Meshless collocation methods with graph Laplacian

Ka Chun Cheung & Leevan Ling

July 20, 2017

Abstract

This work aims to provide a numerical treatment to deal with PDEs on surfaces with singularity along smooth curves. Previously, we proposed a low-order localized meshless method for discretizing the surface Laplacian operator on surfaces with folded. It is observed that the previously proposed strategy of approximating surface Laplacian at fold does not work with a recently developed high-order embedding meshless method for smooth surface. In this paper, we propose using the graph Laplacian to handel the surface singularity and study how it can be used to solve PDEs on folded surfaces.

1 Introduction

Partial differential equations (PDEs) on surfaces (or manifolds) arise in a wide variety of applications including mathematical biology , mathematical physics, imaging processing and computer graphics, see [1, 2, 3, 4]. There are many methods in literature developed for solving surface PDEs. These methods can be classified into two groups: intrinsic methods and embedding methods. For intrinsic, it is typically required to parameterize the surface. Discretization of surface differential operators are less trivial and could lead to distortions and singularities on the surface [5, 6, 7, 8]. This group of methods are generally of low order of accuracy and the dimension of the discretization of the surface differential operator is identical to the original problem. For the embedding types, it is necessary to embed the surface differential operator into some subsets of the Euclidean space $\mathbb{R}^d$. The solution of the original problem is obtained by taking an appropriate restriction to the solution of the embedding problem. Hence, the dimension of the discretization will be greater than the original problem and naturally the computational cost could increase if no extra cares are given. Ruuth. et al [9] proposed a closest point method, which is an embedding method by means of a closest point mapping. The closest point method is simple and the order of convergence depends on the numerical method for the embedding problem. Hence, it is not difficult to obtain a highly accurate method for solving partial differential equation on smooth surfaces.

Kernel approximation are normally of high accuracy. With some modifications of the closest point method, a kernel method using radial basis function was developed for solving partial differential equation on surface [10]. This method is highly accurate when applied to smooth surfaces. For those surface that are not smooth, a localized meshless method to solve diffusion equation on folded surface was proposed in [11]. In this paper, we combine the ideas in these papers and propose a new embedding condition for the surface Laplacian and surface gradient. This embedding condition can be viewed as a limiting version of our previous approach. Hence, the new method reduces the error induced from the operator embedding. In [10], the author also provide a high order embedding condition for the surface Laplacian and the surface gradient. However, the embedding condition requires the derivative of the normal vector of the surface while our new embedding condition do not involve any derivative of the normal vector. In particular, we solve diffusion equations by the method of line by obtaining a differentiation matrix D as an approximation of the surface Laplacian so that the semi-discrete form of the diffusion equation is given by $\mathbf{u}_t = D\mathbf{u}$. Finally, the solution will be obtained by applying some suitable numerical method for this system of ODE.

2 Model formulation

In this section, we focus on solving diffusion equation on folded manifolds. Let $\mathcal{S}$ be a folded manifold and assume $\mathcal{S} = \cup\mathcal{S}_i$, where each $\mathcal{S}_i$ is a smooth sub-manifold for $i \in \mathbb{N}$. Let us assume further that the curves $\Gamma_{ij} = \mathcal{S}_i \cap \mathcal{S}_j$, on which the manifold $\mathcal{S}$ folds, are piecewise smooth. Our objective is to solve

$$\frac{\partial u_{\mathcal{S}}}{\partial t} = \Delta_{\mathcal{S}} u_{\mathcal{S}} \quad \text{on} \quad \mathcal{S}. \tag{2.1}$$

Note that $u_{\mathcal{S}}$ is defined on $\mathcal{S}$ and $\Delta_{\mathcal{S}}$ is the surface Laplace operator that will be explained later.

The surface $\mathcal{S}$ in our consideration is a composition of many pieces of smooth manifolds $\mathcal{S}_i$. In the interior of each $\mathcal{S}_i$, we can approximate $\Delta_{\mathcal{S}} u_{\mathcal{S}}$ of (2.1) by some standard techniques for smooth manifold. For those points on the fold Γ_{ij} of $\mathcal{S}$, most approaches for smooth surface fail. That is, it does not make sense to approximate a non-smooth manifold by any approximation for smooth manifolds. Therefore, we apply a special strategy for the approximation Laplacian. For $x \in \mathcal{S}_i$, we denote the approximation of $\delta_{x,\mathcal{S}_i}(\Delta_{\mathcal{S}} u_{\mathcal{S}})$ by $\delta_{x,\mathcal{S}_i}(D(u_{\mathcal{S}}))$. Supose $x \in \Gamma_{ij} = \mathcal{S}_i \cap \mathcal{S}_j$. Let $x_{\mathcal{S}_i} \in \mathcal{S}_i$ and $x_{\mathcal{S}_j} \in \mathcal{S}_j$ having the property that $\text{dist}(x, x_{\mathcal{S}_i}) = h = \text{dist}(x, x_{\mathcal{S}_j})$. Let $x_\Gamma^k \in \Gamma$ such that $\text{dist}(x, x_\Gamma^k) = h$, $k = 1, 2$. We approximate the surface Laplacian at the fold by $\delta_{x,\Gamma}(D(u_{\mathcal{S}})) = \frac{1}{h^2}(u(x_\Gamma^1) + u(x_\Gamma^2) + u(x_{\mathcal{S}_i}) + u(x_{\mathcal{S}_j}) - 4u(x))$. This discrete Laplacian is known as the Graph Laplacian. It is easy to see that it is identical to the finite difference approximation of Laplacian when it is applied to a flat plane.

In the following, we briefly introduce the embedding method for solving diffusion equation on folded manifolds and discretization of surface Laplacian by kernel approxmiation.

2.1 Surface Laplacian

We solve diffusion equation on manifolds by an embedding method. With certain embedding conditions, we can solve our problem on some subsets in the Euclidean space $\mathbb{R}^d$. Let $\mathbf{n}$ be the normal vector of the manifold $\mathcal{S}$ and ∇ be the cartesian gradient operator, the surface gradient operator on $\mathcal{S}$ and the surface Laplacian, a.k.a. the Laplace-Beltrami operator, are respectively defined as

$$\nabla_{\mathcal{S}} = \nabla - \mathbf{n}\mathbf{n}^T \nabla \tag{2.2}$$

and

$$\Delta_{\mathcal{S}} = (\nabla - \mathbf{n}\mathbf{n}^T \nabla) \cdot (\nabla - \mathbf{n}\mathbf{n}^T \nabla). \tag{2.3}$$

In order to embed the PDE to the Euclidean space, it is interesting to find some embedding conditions so that $\nabla_{\mathcal{S}} u_E = \nabla u_E$ and $\Delta_{\mathcal{S}} u_E = \Delta u_E$ agree on $\mathcal{S}$. Note that u_E is an extension of $u_{\mathcal{S}}$ defined on $\mathbb{R}^d$ such that $u_E(x) = u_{\mathcal{S}}(x)$ for all $x \in \mathcal{S}$. Based on the closest point method [9], we previously proposed an embedding condition to solve diffusion equation on manifolds. i.e.,

$$u_E(x) = u_E(x \pm \delta \mathbf{n}), \qquad x \in \mathcal{S}. \tag{2.4}$$

By some simple calculations, (2.4) can be easily transformed to

$$\begin{aligned} u_E(x + \delta\mathbf{n}) - u_E(x) + u_E(x - \delta\mathbf{n}) &= 0 \qquad (2.5)\\ u_E(x + \delta\mathbf{n}) - u_E(x - \delta\mathbf{n}) &= 0. \qquad (2.6)\end{aligned}$$

If we assign (2.5) and (2.6) with some suitable weighings, they become

$$\begin{aligned} \frac{1}{\delta^2}\big(u_E(x + \delta\mathbf{n}) - u_E(x) + u_E(x - \delta\mathbf{n})\big) &= 0 \qquad (2.7)\\ \frac{1}{2}\big(u_E(x + \delta\mathbf{n}) - u_E(x - \delta\mathbf{n})\big) &= 0. \qquad (2.8)\end{aligned}$$

We can see that (2.7) and (2.8) converge to $\frac{\partial^2 u_E}{\partial \mathbf{n}^2}$ and $\frac{\partial u_E}{\partial \mathbf{n}}$ as $\delta \to 0$, respectively. Hence, the embedding condition (2.4) is an approximation to the first and second normal derivative. In this paper, we consider a new embedding condition for the embedding:

$$\begin{aligned} \nabla u_E \cdot \mathbf{n} =: \mathcal{E}_1 u_E &= 0 \quad (2.9)\\ \mathbf{n}^T \mathrm{H}(u_E)\mathbf{n} =: \mathcal{E}_2 u_E &= 0, \quad (2.10)\end{aligned}$$

where $\mathrm{H}(u_E)$ is the Hessian matrix of u_E. This embedding condition is induced from the the embedding condition in [11]. Therefore, it remains to solve the embedding PDEs in some tubular neighborhoods $\Omega_\delta \subset \mathbb{R}^d$ by some suitable approximation:

$$\left\{ \begin{array}{lll} \frac{\partial u_E}{\partial t} = \Delta u_E & \text{in} & \Omega_\delta \\ 0 = \nabla u_E \cdot \mathbf{n} = \mathbf{n}^T H(u_E)\mathbf{n} & \text{on} & \mathcal{S}. \end{array} \right. \quad (2.11)$$

The solution of the original partial differential equation on manifold can be obtained by taking restriction to u_E on $\mathcal{S}$. In the next section, we will discuss how to discretize the Laplacian by kernel approximation.

2.2 Discretization of the surface Laplacian

Let $\Omega_\delta = \{y \in \mathbb{R}^d : \mathrm{dist}(y, \mathcal{S}) < \delta\}$ be the tubular neighborhood of manifold $\mathcal{S}$. Let $X \subset \mathcal{S}$ be a discrete set of N points on manifold $\mathcal{S}$ and $Z_\delta = \{y : y = x \pm \delta \mathbf{n}\} \subset \Omega_\delta$ be a discrete set of $3N$ points in Ω_δ. Let $K : \Omega_\delta \times \Omega_\delta \to \mathbb{R}$ be a symmetric positive definite kernel and $V_{Z_\delta,\varphi} = \mathrm{span}\big(\{K(\cdot, z_j) : z_j \in Z_\delta\}\big)$ be the trial space which is spanned by the basis functions $K(\cdot, z_j)$ for $z_j \in Z_\delta$. Let $u_{Z_\delta,K,\boldsymbol{\lambda}} = \sum_j \lambda_j K(x, z_j) \in V_{Z_\delta,K}$ be an approximation to u_E in $V_{Z_\delta,K}$. The objective of this section is to discretize the surface Laplacian on $\mathcal{S}$. By the strategy for the approximated Laplacian on the fold, we should consider a localized appproximation of Laplacian instead of a global one.

To discretize Laplacian locally, let $X_i \subset X$ be a discrete subset of N_i points of X such that X_i contains $x_i \in X$ and $N_i - 1$ nearest points of x_i in X. The analogous subset of Z and trial space are defined as $Z_i = \{z + \delta \mathbf{n} : z \in X\}$ and $V_{Z_i,K} = \mathrm{span}\big(\{K(\cdot, z_j) : z_j \in Z_i\}\big)$, respectively. For each $x_i \in X$, let $u_{Z_i,K,\boldsymbol{\lambda}}$ be the approximation to u_E. Hence, we solve the following system of equations for $\boldsymbol{\lambda}$:

$$\begin{bmatrix} u_E(X_i) \\ \mathbf{0}_{N_i\times 1} \\ \mathbf{0}_{N_i\times 1} \end{bmatrix} = \begin{bmatrix} K(X_i, Z_i) \\ \mathcal{E}_1^x K(X_i, Z_i) \\ \mathcal{E}_2^x K(X_i, Z_i) \end{bmatrix} \boldsymbol{\lambda}, \quad (2.12)$$

where $\mathbf{0}_{N_i\times 1}$ is an $N_i \times 1$ zeros vector and the superscript in $\mathcal{E}_i^x$ means that the operator acts on the first variable of $K(x, y)$. Therefore, the approximation $u_{Z_i,K,\boldsymbol{\lambda}}(x) =$

$\sum_j \lambda_j K(x, z_j)$ satisfies the embedding condition (2.9)-(2.10) and interpolates u_E at X_i. The approximated Laplacian is obtained by taking Laplacian $\mathcal{L} = \Delta$ of $u_{Z_i,K,\boldsymbol{\lambda}}$. i.e.,

$$\Delta_{\mathcal{S}} u_{\mathcal{S}}(x_i) \approx \mathcal{L} u_{Z_i,K,\boldsymbol{\lambda}}(x_i) = \sum_{z_j} \lambda_j \mathcal{L}^x \varphi(x_i, z_j). \tag{2.13}$$

Substituting $\boldsymbol{\lambda}$ of (2.12) into (2.13) yields

$$\begin{aligned} \mathcal{L} u_{Z_i,K,\boldsymbol{\lambda}}(x_i) &= \sum_{z_j} \lambda_j \mathcal{L}^x \varphi(x_i, z_j) \\ &= \mathcal{L}^x K(x_i, Z_i) \boldsymbol{\lambda} \\ &= \mathcal{L}^x K(x_i, Z_i) \begin{bmatrix} \mathcal{L}^x K(X_i, Z_i) \\ \mathcal{E}_1^x K(X_i, Z_i) \\ \mathcal{E}_2^x K(X_i, Z_i) \end{bmatrix}^{\dagger} \begin{bmatrix} u_E(X_i) \\ \mathbf{0}_{N_i \times 1} \\ \mathbf{0}_{N_i \times 1} \end{bmatrix} \\ &=: \begin{bmatrix} \mathbf{w}^T & \widetilde{\mathbf{w}}^T & \widehat{\mathbf{w}}^T \end{bmatrix} \begin{bmatrix} u_E(X_i) \\ \mathbf{0}_{N_i \times 1} \\ \mathbf{0}_{N_i \times 1} \end{bmatrix} = \sum_{x_j \in X_i} w_j u_E(x_j). \end{aligned} \tag{2.14}$$

There is no theory to ensure the matrix in (2.12) is invertible. Hence, we use pseudo-inverse † in (2.14). Now, the approximated Laplacian can be written as a linear combination of the function values of u_E around x_i. It is easily to obtain $\mathbf{w}$ by solving the linear system

$$\begin{bmatrix} K(X_i, Z_i) \\ \mathcal{E}_1^x K(X_i, Z_i) \\ \mathcal{E}_2^x K(X_i, Z_i) \end{bmatrix}^T \begin{bmatrix} \mathbf{w} \\ \widetilde{\mathbf{w}} \\ \widehat{\mathbf{w}} \end{bmatrix} = \mathcal{L}^y K(Z_i, x_i). \tag{2.15}$$

Note that in the above linear system, $\widetilde{\mathbf{w}}$ and $\widehat{\mathbf{w}}$ are unnecessary for the Lapacian approximation. It is because $\Delta u_{Z_i,K,\boldsymbol{\lambda}}(x_i) = \mathbf{w}^T u_E(X_i) + \widetilde{\mathbf{w}}^T \mathbf{0}_{N_i \times 1} + \widehat{\mathbf{w}}^T \mathbf{0}_{N_i \times 1} = \mathbf{w}^T u_E(X_i)$.

In the above discretization of surface Laplacian, we obtained a set of weighting for the approximated surface Laplacian at x_i. The final step is to construct a differentiation matrix D for the surface Laplacian at X such that

$$\Delta_{\mathcal{S}} u_{\mathcal{S}}(X) \approx D u_E(X), \tag{2.16}$$

where D is a sparse matrix having N_i non-zero elements at i-th row. The advantage of local approximation is the resulting matrix is sparse. It takes $N\mathcal{O}(NN_i^3) + \mathcal{O}(Nlog(N))$ to form D while the global approximation requires $\mathcal{O}(N^3)$. Since $N_i \ll N$, the computational cost of local discretization is much more lower than that of global discretization. However, the order of convergence of this local approach is usually much lower than global approximation. It is remarked that discretizing Laplacian using embedding condition (2.9)-(2.9) with the strategy in [11] for the fold Γ_{ij} is not stable with the above discretization. This is the reason why we propose using the graph Laplacian on the fold.

It remains to discretize the diffusion equation on manifold $\mathcal{S}$. By the discretization of surface Laplacian described above, the semi-discrete form of diffusion equation is given by:

$$\frac{\partial u_E(X)}{\partial t} = Du_E(X). \tag{2.17}$$

The semi-discrete form (2.17) is a system of ordinary differential equation and can be solved by some suitably chosen time-integration method.

3 Numerical demonstration

We consider solving diffusion equation on two types of folded surfaces, i.e., flatted sphere and punch in sphere, see Figure 1. Let $X \in \mathcal{S}$ be a discrete set of N points on the folded surface $\mathcal{S}$. Then we apply the proposed method to dicretize the Laplacian so that $\Delta u_{\mathcal{S}}(X) \approx Du_{\mathcal{S}}(X)$ where D is a matrix of discrete Laplacian. The resulting system of ordinary differential equation $\frac{d}{dt}u_{\mathcal{S}}(X) = Du_{\mathcal{S}}(X)$ is solved by second order backward differentiation formula (2-BDFs). In all of our numerics, a scaled multiquadric is used as our kernel K.

Example 1

In this numerical example, we show the convergence behavior when solving diffusion equations by the proposed method. Suppose the unit sphere is parameterized by the spherical coordinate (ϕ, θ). i.e., $x = \sin(\theta)\cos(\phi)$, $y = \sin(\theta)\sin(\phi)$, $z = \cos(\theta)$. It is easy to show by direct calculation that $u_{\mathcal{S}}(t, \theta, \phi) = e^{-2t} cos(\phi)$ is the analytical solution of the diffusion equation on unit sphere. For our case, since $\mathcal{S}$ is a punch-in sphere, we assume the solution of diffusion equation on the punch-in sphere equal to that of unit sphere. Assume $(\theta_{sphere}, \phi_{sphere})$ and (θ, ϕ) be the parameterizations of unit sphere and punch-in sphere, respectively. The solution of the diffusion equation on the punch-in sphere is given by $u_{\mathcal{S}}(t, \theta, \phi) = \cos(\phi_{sphere})$. In Figure 2, The slope 1.00416 indicates that our method converges to the solution linearly. In this case, the number of local neighboring points is set to 15. Generally, the order of convergence will increase as the number of local neighboring points increase. However, increasing the number of local neighboring points will result in instability for the ordinary differential equation. One should apply a hyperviscosity operator for stabilization [12].

Figure 1: Left: Flatted sphere. Right: punch-in sphere.

Example 2

We demonstrate some simulations of diffusion equation on flatted sphere and punch-in sphere with different initial condition. The initial condition are set as follows; first, we assign one on the folded Γ and zeros otherwise. Second, we assign one on half portion of the folded Γ and zeros otherwise. Finally, we assign one on the half portion of the surface $\mathcal{S}$ and zeros otherwise. For comparison, we simulate both flatted sphere and punch-in sphere with the same type of initial condition. In Figure 3, 4 and 5, simulations of flatted sphere are placed on the left column while that of punch-in sphere are placed on the right column. It is easy to see that diffusion property are seen in all of the simulations.

4 Conclusion

We proposed a localized meshless method to solve diffusion equation on folded surfaces. It is an embedding method based on the closest point method with some modifications. We previously proposed a embedding condition for its special data allocation, In this paper, we modify its embedding condition to obtain a higher order embedding condition. This new embedding condition does not require to differentiate the normal vector field of the surface. At the folded of surface, the Laplacian is approximated by a graph Laplacian. Numerical evidence show that our proposed method is convergent linearly.

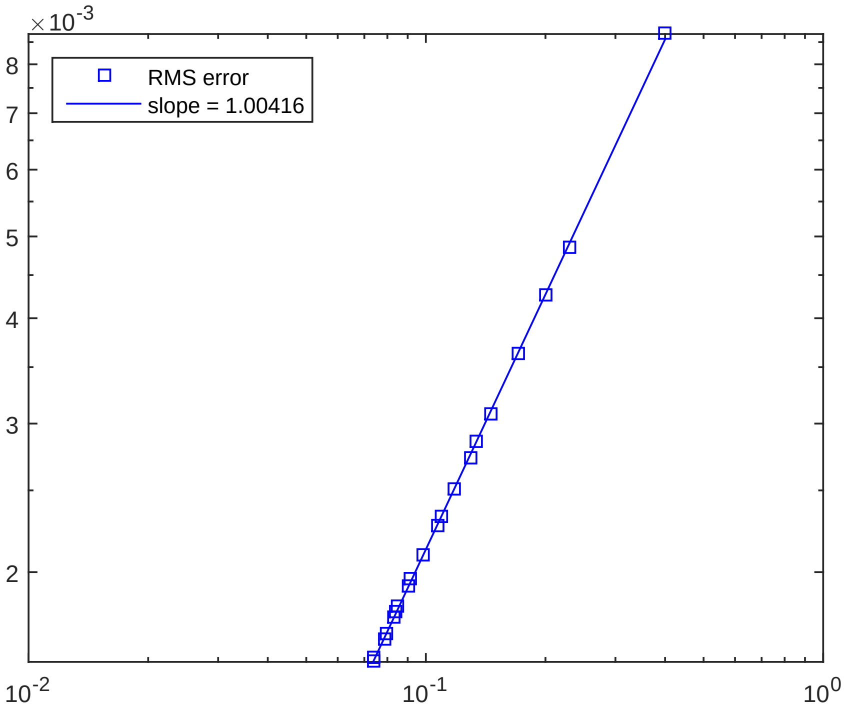

Figure 2: Convergence plot of diffusion on punch-in sphere.

References

[1] Mark AJ Chaplain, Mahadevan Ganesh, and Ivan G Graham. Spatio-temporal pattern formation on spherical surfaces: numerical simulation and application to solid tumour growth. *Journal of mathematical biology*, 42(5):387–423, 2001.

[2] C Varea, JL Aragon, and RA Barrio. Turing patterns on a sphere. *Phys Review E*, 60(4):4588, 1999.

[3] Greg Turk. *Generating textures on arbitrary surfaces using reaction-diffusion*, volume 25. ACM, 1991.

[4] Marcelo Bertalmio, Guillermo Sapiro, Li-Tien Cheng, and Stanley Osher. A framework for solving surface partial differential equations for computer graphics applications. *CAM Report 00-43, UCLA, Mathematics Department*, 2000.

[5] Michael S Floater and Kai Hormann. Surface parameterization: a tutorial and survey. *Advances in multiresolution for geometric modelling*, 1(1), 2005.

[6] Lok Ming Lui, Yalin Wang, and Tony F Chan. Solving pdes on manifolds with global conformal parametriazation. In *Variational, Geometric, and Level Set Methods in Computer Vision*, pages 307–319. Springer, 2005.

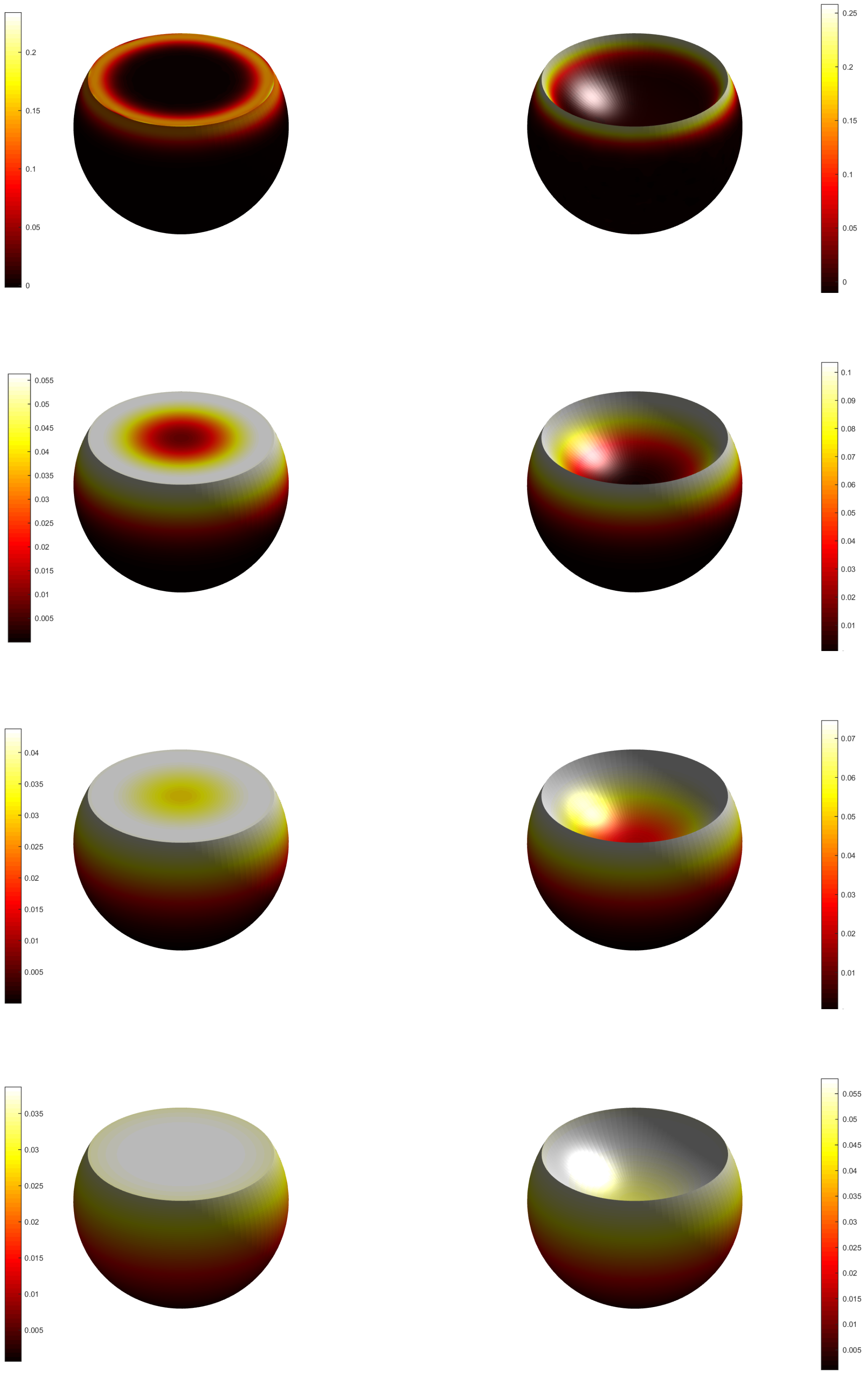

Figure 3: Numerical simulation for different time $t = 0.01, 0.05, 0.1, 0.2$.

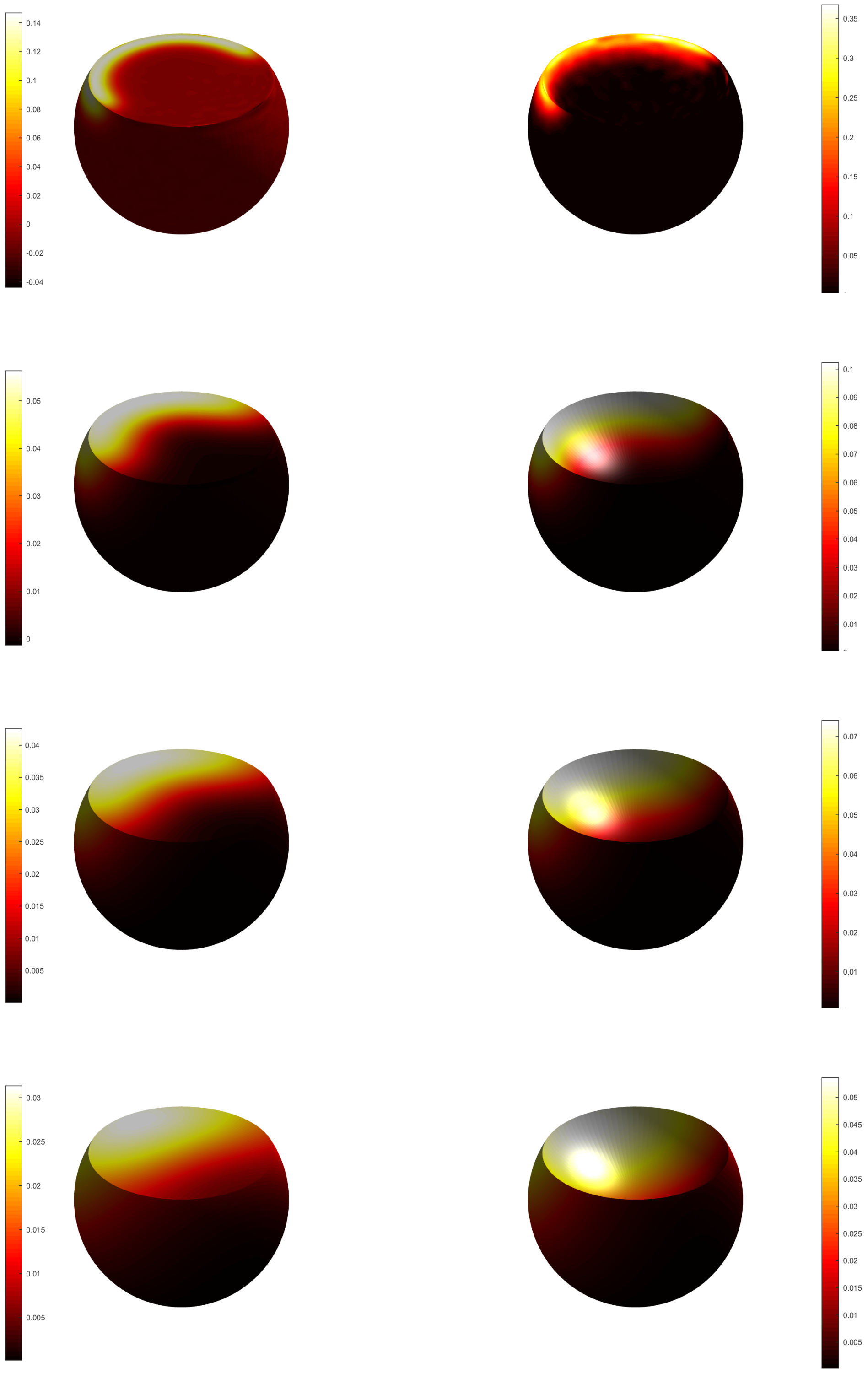

Figure 4: Numerical simulation for different time $t = 0.01, 0.05, 0.1, 0.2$.

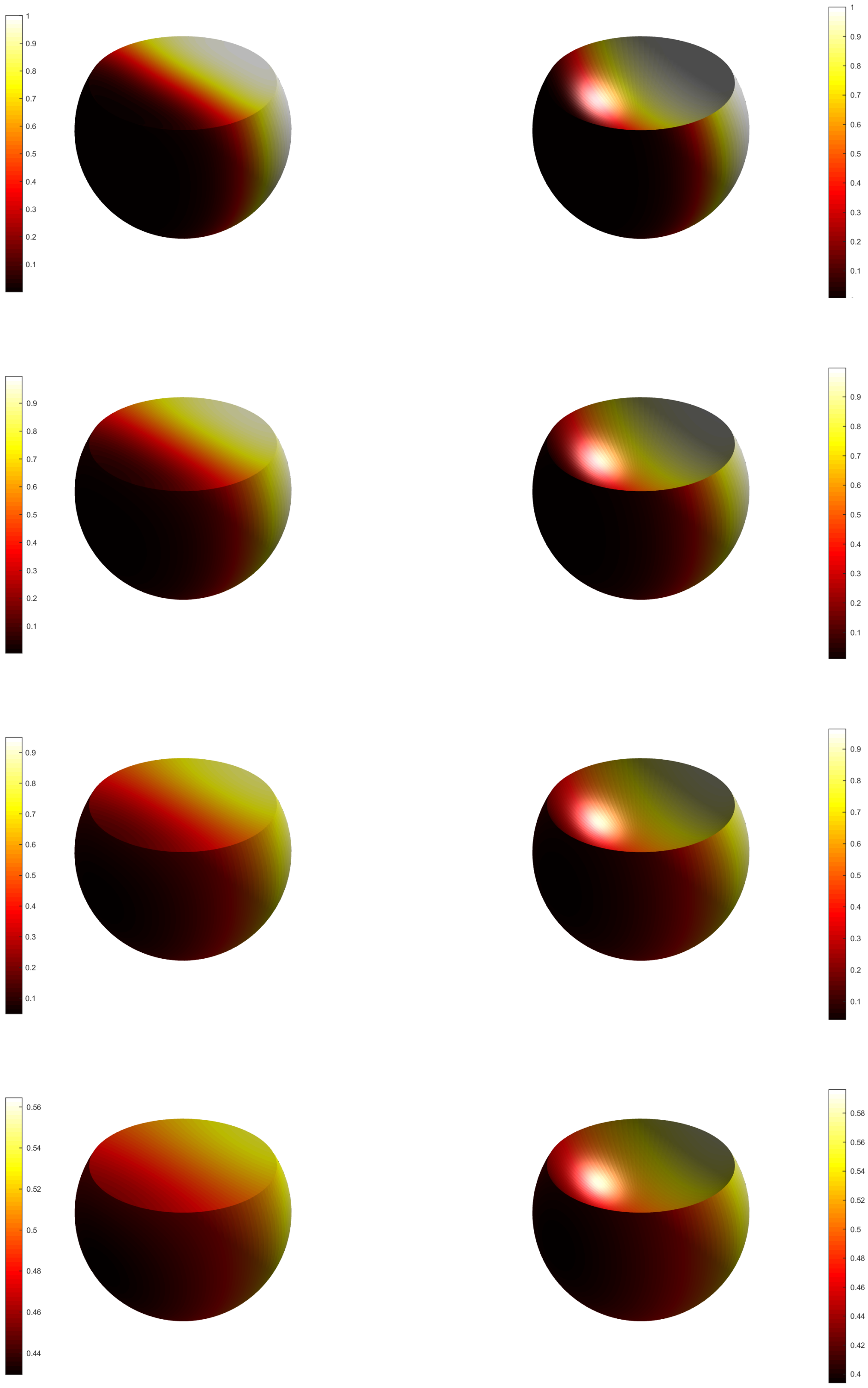

Figure 5: Numerical simulation for different time $t = 0.05, 0.1, 0.2, 1$.

[7] Jos Stam. Flows on surfaces of arbitrary topology. In *ACM Transactions On Graphics (TOG)*, volume 22, pages 724–731. ACM, 2003.

[8] Andrew Witkin and Michael Kass. Reaction-diffusion textures. *ACM Siggraph Computer Graphics*, 25(4):299–308, 1991.

[9] Steven J Ruuth and Barry Merriman. A simple embedding method for solving partial differential equations on surfaces. *J Comput Phys*, 227(3):1943–1961, 2008.

[10] Cécile Piret. The orthogonal gradients method: A radial basis functions method for solving partial differential equations on arbitrary surfaces. *J Comput Phys*, 231(14):4662–4675, 2012.

[11] Ka Chun Cheung, Leevan Ling, and Steven J Ruuth. A localized meshless method for diffusion on folded surfaces. *J Comput Phys*, 297:194–206, 2015.

[12] Bengt Fornberg and Erik Lehto. Stabilization of rbf-generated finite difference methods for convective pdes. *J Comput Phys*, 230(6):2270–2285, 2011.