_i. \quad (4.35)$$

Now, if $\gamma \rightarrow +\infty$, the desired result is obtained. $\square$

Remark 4.12. The problem (4.33) is known as ridge regression, and it can be observed as the diagonal stabilization/regularization of the usual approximation linear system (4.23).

Proposition 4.13. Assume that $\mathcal{L} = \text{Id}$ in problem (3.1). Suppose also that all assumptions proposed in Theorem 4.11 are satisfied. If $\gamma = +\infty$, then the approximate solution (3.8) is interpolant for given data $\{(\mathbf{x}_i, f(\mathbf{x}_i))\}_{i=1}^N$ (see, e.g., [13, 14]).

Proof. According to (3.7) and (4.19), for any $\mathbf{x} \in \mathbb{S}^2$, we can write

$$\begin{aligned} \hat{f}_\gamma(\mathbf{x}) &= \mathbf{k}^T(\mathbf{x}) \left(K + \frac{1}{\gamma} I_N \right)^{-1} \mathbf{b} \\ &= \mathbf{k}^T(\mathbf{x}) K^{-1} \left(K + \frac{1}{\gamma} I_N \right)^{-1} K \mathbf{b} \\ &= \boldsymbol{\ell}^T(\mathbf{x}) \left(K + \frac{1}{\gamma} I_N \right)^{-1} K \mathbf{b}, \end{aligned} \quad (4.36)$$

where in the second line of this equality, we have used the push-through identity (cf. [14]), and $\mathbf{k}^T(\mathbf{x}) K^{-1} := \boldsymbol{\ell}^T(\mathbf{x}) = [\ell_1(\mathbf{x}) \dots \ell_N(\mathbf{x})]$. Due to the definition of $\mathbf{k}^T(\mathbf{x})$, it is clear that $\ell_j(\mathbf{x}_i) = 1$ if $i = j$, and $\ell_j(\mathbf{x}_i) = 0$ if $i \neq j$. Considering $\gamma = +\infty$ and using the fact that $\mathbf{b} = [f(\mathbf{x}_i)]_{1 \leq i \leq N}$, we thus have

$$\hat{f}_\infty(\mathbf{x}) = \boldsymbol{\ell}^T(\mathbf{x}) \mathbf{b}, \quad (4.37)$$

which gives the desired results. $\square$

Remark 4.14. If we assume that $f(\mathbf{x}_i)$ is contaminated by error e_i , i.e., $f(\mathbf{x}_i) + e_i$ for $i = 1, \dots, N$, then the new right hand side is given by $\mathbf{b} = [f(\mathbf{x}_i) + e_i]_{1 \leq i \leq N}$. These can be viewed as the noisy data. According

to the approximation (4.11), the term $\tilde{\mathbf{b}} =: (K + \frac{1}{\gamma}I_N)^{-1}K\mathbf{b}$ refers to the smoothed data or filtered data. Hence, relation (4.11) can be known as the kernel interpolation for these filtered data. In this case, we have

$$\hat{f}(\mathbf{x}) = \ell^T(\mathbf{x})\tilde{\mathbf{b}}, \quad (4.38)$$

5 Application to two phase field models on $\mathbb{S}^2$

In this section, we derive the new fully discrete schemes for both phase-field models, i.e., Eqs. (1.1) and (1.2). We begin by applying an implicit time-splitting approach of accuracy order one to discretize each equation in time. Then, we develop the presented LS-SVR for the spatial discretization of the obtained semi-discrete equations.

5.1 An implicit time-splitting technique

The sequential time-splitting idea [61] on Eq. (1.1) and a type of first-order operator splitting approach [31] on (1.2) have been applied (see also [41]).

To start, suppose that the time interval is $[0, T]$, where T is the final time. To partition this interval, consider a uniform time step $\Delta t = \frac{T}{K}$ with the number of sub-intervals to be K . So, $[0, T] = \bigcup_{n=1}^K [t_{n-1}, t_n]$, $t_0 = 0$, $t_n = n\Delta t$ for $n = 1, \dots, K-1$ and $t_K = K\Delta t = T$. Hence, the proposed sequential-implicit splitting method could be executed for Eq. (1.1) as follows:

$$(\text{id} - \Delta t \mathbf{L})u^{n*} = u^n, \quad (5.1)$$

$$u^{n+1} = \frac{u^{n*}}{\sqrt{\exp\left(-\frac{2\Delta t}{\epsilon^2}\right) + (u^{n*})^2 \left(1 - \exp\left(-\frac{2\Delta t}{\epsilon^2}\right)\right)}}, \quad n = 0, \dots, K-1. \quad (5.2)$$

In the above semi-discrete equations, u^n shows the approximate value of the exact solution $u(\cdot, t_n)$, and u^{n*} is the approximate intermediate solution.

Also, by applying a type of time-splitting technique utilized in [31] to Eq. (1.2), we get the following semi-discrete equations:

$$(\text{id} - \Delta t \mathbf{L})u^{n*} = u^n, \quad (5.3)$$

$$u^{n+1**} = \frac{u^{n*}}{\sqrt{\exp\left(-\frac{2\Delta t}{\epsilon^2}\right) + (u^{n*})^2 \left(1 - \exp\left(-\frac{2\Delta t}{\epsilon^2}\right)\right)}}, \quad (5.4)$$

$$u^{n+1} = u^{n+1**} + \Delta t \beta^{n+1**} \sqrt{F(u^{n+1**})}, \quad n = 0, \dots, K-1, \quad (5.5)$$

in which u^{n+1**} is known as the second approximate intermediate solution, and

$$\beta^{n+1**} = \frac{\int_{\mathbb{S}^2} (u^0 - u^{n+1**}) \, d\mathbf{x}}{\Delta t \int_{\mathbb{S}^2} \sqrt{F(u^{n+1**})} \, d\mathbf{x}}. \quad (5.6)$$

In Appendix A, we have proved that the proposed time discretization (5.3)-(5.5) preserves the total mass in time.

5.2 The LS-SVR for AC and CAC models

Here, we show how the presented LS-SVR can be applied to the semi-discrete problems (5.1)-(5.2) and (5.3)-(5.5). For doing so, the set of N scattered trial points such as $X = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ on $\mathbb{S}^2$ is considered. Hence, the approximate solution at each time step n can be written as:

$$\hat{u}(\mathbf{x}, t^n) = \mathbf{a}^T(\mathbf{x}) \mathbf{w}_n, \quad \mathbf{x} \in \mathbb{S}^2, \quad (5.7)$$

where $\mathbf{w}_n = [w_1^n, \dots, w_N^n]^T$, and $\mathbf{a}^T(\mathbf{x}) = [a_1(\mathbf{x}), \dots, a_N(\mathbf{x})]$ denotes the vector with trial functions.

5.2.1 AC model

We first note that the linear differential operator in both studied mathematical models is $\mathcal{L} = \text{Id} - \Delta t L$. Replacing the approximate solution (5.7) and executing the collocation technique based on the GMLS approximation, we get the following problem per time step $n = 0, \dots, K-1$;

Solve the following minimization problem.

$$\min_{\mathbf{w}_n^*, \mathbf{e}_n^* \in \mathbb{R}^N} \frac{1}{2} \mathbf{w}_n^{*T} \mathbf{w}_n^* + \frac{\gamma}{2} \mathbf{e}_n^{*T} \mathbf{e}_n^*, \quad (5.8)$$

$$s.t. \ A_X \mathbf{w}_n^* - \Delta t L_X \mathbf{w}_n^* = A_X U_X^n + \mathbf{e}_n^*, \quad (5.9)$$

where $\mathbf{e}_n$ is the residual term, $A_X = [a_j(\mathbf{x}_i)]_{1 \leq i, j \leq N}$ and $L_X = [a_j^L(\mathbf{x}_i)]_{1 \leq i, j \leq N}$ are sparse matrices thanks to the presented GMLS approximation, which are the same with those differentiation matrices computed for approximating the point-evaluation functional and Laplace-Beltrami operators in Section 2, respectively. and $\mathbf{w}_n^*$ is the vector of intermediate weights, which can be computed by applying KKT conditions as follows:

$$(A_L A_L^T + \frac{1}{\gamma} I_N) \boldsymbol{\alpha}_n^* = A_X U_X^n, \quad (5.10)$$

which is a dual form of the problem (5.8)-(5.9). Besides, U_X^n is the approximate solution at points X in $t = t_n$ with $A_L = A_X - \Delta t L_X$, and

$$\mathbf{w}_n^* = A_L^T \boldsymbol{\alpha}_n^*, \quad \mathbf{e}_n^* = -\frac{1}{\gamma} \boldsymbol{\alpha}_n^*. \quad (5.11)$$

We can now compute the approximate solution to the intermediate variable as

$$U_X^{n*} = A_X \mathbf{w}_n^*. \quad (5.12)$$

Finally, due to Eq. (5.2), the approximate solution at time level $t = t_{n+1}$ can be calculated with the following relation.

$$U_X^{n+1} = \frac{U_X^{n*}}{\sqrt{\exp\left(-\frac{2\Delta t}{\epsilon^2}\right) + (U_X^{n*})^2 \left(1 - \exp\left(-\frac{2\Delta t}{\epsilon^2}\right)\right)}}. \quad (5.13)$$

Now, we can give the following theorem that shows the proposed numerical scheme applied to the AC equation (1.1) has a unique solution.

Theorem 5.1. *For $n \geq 0$, the discrete scheme given in (5.10)-(5.13) has a unique solution U_X^{n+1} .*

Proof. For $n \geq 0$, the linear system (5.10) has a unique solution α_n^* because the coefficient matrix $B_L =: A_L A_L^T + \frac{1}{\gamma} I_N$ is positive definite (see the proof of Theorem 3.1). Accordingly, it is invertible, and hence the vectors (5.11) and the approximate solution of the intermediate variable u^{n*} , i.e., U_X^{n*} are uniquely obtained. Finally, since U_X^{n+1} is directly computed by relation (5.13), the proof is completed. $\square$

5.2.2 CAC model

Additionally, to apply the proposed LS-SVR to the semi-discrete problem (5.3)-(5.4), we employ the same procedure executed to the AC model. Accordingly, using the collocation technique based on the GMLS approximation with N scattered trial points on $\mathbb{S}^2$, we have the following.

For $n = 0, \dots, K-1$, the following minimization problem should be solved.

$$\min_{\mathbf{w}_n^*, \mathbf{e}_n^* \in \mathbb{R}^N} \frac{1}{2} \mathbf{w}_n^{*T} \mathbf{w}_n^* + \frac{\gamma}{2} \mathbf{e}_n^{*T} \mathbf{e}_n^*, \quad (5.14)$$

$$s.t. \ A_X \mathbf{w}_n^* - \Delta t L_X \mathbf{w}_n^* = A_X U_X^n + \mathbf{e}_n^*, \quad (5.15)$$

In addition, the vector of intermediate weights, i.e., $\mathbf{w}_n^*$ is computed by using the KKT conditions as follows:

$$(A_L A_L^T + \frac{1}{\gamma} I_N) \alpha_n^* = A_X U_X^n, \quad (5.16)$$

that is a dual form of the problem (5.14)-(5.15). Also, U_X^n shows the approximate solution at points X in $t = t_n$ and $A_L = A_X - \Delta t L_X$, and

$$\mathbf{w}_n^* = A_L^T \alpha_n^*, \quad \mathbf{e}_n^* = -\frac{1}{\gamma} \alpha_n^*. \quad (5.17)$$

The approximate solution to the intermediate variable can now be computed as

$$U_X^{n*} = A_X \mathbf{w}_n^*. \quad (5.18)$$

Besides, the approximate solution to the second intermediate variable can be calculated with the following relation due to Eq. (5.4).

$$U_X^{n+1**} = \frac{U_X^{n*}}{\sqrt{\exp(-\frac{2\Delta t}{\epsilon^2}) + (U_X^{n*})^2 (1 - \exp(-\frac{2\Delta t}{\epsilon^2}))}}. \quad (5.19)$$

Finally, according to (5.6), the approximate solution at time level $t = t_{n+1}$ is obtained by

$$U_X^{n+1} = U_X^{n+1**} + \Delta t \beta^{n+1**} \sqrt{F(U_X^{n+1**})}, \quad (5.20)$$

where β^{n+1**} is computed based on the integration of scattered data technique applied to (5.6), where 16382 spherical t -design points is used [5, 59]. For instance, to compute the integration of u over $\mathbb{S}^2$, we have utilized the following approximation.

$$\int_{\mathbb{S}^2} u(\mathbf{x}) d\mathbf{x} \approx \frac{4\pi}{N_g} \sum_{j=1}^{N_g} u(\mathbf{y}_j), \quad (5.21)$$

in which $\{\mathbf{y}_1, \dots, \mathbf{y}_{N_g}\}$ denotes N_g spherical t -design points (cf. [5, 59]). In Figure 1, two sets of spherical t -design points with cardinalities 969 and 2666 are plotted.



Figure 1: The plots of 966 (left) and 2666 (right) spherical t -design points on $\mathbb{S}^2$.

Theorem 5.2. *For $n \geq 0$, the discrete scheme given in (5.16)-(5.20) has a unique solution U_X^{n+1} .*

Proof. The proof can be done in the same manner presented in Theorem 5.1. $\square$

In Appendix B, for the reader's convenience, we present the computational algorithms associated with the numerical methods used to solve the AC and CAC equations.

6 Numerical results

To demonstrate the potency of the developed numerical method based on a type of SVM, i.e., the LS-SVR combined with a meshfree approximation on spheres, some numerical simulations are presented in this section.

To use the GMLS approximation, we consider the spherical harmonic functions with degree $m = 2$ or less and the weight function as follows (cf. [57, Chapter 9]):

$$\phi(r) = \begin{cases} (1-r)^{10}(429r^4 + 450r^3 + 210r^2 + 50r + 5), & 0 \leq r \leq 1, \\ 0, & r > 1, \end{cases} \quad (6.1)$$

in which $r = \text{dist}(\mathbf{x}, \mathbf{y})/\delta$ and $\mathbf{x}, \mathbf{y} \in \mathbb{S}^2$.

Additionally, the trial approximation (3.2) is constructed by choosing two sets of quasi-uniform nodes, i.e., the minimum energy (ME) and the phyllotaxis spiral (PTS) on $\mathbb{S}^2$, where their fill distance is of order $\mathcal{O}(N^{-1/2})$ (cf. [20, 41]). In Figure 2, 2601 ME and PTS nodes are plotted, where a spherical cap related to a test point and its trial points are denoted.

In the computations presented below, to avoid any instability in solving the local normal linear systems arising in Eqs. (2.2) and (2.6), the QR factorization of the matrix $\sqrt{W}P$ to compute $C = WP(P^TW(\mathbf{x})P)^{-1}$ via the backward substitution from $RC^T = Q^T\sqrt{W}$ is applied. Note that, the computational complexity of the approximation (2.2) (or (2.6)) for each test point $\mathbf{x} \in \mathbb{S}^2$ is of order $\mathcal{O}(N(3, m)^3)$ (cf. [38]).

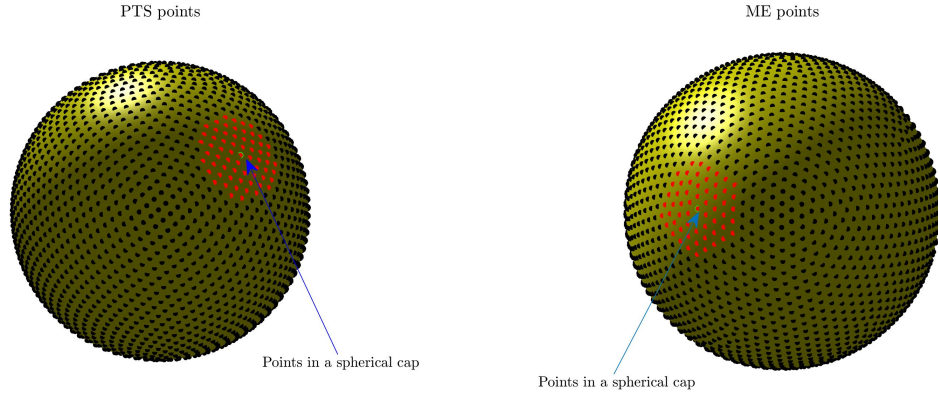


Figure 2: The plots of 2601 ME (left) and PTS (right) points with a spherical cap containing red points neighboring 2000-th point.

On the other hand, instead of solving the obtained linear systems (5.10) and (5.16), we first use the reverse Cuthill–McKee (RCM) reordering algorithm [24, 53]. Then, the preconditioned conjugate gradient (PCG) algorithm with a zero-fill incomplete lower-upper (ILU(0)) factorization of the coefficient matrix, obtained by executing the RCM algorithm on B_L , is employed, where, for stopping the algorithm, tolerance and maximum number of iterations are chosen 10^{-6} and 100, respectively (cf. [51]). The observed numerical results (not reported here for brevity) show that the ILU(0) factorization is a proper preconditioner. Note that the implementation of the RCM approach leads to faster convergence when an iterative solver is used, as it permutes a sparse matrix with a symmetric sparsity pattern into a band matrix form with a small bandwidth (cf. [53]). In Figure 3, the sparsity pattern of B_L of size 6561×6561 (assumed $\mathcal{L} = \text{Id}$) before and after re-ordering with the RCM algorithm is drawn using both PTS and ME points.

6.1 Convergence test

Here, we numerically investigate the error bounds (4.21). To do so, the following smooth test function is considered [21, 40]:

$$u(x, y, z) = \cos(10z + 3)e^{-x^2} - \sin(4x + 5y), \quad (x, y, z) \in \mathbb{S}^2, \quad (6.2)$$

in which $\mathbb{S}^2$ is the unit sphere, i.e.,

$$\mathbb{S}^2 = \{(x, y, z) \in \mathbb{R}^3 : x^2 + y^2 + z^2 = 1\}. \quad (6.3)$$

In all numerical results presented here, PTS points on $\mathbb{S}^2$ are used. Besides, the size of each spherical cap is considered to be $\delta = 4mh$, where $m = 2$ and $h = 1/\sqrt{N}$. Besides, $\gamma = h^{-2}$ to deal with the errors, which obey $\mathcal{O}(h)$. The accuracy of LS-SVR based on the GMLS approximation is measured by computing the discrete maximum errors, i.e., ℓ_∞ with the numerical order, as displayed in Figure 4. However, the observed numerical convergence demonstrates the results being approximately twice order more accurate than the theoretical rate. From our observations, we also have seen that the numerical orders for different values of γ , i.e., h^{-3} , h^{-4} and h^{-5} are 3, 4 and 5 respectively.

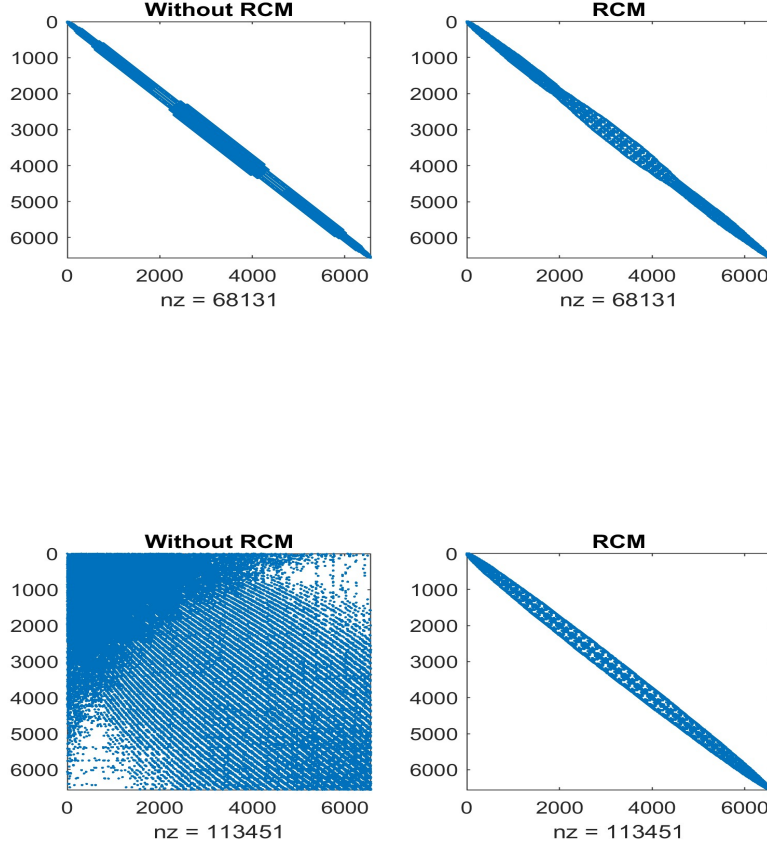


Figure 3: The sparsity pattern of B_L before and after re-ordering with the RCM algorithm, where 6561 PTS points (left) and ME points (right) are used.

6.2 Test 1 (Accuracy test): Motion by mean curvature on the unit sphere

As the first example, we denote the accuracy of numerical scheme in both time and space domains. So, the AC equation (1.1) with the following initial condition is considered [10].

$$u(x, y, z, 0) = \tanh\left(\frac{\sqrt{x^2 + y^2} - z}{\sqrt{2}\epsilon}\right), \quad (x, y, z) \in \mathbb{S}^2, \quad (6.4)$$

in which $\epsilon = 0.1$.

This problem is known as the motion by mean curvature on the unit sphere (cf. [10, 41]). Indeed, the interface of a circle on the unit sphere is evolving in time due to the motion of surface curvature. Accordingly, let the radii of the sphere and the circle be R and r , respectively. By considering some assumptions and simple calculations proposed in [10], the governing equation related to the motion of a circular interface on

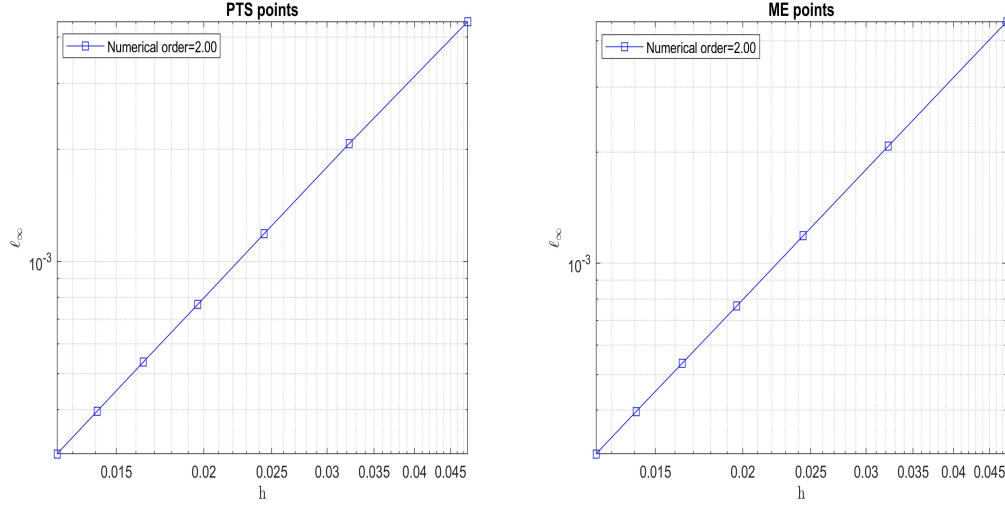


Figure 4: ℓ_∞ errors with the numerical order, PTS points (left) and ME points (right).

the sphere is derived as:

$$\frac{dr}{dt} = \frac{r^2 - R^2}{rR^2}, \quad (6.5)$$

where its exact solution is [10]

$$r(t) = \sqrt{R^2 - (R^2 - r_0^2) \exp\left(\frac{2t}{R^2}\right)}, \quad (6.6)$$

where $R = 1$ and $r_0 = \frac{1}{\sqrt{2}}$ [10].

To measure the accuracy of the developed numerical approach in both time and spatial discretizations, the radius of the spherical cap for each test point is chosen to be $\delta = 6mh$, where $m = 2$ and $h = 1/\sqrt{N}$. Also, the coefficient constant γ is empirically chosen to be 10^{12} .

In Tables 1 and 2, the discrete maximum errors, i.e., ℓ_∞ corresponding to the time and space discretizations for approximating the radii (6.6) are reported at $T = 0.2$. Note that, in obtaining the errors related to the time discretization, we have used 10201 PTS points. And, the errors computed based on the spatial discretization are reported by choosing $\Delta t = 10^{-4}$. Note also that the average of last three errors is approximately 1.35 in time, and its value is about 2.93 in space.

In addition, Figure 5 demonstrates the exact and approximate values of the radii r in time, where 6561 PTS points and $\Delta t = 10^{-4}$ are used.

Besides, to show the temporal evolution of u , we have chosen N and Δt to be 6561 and 10^{-2} , respectively, where both PTS and ME points are utilized. The desired results are drawn in Figures 6 and 7, which follow from those reported in [10, 41].

6.3 Test 2

In the second test, we present the numerical results related to the CAC equation (1.2) on $\mathbb{S}^2$ with $\epsilon = 0.0192$ and the initial condition as follows [32]:

Δt	ℓ_∞	Orders
10^{-2}	6.99×10^{-2}	—
5×10^{-3}	4.27×10^{-2}	0.71
2.5×10^{-3}	2.23×10^{-2}	0.94
1.25×10^{-3}	9.91×10^{-3}	1.17
6.25×10^{-4}	2.86×10^{-3}	1.79
3.125×10^{-4}	1.33×10^{-3}	1.10

Table 1: Maximum errors with the convergence rate related to the time discretization.

N	ℓ_∞	Orders
1681	3.78×10^{-2}	—
2601	1.96×10^{-2}	3.02
3721	1.34×10^{-2}	2.10
5041	7.69×10^{-3}	3.68

Table 2: Maximum errors with the convergence rate related to the spatial discretization.

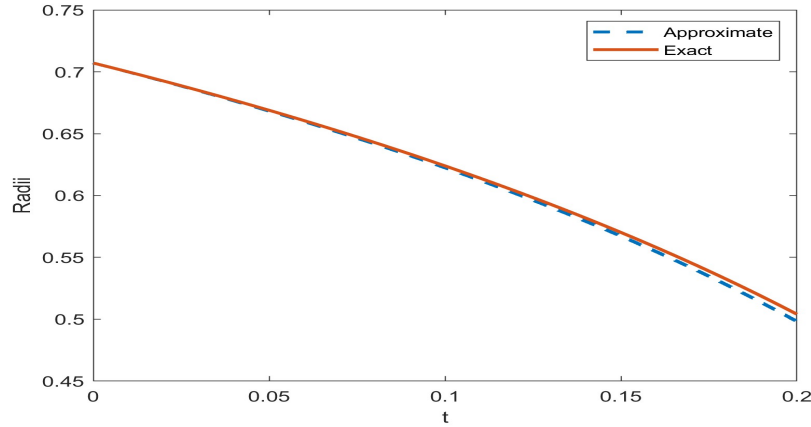


Figure 5: Exact and approximate values of $r(t)$ at different time levels up to 0.2.

$$u(x, y, z, 0) = \begin{cases} \tanh\left(\frac{0.8 - \sqrt{x^2 + y^2}}{\sqrt{2}\epsilon}\right), & z \geq 0, \\ \tanh\left(\frac{0.5 - \sqrt{x^2 + y^2}}{\sqrt{2}\epsilon}\right), & z < 0, \end{cases} \quad (6.7)$$

which shows two spherical caps with radii 0.8 and 0.5 located at the north and south poles, respectively. The interested reader is also referred to [32] for more information on this test problem.

Here, the PTS points on the unit sphere are used for presenting simulation results. Besides, the radius of each spherical cap is chosen to be $\delta = 6mh$, where $m = 2$ and $h = 1/\sqrt{N}$ with $N = 25600$. The time step Δt is also fixed, and its value is 4.5×10^{-5} . In Figure 8, the initial value with the approximate solution to u

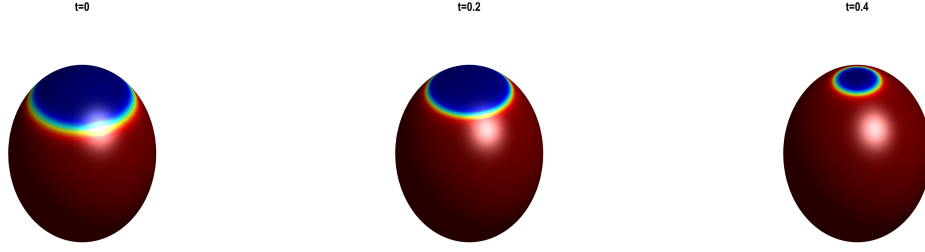


Figure 6: The initial condition with the approximate solution of u at $t = 0.2$ and $t = 0.4$ via 6561 PTS points.

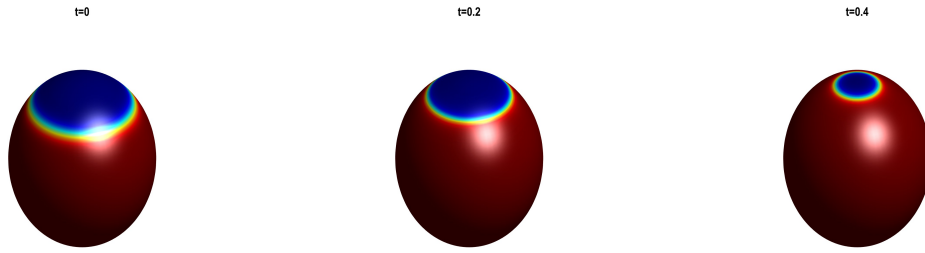


Figure 7: The initial condition with the approximate solution of u at $t = 0.2$ and $t = 0.4$ via 6561 ME points.

is plotted at $t = 0.112, 0.224$, and 0.272 . As can be observed here, the larger spherical cap grows, while its smaller counterpart shrinks until it disappears. However, the same results are obtained in [32], but using more points, i.e., 2803221, where the quasi-Neumann boundary condition on the narrow band domain using the closest point approach has also been considered. Besides, the approximate values of mass in time are displayed in a Figure 9. In Figure 10, we also report the approximate values of radii R_1 and R_2 in time.

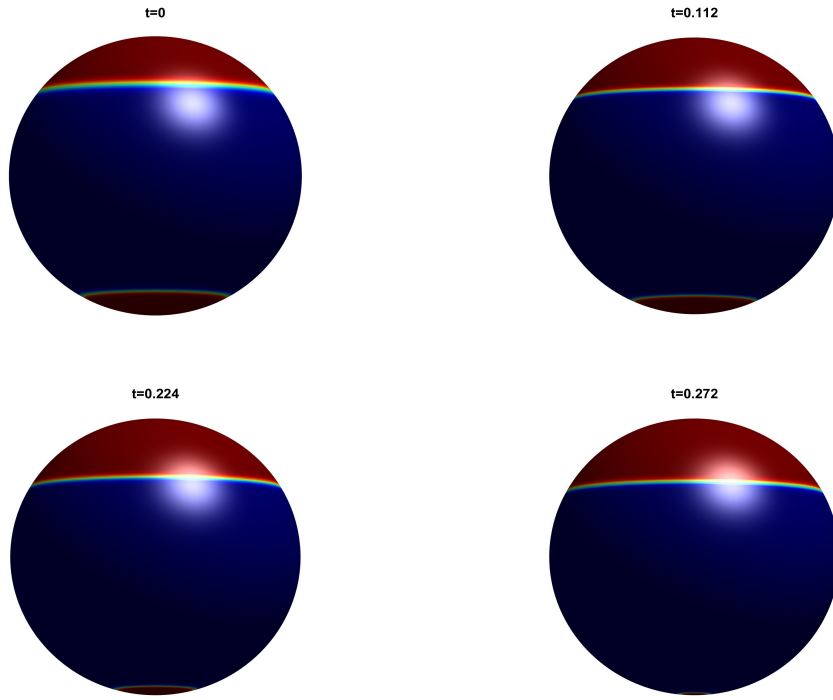


Figure 8: The initial condition with the approximate solution of u at $t = 0.112$, $t = 0.224$ and $t = 0.272$ via 25600 PTS points.

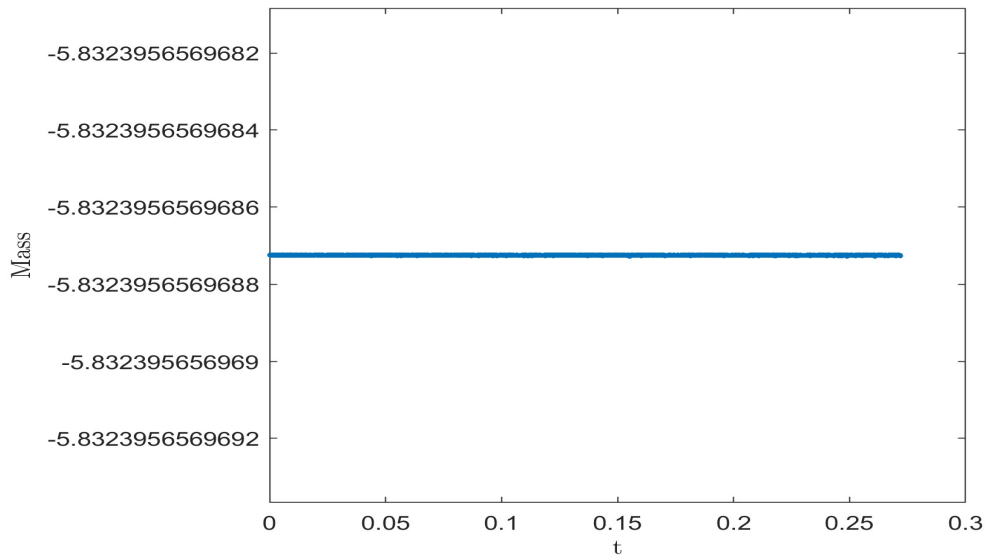


Figure 9: Approximate mass values over time.

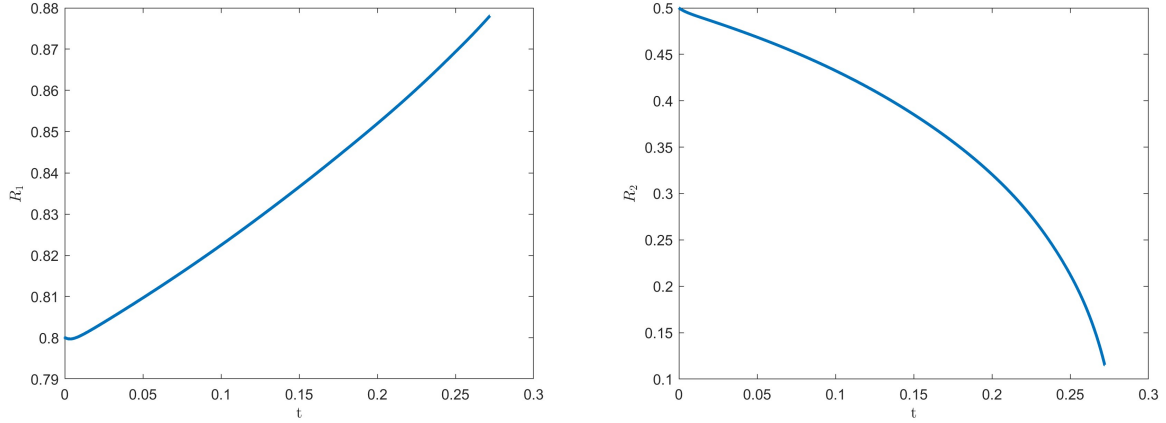


Figure 10: Temporal evolution of the radii R_1 (left) and R_2 (right).

7 Concluding remarks

We have defined a new RKHS whose kernel is constructed from the dual representation of the approximate solution obtained by applying the proposed LS-SVR based on the GMLS technique to the approximation problem on spheres. The connection of the presented learning algorithm to the smoothing problem and KRR has also been investigated in its associated RKHS. As an application, two phase-field equations on spheres, i.e., AC and CAC models have been solved via the operator-splitting approach in time and the proposed LS-SVR in space. For doing so, we first discretize both mathematical models using a first-order time-splitting approach, resulting in linear and nonlinear problems. Then, the LS-SVR with the GMLS procedure is applied to the linear problem per time step to deal with the approximate solution. Hence, using the KKT conditions, the unique approximate intermediate solution is computed by solving a linear system of algebraic equations in time, where the coefficient matrix is large, sparse, and symmetric positive definite, and yields the unique numerical solution per time step. Accordingly, the preconditioned conjugate gradient (PCG) algorithm, where an appropriate preconditioner, i.e., ILU(0) factorization of the coefficient matrix is proposed, is executed. After that, the approximate intermediate solution obtained at this stage is directly substituted into the semi-discrete scheme of the nonlinear problem to get the numerical solution. Finally, we have presented the results to illustrate the claimed error bounds at points X . Moreover, some presented simulation results confirm the suitability of this development in finding stable and accurate numerical solutions for the studied phase-field models on spheres.

References

- [1] M. Abbaszadeh, A. Khodadadian, M. Parvizi, M. Dehghan, D. Xiao, A reduced-order least squares-support vector regression and isogeometric collocation method to simulate Cahn-Hilliard-Navier-Stokes equation, *J. Comput. Phys.* 523 (2025) 113650.
- [2] M. Abbaszadeh, M. A. Zaky, A. S. Hendy, M. Dehghan, Supervised learning and meshless methods for two-dimensional fractional PDEs on irregular domains, *Math. Comput. Simulation.* 216 (2024) 77-103.
- [3] N. Aronszajn, Theory of reproducing kernels, *T. Am. Math. Soc.* 68 (3) (1950) 337- 404.

- [4] K. Atkinson, W. Han, Theoretical Numerical Analysis: A Functional Analysis Framework, Springer, USA, 2005.
- [5] K. Atkinson, W. Han, Spherical Harmonics and Approximations on the Unit Sphere: An Introduction. Vol. 2044, Springer Science & Business Media, New York, 2012.
- [6] M. Ahmadi-Darani, D. Mirzaei, A Petrov-Galerkin RBF method for diffusion equation on the unit sphere, Numer. Methods Partial Differ. Equ. 36(6) (2020) 1682-1698.
- [7] S. M. Allen, J. W. Cahn, A microscopic theory for antiphase boundary motion and its application to antiphase domain coarsening, Acta Metall. 27 (1979) 1085-1095.
- [8] V. Bayona, A. Petras, C. Piret, S. J. Ruuth, A meshfree RBF-FD constant along normal method for solving PDEs on surfaces, SIAM J. Sci. Comput. 46(6) (2024) A3897-A3921.
- [9] H. Brezis, Functional Analysis, Sobolev Spaces and Partial Differential Equations, Springer, London, 2011.
- [10] Y. Choi, D. Jeong, S. Lee, M. Yoo, J. Kim, Motion by mean curvature of curves on surfaces using the Allen-Cahn equation, Int. J. Eng. Sci. 97 (2015) 126-132.
- [11] F. Dai, Y. Xu, Polynomial approximation in Sobolev spaces on the unit sphere and the unit ball, J. Approx. Thy. 163 (2011) 1400-1418.
- [12] F. Dai, Y. Xu, Approximation Theory and Harmonic Analysis on Spheres and Balls, Springer, New York, 2013.
- [13] G. E. Fasshauer, Meshfree Approximation Methods with MATLAB, Vol. 6, World Scientific, USA, 2007.
- [14] G. Fasshauer, M. McCourt, Kernel-based Approximation Methods using MATLAB, World Scientific Publishing Co. Pte. Ltd. Singapore, 2016.
- [15] Z. Feng, J. Yin, J. Zhou, Inpainting algorithm for jacquard image based on phase-field model, IEEE Intell. Syst. Knowl. Eng. ISKE (2008) 1203-1207.
- [16] N. Flyer, E. Lehto, S. Blaise, G. B. Wright, A. St-Cyr, A guide to RBF-generated finite differences for nonlinear transport: Shallow water simulations on a sphere, J. Comput. Phys. 231 (2012) 4078-4095.
- [17] N. Flyer and G. B. Wright, Transport schemes on a sphere using radial basis functions, J. Comput. Phys. 226 (2007), 1059-1084.
- [18] N. Flyer, G. B. Wright, A radial basis function method for the shallow water equations on a sphere, Proc. R. Soc. A, 465 (2009), 1949-1976
- [19] E. J. Fuselier, G. B. Wright, Scattered data interpolation on embedded submanifolds with restricted positive definite kernels: Sobolev error estimates, SIAM J. Numer. Anal. 50 (3) (2012) 1753-1776.
- [20] E. J. Fuselier, G. B. Wright, A high-order kernel method for diffusion and reaction-diffusion equations on surfaces, J. Sci. Comput. 56 (3) (2013) 535-565.

- [21] E. J. Fuselier, G. B. Wright, Order-preserving derivative approximation with periodic radial basis functions, *Adv. Comput. Math.* 41 (2015) 23-53.
- [22] P. Gera, D. Salac, Cahn-Hilliard on surfaces: A numerical study, *Appl. Math. Lett.* 73 (2017) 56-61.
- [23] P. Gera, D. Salac, Stochastic phase segregation on surfaces. *R. Soc. Open Sci.* 4(8) (2017) 170472.
- [24] A. George and J. W. Liu, *Computer Solution of Large Sparse Positive Definite Systems*, Prentice Hall Professional Technical Reference, Englewood Cliffs, NJm, 1981.
- [25] Q. T. Le Gia, Galerkin approximation for elliptic PDEs on spheres, *J. Approx. Theory*, 130 (2004) 123-147.
- [26] Q. T. Le Gia, Approximation of parabolic PDEs on spheres using spherical basis functions, *Adv. Comput. Math.* 22 (2005) 377-397.
- [27] Q. T. Le Gia, I. H. Sloan, H. Wendland, Multiscale RBF collocation for solving PDEs on spheres, *Numer. Math.* 121 (2012) 99-125.
- [28] S. Hubbert, Radial basis function interpolation on the sphere. PhD thesis, Imperial College London, London, 2002.
- [29] Y. Hwang, S. Kwak, H. Kim, J. Kim, An explicit numerical method for the conservative Allen-Cahn equation on a cubic surface, *AIMS Mathematics*, 9(12) (2024) 34447-34465.
- [30] A. M. Jones, P. A. Bosler, P. A. Kuberry, G. B. Wright, Generalized moving least squares vs. radial basis function finite difference methods for approximating surface derivatives, *Comput. Math. Appl.* 147 (2023) 1-13.
- [31] J. Kim, S. Lee, Y. Choi, A conservative Allen-Cahn equation with a space-time dependent Lagrange multiplier, *Int. J. Eng. Sci.* 84 (2014) 11-17.
- [32] J. Kim, D. Jeong, S.-D. Yang, Y. Choi, A finite difference method for a conservative Allen-Cahn equation on non-flat surfaces, *J. Comput. Phys.* 334 (2017) 170-181.
- [33] H. G. Lee, J. Kim, A simple and efficient finite difference method for the phase-field crystal equation on curved surfaces, *Comput. Methods Appl. Mech. Engrg.* 307 (2016) 32-43.
- [34] E. Lehto, V. Shankar, and G. B. Wright, A radial basis function (RBF) compact finite difference (FD) scheme for reaction-diffusion equations on surfaces, *SIAM J. Sci. Comput.* 39(5) (2017) A2129-A2151.
- [35] Y. Li, D. Jeong, J. Choi, S. Lee, J. Kim, Fast local image inpainting based on the Allen-Cahn model, *Digit. Signal Process.* 37 (2015) 65-74.
- [36] S. Mehrkanon, T. Falck, J. A. Suykens, Approximate solutions to ordinary differential equations using least squares support vector machines. *IEEE Trans. Neural Netw. Learn. Syst.* 23 (2012) 1356-1367.
- [37] S. Mehrkanon, J. A. K. Suykens, Learning solutions to partial differential equations using LS-SVM, *Neurocomputing*, 159 (2015) 105-116.
- [38] D. Mirzaei, Direct approximation on spheres using generalized moving least squares, *BIT Numerical Mathematics*, 57 (2017) 1041-1063.

- [39] D. Mirzaei, A Petrov-Galerkin kernel approximation on the sphere, *SIAM J. Numer. Anal.* 56(1) (2018) 274-295.
- [40] R. Mir, D. Mirzaei, The D-RBF-PU method for solving surface PDEs, *J. Comput. Phys.* 479 (2023) 112001.
- [41] V. Mohammadi, D. Mirzaei, M. Dehghan, Numerical simulation and error estimation of the time-dependent Allen-Cahn equation on surfaces with radial basis functions, *J. Sci. Comput.* 79 (2019) 493-516.
- [42] V. Mohammadi, M. Dehghan, A. Khodadadian, Nima Noii, T. Wick, An asymptotic analysis and numerical simulation of a prostate tumor growth model via the generalized moving least squares approximation combined with semi-implicit time integration, *Appl. Math. Model.* 104 (2022) 826-849.
- [43] V. Mohammadi, M. Dehghan, A POD-RBF-FD scheme for simulating chemotaxis models on surfaces, *Eng. Anal. Bound. Elem.* 143 (2022) 316-330.
- [44] T. M. Morton, M. Neamtu, Error bounds for solving pseudodifferential equations on spheres, *J. Approx. Theory*, 114 (2002). 242–268.
- [45] F. J. Narcowich, J. D. Ward, Scattered-data interpolation on spheres: error estimates and locally supported basis functions. *SIAM J. Math. Anal.* 33 (2002) 1393–1410, .
- [46] F. J. Narcowich, S. T. Rowe, J. D. Ward, A novel Galerkin method for solving PDEs on the sphere using highly localized kernel bases, *Math. Comput.* 86 (2017) 197-231.
- [47] K. Parand, M. Razzaghi, R. Sahleh, M. Jani, Least squares support vector regression for solving Volterra integral equations, *Eng. Comput.* 38 (2020) 789-796.
- [48] K. Parand, A. A. Aghaei, M. Jani, A. Ghodsi, A new approach to the numerical solution of Fredholm integral equations using least squares-support vector regression, *Math. Comput. Simulation.* 180 (2021) 114-128.
- [49] A. Pakniyat, K. Parand, M. Jani, Least squares support vector regression for differential equations on unbounded domains, *Chaos Solitons Fractals*, 151 (2021) 111232.
- [50] M. Razzaghi, S. Shekarpaz, A. Rajabi, Solving ordinary differential equations by LS-SVM, *Learning with Fractional Orthogonal Kernel Classifiers in Support Vector Machines: Theory, Algorithms and Applications*. Singapore: Springer Nature Singapore, (2023) 147-170.
- [51] Y. Saad, *Iterative Methods for Sparse Linear Systems*, Soc. Ind. Appl. Math. (2003).
- [52] R. Schaback, Error analysis of nodal meshless methods, in: *Meshfree Methods for Partial Differential Equations VIII*, Bonn, 2015, in: *Lecture Notes in Computational Science and Engineering*, vol.115, Springer, 2017, pp.117-143.
- [53] V. Shankar, G. B. Wright, A. L. Fogelson, and R. M. Kirby, A radial basis function (RBF)-finite difference (FD) method for diffusion and reaction-diffusion equations on surfaces, *J. Sci. Comput.* 63 (2015) 745-768.
- [54] I. Steinwart, A. Christmann, *Support Vector Machines*. Springer Science & Business Media, 2008.

- [55] M. Torkian, M. Abbaszadeh, S. M. Alavi, M. Haghverdi, A supervised learning algorithm to simulate tumor growth: Cahn-Hilliard model on surfaces, *Eng. Anal. Bound. Elem.* 173 (2025) 106132.
- [56] H. Wendland, Moving least squares approximation on the sphere, *Mathematical Methods for Curves and Surfaces*, Vanderbilt Univ. Press, Nashville, TN (2001): 517-526.
- [57] H. Wendland, Scattered Data Approximation, in: *Cambridge Monograph on Applied and Computational Mathematics*, Cambridge University Press, New York, 2005.
- [58] H. Wendland, C. Rieger, Approximate interpolation with applications to selecting smoothing parameters, *Numer. Math.* 101 (2005) 729-748.
- [59] R. S. Womersley, I. H. Sloan, Interpolation and cubature on the sphere, Website. <http://web.maths.unsw.edu.au/~rsw/Sphere/>.
- [60] X. Wu, G.J. Zwieten, K.G. Zee, Stabilized second-order convex splitting schemes for Cahn-Hilliard models with application to diffuse-interface tumor growth models, *Int. J. Numer. Methods Biomed. Eng.* 30 (2014) 180-203.
- [61] X. Xiao, X. Feng, J. Yuan, The stabilized semi-implicit finite element method for the surface Allen-Cahn equation, *Discrete and Continuous Dynamical Systems, Series B*, 22 (2017) 2857-2877.
- [62] Y. Xu, E. W. Cheney, Strictly positive definite functions on spheres, *Proc. Amer. Math. Soc.* 116 (1992) 977-981.
- [63] H. Zamani-Gharaghoshi, M. Dehghan, M. Abbaszadeh, Numerical simulation of Cahn-Hilliard systems on surfaces using the moving Taylor polynomial (MTP) approximation and the closest point method, *Eng. Anal. Bound. Elem.* 180 (2025) 106478.

Appendix A

Lemma 7.1. *If $u(\mathbf{x}, t)$ is the solution of Eq. (1.2). Then, it satisfies the following mass conservation property.*

$$\frac{d}{dt} \int_{\mathbb{S}^2} u(\mathbf{x}, t) d\mathbf{x} = 0, \quad t > 0. \quad (7.1)$$

Proof. By multiplying both sides of Eq. (1.2) to the test function 1 in $L_2(\mathbb{S}^2)$, we have

$$\left(\frac{\partial u(\mathbf{x}, t)}{\partial t}, 1 \right)_{L_2(\mathbb{S}^2)} = (\Delta_{\mathbb{S}^2} u(\mathbf{x}, t), 1)_{L_2(\mathbb{S}^2)} - \left(\frac{F'(u(\mathbf{x}, t))}{\epsilon^2}, 1 \right)_{L_2(\mathbb{S}^2)} + \beta(t) (\sqrt{F(u(\mathbf{x}, t))}, 1)_{L_2(\mathbb{S}^2)}. \quad (7.2)$$

Since $\mathbb{S}^2$ is a smooth, 2-dimensional compact manifold, applying the Green-Beltrami (cf. [5]) identity yields

$$(\Delta_{\mathbb{S}^2} u(\mathbf{x}, t), 1)_{L_2(\mathbb{S}^2)} = (\nabla_{\mathbb{S}^2} u(\mathbf{x}, t), \nabla_{\mathbb{S}^2} 1)_{L_2(\mathbb{S}^2)} = 0. \quad (7.3)$$

On the other hand, according to the definition $\beta(t)$, the second and third terms on the right hand side of

(7.2) can be written as follows:

$$\begin{aligned}
& - \left(\frac{F'(u(\mathbf{x}, t))}{\epsilon^2}, 1 \right)_{L_2(\mathbb{S}^2)} + \beta(t) (\sqrt{F(u(\mathbf{x}, t))}, 1)_{L_2(\mathbb{S}^2)} = \\
& - \int_{\mathbb{S}^2} \frac{F'(u(\mathbf{x}, t))}{\epsilon^2} d\mathbf{x} + \left(\frac{\int_{\mathbb{S}^2} F'(u(\mathbf{x}, t)) d\mathbf{x}}{\epsilon^2 \int_{\mathbb{S}^2} \sqrt{F(u(\mathbf{x}, t))} d\mathbf{x}} \right) \int_{\mathbb{S}^2} \sqrt{F(u(\mathbf{x}, t))} d\mathbf{x} = 0,
\end{aligned} \tag{7.4}$$

which completes the proof. $\square$

Lemma 7.2. *The time discretization (5.3)-(5.5) is mass conservative, i.e.,*

$$(u^{n+1}, 1)_{L_2(\mathbb{S}^2)} = (u^0, 1)_{L_2(\mathbb{S}^2)}, \quad \forall n \geq 0. \tag{7.5}$$

Proof. According to relation (5.5), for $n \geq 0$, we can write

$$\begin{aligned}
(u^{n+1}, 1)_{L_2(\mathbb{S}^2)} &= (u^{n+1**} + \Delta t \beta^{n+1**} \sqrt{F(u^{n+1**})}, 1)_{L_2(\mathbb{S}^2)} \\
&= (u^{n+1**}, 1)_{L_2(\mathbb{S}^2)} + \Delta t \beta^{n+1**} (\sqrt{F(u^{n+1**})}, 1)_{L_2(\mathbb{S}^2)} \\
&= (u^{n+1**}, 1)_{L_2(\mathbb{S}^2)} + \Delta t \left(\frac{(u^0 - u^{n+1**}, 1)_{L_2(\mathbb{S}^2)}}{\Delta t (\sqrt{F(u^{n+1**})}, 1)_{L_2(\mathbb{S}^2)}} \right) (\sqrt{F(u^{n+1**})}, 1)_{L_2(\mathbb{S}^2)} \\
&= (u^0, 1)_{L_2(\mathbb{S}^2)},
\end{aligned} \tag{7.6}$$

where in the third equality, we have used relation (5.6). $\square$

Appendix B

Here are the developed numerical schemes applied to Eqs. (1.1) and (1.2) for the reader's convenience, where Algorithm 1 refers to the GMLS approximation. Algorithms 2 and 3 propose the detailed procedures.

Algorithm 1 The computational algorithm of the GMLM approach on $\mathbb{S}^2$ for approximating the Laplace-Beltrami operator.

Input: $X = [\mathbf{x}_1, \dots, \mathbf{x}_N]^T \in \mathbb{R}^{N \times 3}$ as trial points on $\mathbb{S}^2$, $Y = [\mathbf{y}_1, \dots, \mathbf{y}_M]^T \in \mathbb{R}^{M \times 3}$ as trial points on $\mathbb{S}^2$, δ as a radius of the spherical cap;
Output: $\mathbf{a}(\mathbf{x})$ and $\mathbf{L}\mathbf{a}(\mathbf{x})$ at test points Y to construct sparse matrices A_X and L_X ;
Set the matrices A_X and L_X to be zeros of size $M \times N$;
for $i = 1, \dots, M$ **do**
 Construct $I(\mathbf{y}_i)$ with respect to X ;
 Compute the matrix W by choosing the weight function (6.1);
 Compute the local matrix $P^T W(\mathbf{y}_i) P$ with respect to trial points in $I(\mathbf{y}_i)$;
 Compute the vectors $\mathbf{Y}^T(\mathbf{y}_i)$ and $\mathbf{L}\mathbf{Y}^T(\mathbf{y}_i)$ (see relation (2.3));
 Compute the vectors $\mathbf{a}(\mathbf{y}_i)$ and $\mathbf{L}\mathbf{a}(\mathbf{y}_i)$ due to Eqs. (2.6) and (2.2) and using the QR factorization of $\sqrt{W}P$ to find $C =: WP(P^T W(\mathbf{y}_i) P)^{-1}$;
 $A_X(i, I(\mathbf{y}_i)) \leftarrow \mathbf{a}(\mathbf{y}_i)$;
 $L_X(i, I(\mathbf{y}_i)) \leftarrow \mathbf{L}\mathbf{a}(\mathbf{y}_i)$;
end for

Algorithm 2 The computational algorithm of the numerical scheme applied to the AC equation (1.1) on $\mathbb{S}^2$.

- 1: **Input:** $X = [\mathbf{x}_1, \dots, \mathbf{x}_N]^T \in \mathbb{R}^{N \times 3}$ as data points on $\mathbb{S}^2$, $\delta > 0$ as the spherical cap, ϵ, γ, T as final time, and the time step Δt ;
- 2: **Output:** Approximate solution U_X at the final time T ;
- 3: Construct the sparse matrices A_X and A_L of size $N \times N$ due to Algorithm 1;
- 4: Form the matrix $B = A_L A_L^T + \frac{1}{\gamma} I_N$;
- 5: Apply the RCM algorithm to the coefficient matrix B , called B_R ;
- 6: Use ILU(0) factorization to the matrix B_R to construct the proper preconditioner;
- 7: $t \leftarrow 0$;
- 8: $n \leftarrow 0$;
- 9: Set the vector of initial condition at points X , i.e., $U_X^0 \in \mathbb{R}^{N \times 1}$;
- 10: **while** $t \leq T$ **do**
- 11: Find $\boldsymbol{\alpha}_n^*$ by applying the RCM algorithm to the linear system (5.10), and solve it via the PCG algorithm;
- 12: Compute $\mathbf{w}_n^*$ from the first relation in (5.11);
- 13: Compute the intermediate solution U_X^{n*} via (5.12);
- 14: Find the approximate solution U_X^{n+1} using (5.13);
- 15: $n \leftarrow n + 1$;
- 16: $t \leftarrow n\Delta t$;
- 17: **end while**
- 18: $U_X \leftarrow U_X^K$;

Algorithm 3 The computational algorithm of the numerical scheme applied to the CAC equation (1.2) on $\mathbb{S}^2$.

- 1: **Input:** $X = [\mathbf{x}_1, \dots, \mathbf{x}_N]^T \in \mathbb{R}^{N \times 3}$ as data points on $\mathbb{S}^2$, $\delta > 0$ as the spherical cap, ϵ, γ, T as final time, and the time step Δt ;
 - 2: **Output:** Approximate solution U_X at the final time T ;
 - 3: Construct the sparse matrices A_X and A_L of size $N \times N$ due to Algorithm 1;
 - 4: Form the matrix $B = A_L A_L^T + \frac{1}{\gamma} I_N$;
 - 5: Apply the RCM algorithm to the coefficient matrix B , called B_R ;
 - 6: Use ILU(0) factorization to the matrix B_R to construct the proper preconditioner;
 - 7: $t \leftarrow 0$;
 - 8: $n \leftarrow 0$;
 - 9: Set the vector of initial condition at points X , i.e., $U_X^0 \in \mathbb{R}^{N \times 1}$;
 - 10: **while** $t \leq T$ **do**
 - 11: Find α_n^* by applying the RCM algorithm to the linear system (5.16), and solve it via the PCG algorithm;
 - 12: Compute $\mathbf{w}_n^*$ from the first relation in (5.17);
 - 13: Compute the intermediate solution U_X^{n*} via (5.18);
 - 14: Find the approximate solution U_X^{n+1**} using (5.19);
 - 15: Compute the coefficient β^{n+1**} via the numerical integration (5.21);
 - 16: Find the approximate solution U_X^{n+1} via (5.20);
 - 17: $n \leftarrow n + 1$;
 - 18: $t \leftarrow n \Delta t$;
 - 19: **end while**
 - 20: $U_X \leftarrow U_X^K$;
-