Eur. Phys. J. B (2016) 89: 181
DOI: 10.1140/epjb/e2016-60960-6

THE EUROPEAN
PHYSICAL JOURNAL B

Regular Article

Numerical simulation and pattern characterization of nonlinear spatiotemporal dynamics on fractal surfaces for the whole-heart modeling applications

Yun Chen[1] and Hui Yang[2,a]

[1] School of Mechanical Engineering, Jiangsu University of Science and Technology, Zhenjiang 212003, P.R. China
[2] Complex Systems Monitoring, Modeling and Control Laboratory, Pennsylvania State University, University Park, PA 16802-4400, USA

Received 14 December 2015 / Received in final form 16 March 2016

Abstract. Engineered and natural systems often involve irregular and self-similar geometric forms, which is called fractal geometry. For instance, precision machining produces a visually flat surface, while which looks like a rough mountain in the nanometer scale under the microscope. Human heart consists of a fractal network of muscle cells, Purkinje fibers, arteries and veins. Cardiac electrical activity exhibits highly nonlinear and fractal behaviors. Although space-time dynamics occur on the fractal geometry, e.g., chemical etching on the surface of machined parts and electrical conduction in the heart, most of existing works modeled space-time dynamics (e.g., reaction, diffusion and propagation) on the Euclidean geometry (e.g., flat planes and rectangular volumes). This brings inaccurate approximation of real-world dynamics, due to sensitive dependence of nonlinear dynamical systems on initial conditions. In this paper, we developed novel methods and tools for the numerical simulation and pattern recognition of spatiotemporal dynamics on fractal surfaces of complex systems, which include (1) characterization and modeling of fractal geometry, (2) fractal-based simulation and modeling of spatiotemporal dynamics, (3) recognizing and quantifying spatiotemporal patterns. Experimental results show that the proposed methods outperform traditional modeling approaches based on the Euclidean geometry, and provide effective tools to model and characterize space-time dynamics on fractal surfaces of complex systems.

1 Introduction

Numerical simulation plays an important role in describing complex phenomena, predicting system behaviors and optimizing control actions to improve the performance of system operations. However, many natural phenomena occur on complex engineered and natural objects that show irregular and self-similar geometric forms, namely fractal geometry. For instance, chemical etching is a space-time process on the surface of silicon wafer [1]. Also, electrical waves propagates on the 2-D tissues of cardiac myocytes [2]. Spatiotemporal simulation provides a better understanding of space-time phenomena and further leads to significant economic and societal impacts. It may be noted that heart disease is responsible for 1 in every 4 deaths in the United States, amounting to an annual loss of $448.5 billion [3]. Simulation-based optimization of cardiac treatments will help improve the quality of healthcare services, reduce healthcare costs and promote the health of our society.

However, most of previous simulation models were developed for space-time dynamics (e.g., reaction, diffusion and propagation) on the Euclidean geometry, e.g., flat planes and rectangular volumes. Traditional simulation models consider finite-difference schemes for space-time computation on finely spaced grids in 3-D spatial dimensions. This brings inaccurate approximation of real-world dynamics on complex surfaces, due to the sensitive dependence of nonlinear dynamical systems on initial conditions. It is well known that many engineered and natural objects show fractal behaviors. For instance, mountain terrain exhibits self-similar geometry across spatial scales. Heartbeat time series shows self-similar patterns across temporal scales [4]. Also, human heart consists of a fractal network of muscle cells, Purkinje fibers, arteries and veins [5]. In the literature, fractal dimension is commonly used to characterize and model the complexity of a fractal object or time series. Fractal dimension describes the self-similar behaviors across spatial or temporal scales and is a quantitative measure of the complexity of fractal patterns. As the complexity of objects increases, a single fractal dimension may not be sufficient for system characterization, but rather multi-fractal spectrums are used. Note that very little has been done to investigate the simulation model of spatiotemporal dynamics (e.g., cardiac electrical excitation and conduction) on the fractal geometry. There is an urgent need to investigate the modeling differences between fractal and Euclidean geometry and further develop effective simulation models of space-time dynamics on complex and irregular surfaces.

---

[a] e-mail: huy25@psu.edu

On the other hand, dimensionality reduction is an effective method to project high-dimensional data into the low-dimensional space, which is widely used in data analysis. However, little has been done to leverage dimensionality-reduction techniques to develop computationally-efficient simulation models of space-time dynamics on fractal surfaces. Notably, high-dimensional fractal surfaces can be isometrically mapped onto a low-dimensional one. As such, space-time simulation of system dynamics can be efficiently conducted in the low-dimensional space, which can then be one-to-one projected onto the original fractal surface. Nonetheless, traditional dimensionality-reduction techniques, e.g., principal component analysis (PCA) and multidimensional scaling (MDS), only preserve the data structure lying on a linear subspace of high-dimensional objects. For instance, PCA identifies a low-dimensional surface that best preserves the data variance as measured in the high-dimensional surface. Classical MDS finds an embedding that preserves inter-point distances, which is equivalent to PCA when those distances are Euclidean. However, fractal surfaces in the high-dimensional space have essential nonlinear structures that pose significant difficulties to PCA and MDS methods. In other words, both PCA and MDS fail to reconstruct true degrees of freedom of fractal surfaces. Notably, isometric feature mapping (ISOMAP) is an extended version of PCA and MDS that handles nonlinear surfaces by preserving geodesic manifold distances [6]. Therefore, we propose to investigate isometric-graphing-based simulation models of space-time dynamics on fractal surfaces.

Furthermore, running simulation model of reaction-diffusion dynamics on fractal surfaces gives rise to spatiotemporal data $\{Y(\boldsymbol{s},t) : \boldsymbol{s} \in R \subset R^d, t \in T\}$ (see Fig. 1), where the dependence of spatial domain R on time T symbolizes the condition where the spatial domain changes over time [7–9]. The next step is to extract useful information from spatiotemporal data. Traditionally, space-time indexed data is analyzed in two ways: (i) spatially-varying time series model $Y(\boldsymbol{s},t) = Y_{\boldsymbol{s}}(t)$, which separates the temporal analysis for each spatial location; (ii) temporally-varying spatial model $Y(\boldsymbol{s},t) = Y_t(\boldsymbol{s})$, which separates spatial analysis for each time point. The first model $Y_{\boldsymbol{s}}(t)$ shows specific interests in time-dependent patterns for each spatial location. The second model $Y_t(\boldsymbol{s})$ focuses more on space-dependent patterns for each time point. However, both approaches are conditional methods studying either the space given time or time given space, and are limited in capturing space-time correlations [10].

This paper presents a new approach to simulate spatiotemporal dynamics of spiral wave formation and turbulent patterns on fractal surfaces. This present approach involves four key steps, namely, (i) fractal surface simulation; (ii) isometric graphing for surface characterization; (iii) dimensionality reduction for reaction-diffusion modeling and (iv) spatiotemporal pattern recognition. First, fractal surfaces generated by random midpoint displacement algorithm in step (i) are characterized by isometric graph approach to extract geodesic distances (i.e., rough estimation of each data point's neighbor on the surface). Second, we simulate and model reaction-diffusion dynamics on the two-dimensional isometric graph, including spiral formation and turbulent patterns. Then, spatiotemporal dynamics are one-to-one projected onto the original fractal surfaces. Third, reaction-diffusion dynamics are investigated on the 3-D heart model to simulate spatiotemporal propagation and conduction of cardiac electrical activity. Finally, we develop new methods to quantify spatiotemporal patterns on irregular surfaces and investigate how fractal characteristics change these patterns.

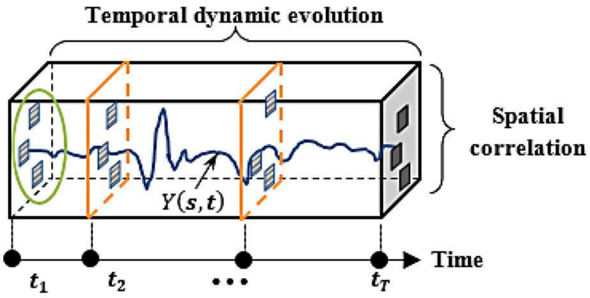


Fig. 1. Spatiotemporal data of reaction-diffusion dynamics.

This paper is organized as follows: Section 2 presents the research background. Section 3 introduces the research methodology. Section 4 shows the first case study – Fitzhugh-Nagumo (FHN) model. Section 5 presents the second one – whole heart modeling, and Section 6 discusses and concludes this investigation.

2 Research background

2.1 Fractal characterization and modeling

Euclidean geometric objects are comprised of lines, planes, cubes, cylinders, spheres with integer dimensions, 1, 2, or 3. However, there are many irregular objects that do not conform to Euclidean geometry. Mandelbrot firstly introduced fractal as "A rough or fragmented geometric shape that can be subdivided in parts, each of which is (at least approximately) a reduced/size copy of the whole" [11]. Note that fractals are not limited to spatial patterns, but can also describe the processes in time. Most of existing works focused on the following two aspects:

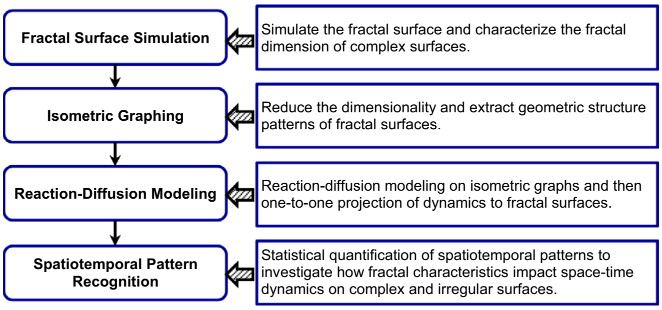
(i) *Characterization of fractal dimension*: the fractal dimension is a statistical measure describing how the patterns change with the scale at which it is measured. Monofractal refers to the homogeneous self-similarity across scales, characterized by a single fractal dimension. Examples of monofractal dimension used for system characterization include physiology [12], gait dynamics [13], and geology [14]. However, multifractal signals or objects require an infinite number of indices (i.e., singularity spectrum) to characterize their scaling properties. Example applications of multifractal spectrum include heart rate variability [4], ECG signals [15], financial markets [16] and material sciences [17]. Note that box counting method is widely used to characterize the fractal dimension [11,18]. In the multifractal case, the probability, P, of a number of measures appearing in a box, i, varies with the box size, ϵ, according to scaling exponents, α_i, which changes over the set, as $\alpha_i \propto \frac{\log P_{i,\epsilon}}{\log \epsilon^{-1}}$. In contrast, the scaling exponent does not change meaningfully over the set for monofractals.

(ii) *Modeling the fractal object or process*: fractals are usually modeled with an iterative or recursive construction or algorithm. Examples of fractal models to generate rough surfaces include shear displacement algorithm [19], diamond-square algorithm [20], and Fourier filtering algorithms [21]. These algorithms are widely used in computer games or movies (e.g., star trek II) to simulate realistic mountains or landscapes. In addition, structure function method, i.e., the random cascade model was utilized to simulate multifractal self-similar behaviors in the heart rate dynamics [22,23]. Heart rate time series $r(t)$ are modeled as a product of J cascade components: $r_J(t) = \prod_{j=1}^{J} \omega_j(t)$ and $\omega_j(t) = 1+\xi_j$, where $\xi_j, j = 1, \ldots, J$ are independent Gaussian variables with $\langle \xi_j \rangle = 0$ and $\langle \xi_i \xi_j \rangle = \delta_{ij}\sigma_j^2$, where δ_{ij} is the Kronecker delta.

However, fractal literature focuses on either the extraction of fractal dimensions for system characterization, or the modeling of fractal geometry. Few, if any, previous studies investigated the simulation model of spatiotemporal dynamics (e.g., cardiac electrical propagation and conduction) on fractal geometry. Further, modeling differences between fractal and Euclidean geometry have not been fully investigated before. It is critical to investigate how fractal characteristics impact the patterns of spatiotemporal dynamics on irregular surfaces.

2.2 Dimensionality reduction

Both simulated and real-world spatiotemporal dynamics bring the proliferation of big data, which is high-dimensional and difficult to visualize and interpret. In order to explore meaningful patterns underlying big data, many previous works developed the methods and tools for dimensionality reduction. Examples of dimensionality reduction approaches include principal component analysis (PCA) [24], multidimensional scaling (MDS) [25], self-organizing map (SOM) [26] and isometric feature mapping (ISOMAP) [6]. PCA uses an orthogonal transformation to find the projected data that captures the principal variations in the original high-dimensional space. Classical MDS transforms the high-dimensional vectors into a low-dimensional embedding that preserves Euclidean distances. Note that the PCA and MDS methods only characterize linear subspaces in the high-dimensional data. However, SOM neural network automatically organizes a low-dimensional map according to the inherent structures in the high-dimensional data. Furthermore, isometric graphing extends the MDS by preserving geodesic manifold distances, which better characterize the degrees of freedom that underlie complex natural observations.



Simulate the fractal surface and characterize the fractal dimension of complex surfaces.

Reduce the dimensionality and extract geometric structure patterns of fractal surfaces.

Reaction-diffusion modeling on isometric graphs and then one-to-one projection of dynamics to fractal surfaces.

Statistical quantification of spatiotemporal patterns to investigate how fractal characteristics impact space-time dynamics on complex and irregular surfaces.

Fig. 2. Flow chart of research methodology.

However, most of previous studies focused on the reduction of high-dimensional data and then the extraction of useful information from the low-dimensional data. Few previous approaches considered the construction of simulation models in the low-dimensional space, and further investigated the projection onto the original high-dimensional space. This will be an original contribution of the present paper that is expected to bring computationally efficient simulation models, and potentially be transplanted to other simulation domains.

3 Research methodology

This present paper studies the numerical simulation of spatiotemporal reaction-diffusion dynamics on irregular surfaces that are generated from the fractal geometry, as opposed to traditional surfaces in the Euclidean geometry. Few, if any, previous investigations have focused on numerical simulation of fractal surfaces in the reduced-dimension space and further characterized the spatiotemporal dynamic patterns on the fractal surfaces. As shown in Figure 2, this present paper is embodied by four core components focusing on numerical simulation on fractal surfaces, including fractal surface simulation, isometric graphing, reaction-diffusion modeling, and spatiotemporal pattern recognition. All four components are eventually integrated together to develop better simulation models on fractal surfaces and better understanding of spatiotemporal phenomena in real-world complex systems.

3.1 Fractal surface simulation

Fractals objects have irregular geometric forms, and cannot be well described using topological dimensions. However, fractal objects often look similar regardless of the magnification, which is so-called self-similar behaviors. Many real-world objects exhibit self-similarity, e.g., scribbles, dust, ocean waves, or clouds. If one zooms in or out the fractal set, there is a similar appearance in the geometric shape. Hence, fractal dimension is introduced

```
Initialization:
    // Start with an square with pixel values at four corners drawn from a Gaussian
    distribution $N(\mu, \sigma^2)$, where $\mu = 0$ and $\sigma = 1$.
While $i \leq$ desired iterations
    Center point:
        // The center is the average of its four neighbors plus a random value $\delta_i$
        generated from a Gaussian distribution $N(\mu, \sigma_i^2)$, where $\sigma_i^2 = \frac{1}{2^{2H(i+1)}}\sigma^2$.
    Edge point:
        // The edge point is the average of its neighbors plus a random value $\delta_i$
        generated from a Gaussian distribution $N(\mu, \sigma_i^2)$, where $\sigma_i^2 = \frac{1}{2^{2H(i+1)}}\sigma^2$.
    $i = i + 1$
End
Calculate the fractal dimension:
    // The simulated fractal surface is monofractal and the fractal dimension $D_F$ is
    determined as $D_F = 3 - H$.
```

Fig. 3. Random midpoint displacement algorithm for the generation of fractal surfaces.

to describe such "infinitely complex" fractal objects (or shape). It is worth mentioning that fractal dimension is not topological, and needs not to be an integer. Fractal sets have theoretical dimensions that exceed their topological dimensions.

As self-similarity across scales is a typical characteristic of fractals, fractal dimension measures the changes of coverings with respect to the scaling factor. It also characterizes the space-filling capacity of a fractal object. In the literature, the box-counting method is widely used to estimate the relationship between scaling and covering so as to estimate the fractal dimension of an irregular object [27]. The basic idea is to cover a fractal set with measure elements (e.g., box) in different scales and examine how the number of boxes changes with respect to the scaling factor. If $N(a)$ is the number of boxes that are needed to cover a fractal object at the scale a, then the fractal dimension D_F specifies how $N(a)$ changes with respect to the scaling factor a as: $N(a) \propto (1/a)^{D_F}$. In general, the box-counting method defines the fractal dimension as

$$D_F := \lim_{a \to 0} \frac{\ln N(a)}{\ln(1/a)}.$$

In order to model space-time dynamics on irregular surfaces, we will first need to generate the fractals. This present investigation utilizes the random midpoint displacement method [20] to generate various types of fractal surfaces. Figure 3 shows the detailed steps of the algorithm, which starts with a square with pixel values at four corners (green triangles in Fig. 4a) drawn from a Gaussian distribution $N(\mu, \sigma^2)$, where $\mu = 0$ and $\sigma = 1$. Then, we recursively generate the center points and edge points at each step until the stopping criterion is satisfied. The fractal surface generated is monofractal and its fractal dimension D_F can be uniquely determined by the parameters (i.e., Hurst exponent H) used in the algorithm.

Three key steps in this recursive algorithm (i.e., center points, edge points and determining fractal dimension) to generate fractal surfaces are described as follows:

Recursive placement of the center point: for each square, a center point (e.g., red dot Fig. 4a) is placed in the middle of X-Y plane, and its height is the average of its four corners (i.e., green triangles in Fig. 4a) plus a random value δ_i generated from a Gaussian distribution $N(\mu, \sigma_i^2)$. Notably, the variance σ_i^2 of random perturbation in the ith iteration is also recursively modified to obtain a fractal Brownian motion (fBm) surface as

$$\sigma_i^2 = \frac{1}{2^{2H(i+1)}}\sigma^2 \tag{1}$$

where $H (0 \leqslant H \leqslant 1)$ is the Hurst exponent. Then, this process is iteratively repeated for each subsquare. Figure 4b shows the second iteration to place four center points (i.e., 4 red dots). Each subsquare is formed by 4 corner points (i.e., green triangles) in Figure 4b. The heights of center points are the average of four corners in each subsquare plus a random perturbation. It may be noted that the first iteration has the biggest influence to the shape of the simulated surface (i.e., large-scale effects), while the following iterations have smaller influence (i.e., small-scale details).

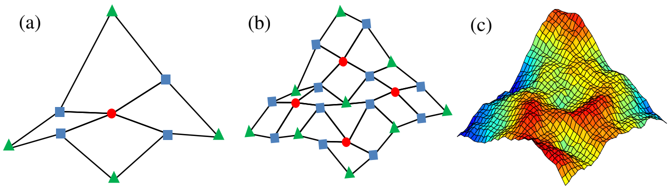


Fig. 4. Recursive steps to generate fractal surfaces; (a) the first iteration, (b) the second iteration, and (c) the fifth iteration.

Recursive placement of edge points: the position of an edge point is in the middle of 2 corner points. The heights of edge points (blue rectangles in Fig. 4a) are calculated as the average of neighbors plus a random perturbation δ_i generated from a Gaussian distribution $N(\mu, \sigma_i^2)$. The variance σ_i^2 is also recursively modified as shown in equation (1). Figure 4a shows the edge points (i.e., 4 blue rectangles) whose heights are calculated as the average of three nearest neighbors. This algorithm leads to nonstationary steps that iteratively generate the fractal surface. Figure 4c shows a fractal surface generated from the algorithm with 33×33 locations, i.e., $\boldsymbol{s}_i = (x_i, y_i, z_i), i = 1, 2, \ldots, 1069$, at the fifth iteration. It may be noted that the surface color represents the height. The hotter color indicates a higher surface, and the cooler color represents a lower surface.

Determine the fractal dimension: the fractal surface provides a better approximation of real-world geometric objects and landscape. Next, the Hurst exponent H uniquely determines the fractal dimension of generated surfaces in the random midpoint displacement algorithm. The singularity spectrum $D(H)$ provides a statistical distribution of Hurst exponents $H(\boldsymbol{s})$ at the locations $\boldsymbol{s}$ such that $H(\boldsymbol{s}) = H$, i.e., $D(H) = D_F(\{\boldsymbol{s}\colon H(\boldsymbol{s}) = H\})$, where D_F is the fractal dimension. As shown in Figure 3, the algorithm iteratively generates random perturbations on the fBm surface and all the locations $\boldsymbol{s}$ have a single Hurst exponent H (also see Eq. (1)). As such, the fractal dimension D_F of fBm surfaces is determined as $D_F = 3 - H$. For example, if the Hurst exponent H is 0.05 in equation (1), then the fractal dimension D_F of fBm

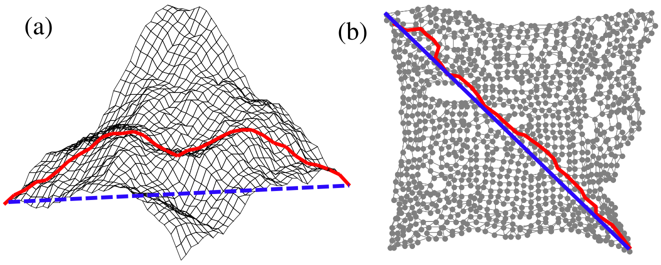


Fig. 5. Euclidean distance (blue dash line) and geodesic distance (red solid line) between two locations on the fractal surface. (b) Isometric graph in the 2-dimensional plane that preserves the geodesic distance on the 3-D fractal surface.

surfaces will be 2.95. If the Hurst exponent H is 0.95 in equation (1), then the fractal dimension D_F of fBm surfaces will be 2.05. In this investigation, we will generate different types of fractal surfaces with the fractal dimensions from 2.05 to 2.95, step size 0.05 to study how fractal characteristics impact spatiotemporal dynamic patterns on complex surfaces.

3.2 Isometric graphing for surface characterization

The fractal surface simulation process is nonlinear and nonstationary, which significantly challenges the characterization of geometric structures and surface patterns. In this present investigation, we introduce the isometric graphing approach [6] to extract inherent geodesic properties of the high-dimensional surface. As shown in Figure 5a, the Euclidean distance between two locations is represented by the blue dash line, and their geodesic distance is shown as red solid line. It may be noted that geodesic distance is the true distance for the reaction-diffusion dynamics (or electrical conduction) to travel between two locations instead of the Euclidean distance. As such, it is imperative to characterize and quantify the intrinsic geometry for reaction-diffusion modeling on the fractal surface. We obtain nonlinear and nonstationary characteristics of fractal surfaces asymptotically by capturing geodesic manifold distances between all the locations. These geodesic distances are further utilized to construct the low-dimensional embedding of fractal surfaces. The proposed approach not only captures the degrees of freedom of fractal surfaces, but also efficiently yields globally optimal solutions for dimensionality reduction. This greatly facilitates the numerical simulation of reaction-diffusion dynamics in the reduced dimension that will be detailed in Section 3.3.

The key steps in the isometric graphing algorithm to characterize the fractal surface and construct the low-dimensional embedding are described as follows:

(1) Build the graph of k nearest neighbours

Define the k-nearest neighbor graph of the fractal surface by connecting each location $\boldsymbol{s}_i$ to its k nearest neighbors $\boldsymbol{s}_j$, $j \in \{k \text{ nearest neighbors of } i\}$. It may be noted that the choice of value k is based on empirical knowledge of the subjects of studies and should be effective and sufficient to preserve the local geometry. Then we compute the distance between locations $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$ as $d(i,j)$, i.e.,

$$d(i,j) = \|\boldsymbol{s}_i - \boldsymbol{s}_j\| = \left((x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2\right)^{\frac{1}{2}}.$$

Otherwise, spatial locations $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$ are disconnected.

(2) Compute the shortest path between two locations $\mathrm{d_G}(\mathrm{i},\mathrm{j})$ on the surface

First, we initialize $d_G(i,j) = d(i,j)$ if locations $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$ are connected in the k-nearest neighbor graph; $d_G(i,j) = \infty$ otherwise. Then, we utilized the Dijkstra's algorithm [28] to compute the shortest path between any two locations. For each $k = 1, \ldots, N$, all entries $d_G(i,j)$ are replaced by $\min\{d_G(i,j), d_G(i,k) + d_G(k,j)\}$. The final matrix $D_G = \{d_G(i,j)\}$ will contain the distance of the shortest path between any two locations on the fractal surface.

(3) Construct low-dimensional embedding

Here, the objective is to derive the isometric graph in the 2-dimensional plane that preserves the shortest distance matrix D_G on the 3-D fractal surface. It may be noted that we are reducing the dimension, while preserving the inherent geodesic properties of fractal surfaces. Let $\tilde{\boldsymbol{s}}_i$ and $\tilde{\boldsymbol{s}}_j$ denote the locations in the isometric graph in the 2-dimensional plane. Then, the objective function is formulated as:

$$\min \sum_{i<j} \left(\|\tilde{\boldsymbol{s}}_i - \tilde{\boldsymbol{s}}_j\| - d_G(i,j) \right); i,j \in [1,\boldsymbol{N}] \qquad (2)$$

where $\|\cdot\|$ is the Euclidean norm. To solve this optimization problem, the Gram matrix M is firstly reconstructed from the $N \times N$ shortest distance matrix D_G: $M = -\frac{1}{2} Q D_G^{(2)} Q$, where the centering matrix $Q = \{\delta_{ij} - 1/N\}_{i,j=1}^{N}$, δ_{ij} is the Kronecker delta. The $D_G^{(2)}$ is a squared matrix and each element is $\left\{d_G^2(i,j)\right\}_{i,j=1}^{N}$ (i.e., the squares of $d_G(i,j)$ in the matrix $D_G^{(2)}$). The element M_{ij} in matrix M is:

$$M(i,j) = -\frac{1}{2}\left[d_G^2(i,j) - \frac{1}{N}\sum_{k=1}^{N} d_G^2(i,k) - \frac{1}{N}\sum_{k=1}^{N} d_G^2(k,j) + \frac{1}{N^2}\sum_{g=1}^{N}\sum_{h=1}^{N} d_G^2(h,g)\right]. \qquad (3)$$

It is known that the Gram matrix M is defined as the scalar product $M = \tilde{S}\tilde{S}^T$, where the matrix $\tilde{S}$ minimizes the aforementioned objective function. The Gram matrix M can be further decomposed as: $M = V\mathit{\Lambda}V^T = V\sqrt{\mathit{\Lambda}}\sqrt{\mathit{\Lambda}}V^T$, where $V = [v_1, v_2, \ldots, v_{D'}]$ is a matrix of eigenvectors and $\mathit{\Lambda} = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_{D'})$ is a diagonal matrix of eigenvalues. Then, the matrix of feature vectors is obtained as: $\tilde{S} = V\sqrt{\mathit{\Lambda}}$. In this present investigation, the reduced dimensionality D' is 2 because we derive the isometric graph in the 2-dimensional plane from the 3-D fractal surface

As shown in Figure 5, the simulated fractal surface (Fig. 5a) is converted into a isometric graph in the two-dimensional plane (Fig. 5b), i.e., $\tilde{\mathbf{s}}_\mathbf{i} = (x_i, y_i), i = 1, 2, \ldots, N$. The isometric graph preserves the shortest path matrix of geodesic distances D_G, thereby preserving the inherent geodesic properties of fractal surfaces. The gray nodes represent all N spatial locations, and the gray lines show the connections between each pair of neighboring locations (with $K = 4$ nearest neighbors and $N = 1089$ data points). Moreover, the geodetic distance of two locations in Figure 5a is embedded in the isometric graph in the two-dimensional plane (see the red solid line in Fig. 5b), where the blue solid line in Figure 5b represents a Euclidean approximation to the true geodetic distance. Therefore, the proposed approach of isometric graphing algorithm not only characterizes the inherent geodesic structure of fractal surfaces, but also provides an optimal solution for dimensionality reduction that facilitates the numerical simulation of reaction-diffusion dynamics in the reduced dimension (see next Sect. 3.3).

3.3 Reaction-diffusion modeling in the reduced dimension

Next, we simulate and model reaction-diffusion dynamics on the two-dimensional isometric graph (see Fig. 5b). This present investigation utilized a two-component reaction-diffusion model on a bounded domain $\mathit{\Omega}$ (i.e., isometric graph) with concentration variables u, v and nonlinear reaction expressions f, g as

$$\frac{\partial u}{\partial t} = f(u,v) + D_1 \nabla^2 u,$$
$$\frac{\partial v}{\partial t} = g(u,v) + D_2 \nabla^2 v, \tag{4}$$

where D_1 and D_2 are the diffusion constants of concentration variables u and v, $\nabla^2 \equiv \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ is the Laplacian operator in the two-dimensional space. We discretize the reaction-diffusion model at the time index $t = 1, 2, \ldots, T$ and locations $i = 1, 2, \ldots, N$ as

$$\left(u_i^t - u_i^{t-1}\right)/\delta t = f\left(u_i^{t-1}, v_i^{t-1}\right) + D_1 \nabla^2 u_i^t,$$
$$\left(v_i^t - v_i^{t-1}\right)/\delta t = g\left(u_i^{t-1}, v_i^{t-1}\right) + D_2 \nabla^2 v_i^t. \tag{5}$$

Then the reaction-diffusion model can be rewritten as follows:

$$\left(1 - D_1 \delta t \nabla^2\right) u_i^t = u_i^{t-1} + \delta t f\left(u_i^{t-1}, v_i^{t-1}\right),$$
$$\left(1 - D_2 \delta t \nabla^2\right) v_i^t = v_i^{t-1} + \delta t g\left(u_i^{t-1}, v_i^{t-1}\right). \tag{6}$$

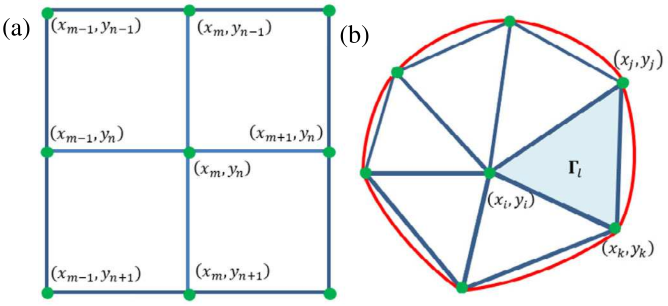


Fig. 6. Illustrations of the domain $\mathit{\Omega}$ with (a) regular grids and (b) irregular grids.

The matrix form of differential equations will become

$$\begin{pmatrix} \boldsymbol{B_1} & 0 \\ 0 & \boldsymbol{B_2} \end{pmatrix} \begin{pmatrix} \boldsymbol{U}^t \\ \boldsymbol{V}^t \end{pmatrix} = \begin{pmatrix} \boldsymbol{U}^{t-1} + \delta t \cdot \boldsymbol{F} \\ \boldsymbol{V}^{t-1} + \delta t \cdot \boldsymbol{G} \end{pmatrix}, \tag{7}$$

where $\boldsymbol{U}^t = [u_1^t, \ldots, u_N^t]^T$ and $\boldsymbol{V}^t = [v_1^t, \ldots, v_N^t]^T$, $\{\boldsymbol{F}\}_i = f(u_i^{t-1}, v_i^{t-1})$ and $\{\boldsymbol{G}\}_i = g(u_i^{t-1}, v_i^{t-1})$ for $i = 1, 2, \ldots, N$, δt is the time step, $\boldsymbol{B_1}$ and $\boldsymbol{B_2}$ are the matrices of constant coefficients pertinent to the domain $\mathit{\Omega}$ and diffusion constants D_1, D_2:

$$\boldsymbol{B_1} = \boldsymbol{I} - D_1 \delta t \cdot \begin{bmatrix} \boldsymbol{\nabla}^2_{u_1} \\ \vdots \\ \boldsymbol{\nabla}^2_{u_N} \end{bmatrix}, \quad \boldsymbol{B_2} = \boldsymbol{I} - D_2 \delta t \cdot \begin{bmatrix} \boldsymbol{\nabla}^2_{v_1} \\ \vdots \\ \boldsymbol{\nabla}^2_{v_N} \end{bmatrix}, \tag{8}$$

where $\boldsymbol{I}$ is the identity matrix of size N, $\boldsymbol{\nabla}^2_{u_i}$ and $\boldsymbol{\nabla}^2_{v_i}$ are scalars of size $1 \times N$, denoting the Laplacian operator of u_i and v_i respectively, i.e., $\boldsymbol{\nabla}^2_{u_i} \cdot \boldsymbol{U}^t = \boldsymbol{\nabla}^2 u_i^t$ and $\boldsymbol{\nabla}^2_{v_i} \cdot \boldsymbol{V}^t = \boldsymbol{\nabla}^2 v_i^t$. Hence, concentration variables $\boldsymbol{U}^t$ and $\boldsymbol{V}^t$ at time step t can be solved through the linear equations $\mathbf{A} \cdot \mathbf{x} = \mathbf{b}$, where $\mathbf{A} = \begin{pmatrix} \boldsymbol{B}_1 & 0 \\ 0 & \boldsymbol{B}_2 \end{pmatrix}$ is a constant matrix and $\mathbf{b} = \begin{pmatrix} \boldsymbol{U}^{t-1} + \delta t \cdot \boldsymbol{F} \\ \boldsymbol{V}^{t-1} + \delta t \cdot \boldsymbol{G} \end{pmatrix}$ that are from the concentration variables $\boldsymbol{U}^{t-1}$ and $\boldsymbol{V}^{t-1}$ at the previous time step $t - 1$. Here, the square matrix $\mathbf{A}$ can be decomposed as $\mathbf{A} = \mathbf{LU}$ through LU factorization, where $\mathbf{L}$ is a lower triangular matrix and $\mathbf{U}$ is an upper triangular matrix. The linear equation $\mathbf{A} \cdot \mathbf{x} = \mathbf{b}$ can be solved by forward and backward substitution [29].

However, it is necessary to formulate the matrices $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$ for the specific domain $\mathit{\Omega}$ and diffusion constants D_1, D_2. As shown in Figure 6a, the discrete Laplacian operator at different locations, i.e., corner, edge and middle, in the regular grid structure can be written as the five-point central difference approximation

Corner:

$$\nabla^2 u^t_{m-1,n-1} = \frac{u^t_{m,n-1} + u^t_{m-1,n} - 2u^t_{m-1,n-1}}{(\delta x)^2},$$

Edge:

$$\nabla^2 u^t_{m-1,n} = \frac{u^t_{m-1,n-1} + u^t_{m,n} + u^t_{m-1,n+1} - 3u^t_{m-1,n}}{(\delta x)^2},$$

Middle:

$$\nabla^2 u^t_{m,n} = \frac{u^t_{m+1,n} + u^t_{m-1,n} + u^t_{m,n+1} + u^t_{m,n-1} - 4u^t_{m,n}}{(\delta x)^2}, \tag{9}$$

where δx is the differences between two adjacent grid points for both x and y directions. Similarly, $\nabla^2 v^t_{m,n}$ can be obtained by substituting v for u in equation (9). For a $N = J \times J$ regular grid, equation (8) can be written as

$$\boldsymbol{B_1} = \boldsymbol{I}_N - \frac{D_1 \delta t}{(\delta x)^2} \cdot \begin{bmatrix} E & \mathbf{I}_J & & & & & \\ \mathbf{I}_J & C & \mathbf{I}_J & & & & \\ & \mathbf{I}_J & C & \mathbf{I}_J & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & \mathbf{I}_J & C & \mathbf{I}_J & \\ & & & & \mathbf{I}_J & C & \mathbf{I}_J \\ & & & & & \mathbf{I}_J & E \end{bmatrix} \tag{10}$$

where $\boldsymbol{I}_J$ is the identity matrix of size J, E and C are $J \times J$ matrices as

$$E = \begin{bmatrix} -2 & 1 & & & \\ 1 & -3 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -3 & 1 \\ & & & 1 & -2 \end{bmatrix}, C = \begin{bmatrix} -3 & 1 & & & \\ 1 & -4 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -4 & 1 \\ & & & 1 & -3 \end{bmatrix}. \tag{11}$$

However, isometric graph in the two-dimensional domain Ω has irregular grid structures. As shown in Figure 6b, the domain $\Omega = \{\Gamma_l\}_{l=1}^L$ involves non-overlapping closed triangles Γ_l, where triangles must intersect along a common edge, a common vertex, or not at all. The five-point central difference approximation in equation (9) is not applicable for the irregular grid. Hence, we introduce the finite element method (FEM) to develop approximate solutions for the matrices $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$.

Let $\phi_1, \phi_2, \ldots, \phi_N$ be the basis functions satisfying $\phi_i(\tilde{\boldsymbol{s}}_j) = \delta_{ij}$, where $\tilde{\boldsymbol{s}}_j, j = 1, 2, \ldots, N$ are locations (triangle vertices) in the domain Ω and δ_{ij} is the Kronecker delta function, i.e., $\delta_{ij} = 1$ when $i = j$, otherwise, $\delta_{ij} = 0$. Then concentration variables u and v can be rewritten as a linear combination of the basis functions, i.e., $u(\tilde{\boldsymbol{s}}) = \sum_{i=1}^N u_i \phi_i(\tilde{\boldsymbol{s}})$. The Laplacian operator of u_i is derived as

$$\nabla^2 u_i = -\sum_{j=1}^N u_j \cdot \left(\int_\Omega \phi_i(\tilde{\boldsymbol{s}}) \, d\Omega \right)^{-1} \times \left(\int_\Omega \nabla\phi_i(\tilde{\boldsymbol{s}}) \cdot \nabla\phi_j(\tilde{\boldsymbol{s}}) \, d\Omega \right) = \boldsymbol{\nabla}^2_{u_i} \cdot \boldsymbol{U}, \tag{12}$$

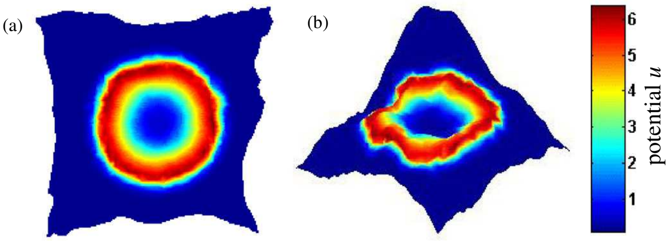


Fig. 7. Reaction-diffusion modeling on the (a) isometric graph and (b) fractal surface.

where $\boldsymbol{U} = [u_1, u_2, \ldots, u_N]^T$. The detailed derivation of equation (12) is in Appendix A.

If we define the *diagonal* lumped mass matrix as

$$\boldsymbol{M} = [M_{ii}]_{i=1}^N = \left[\int_\Omega \phi_i d\Omega \right]_{i=1}^N \tag{13}$$

and the *symmetric* stiffness matrix as

$$\boldsymbol{K} = [K_{ij}]_{i,j=1}^N = \left[\int_\Omega \nabla\phi_i \cdot \nabla\phi_j d\Omega \right]_{i,j=1}^N. \tag{14}$$

Then we have

$$\begin{bmatrix} \boldsymbol{\nabla}^2_{u_1} \\ \vdots \\ \boldsymbol{\nabla}^2_{u_N} \end{bmatrix} = -\boldsymbol{M^{-1}K},$$

and

$$\begin{bmatrix} \boldsymbol{\nabla}^2_{v_1} \\ \vdots \\ \boldsymbol{\nabla}^2_{v_N} \end{bmatrix} = -\boldsymbol{M^{-1}K}$$

according to equation (12). The practical calculation of matrices $\boldsymbol{M}$ and $\boldsymbol{K}$ is given in Appendix B. Therefore, the matrices $\boldsymbol{B_1}$ and $\boldsymbol{B_2}$ can be approximated as

$$\boldsymbol{B_1} = \boldsymbol{I} + D_1 \delta t \cdot \boldsymbol{M^{-1}K}, \quad \boldsymbol{B_2} = \boldsymbol{I} + D_2 \delta t \cdot \boldsymbol{M^{-1}K}. \tag{15}$$

To this end, we can utilize the LU factorization to solve the equation (7) and simulate reaction-diffusion dynamics on the isometric graph with irregular grid structures. Figure 7a shows the $\boldsymbol{U}^t = [u_1^t, \ldots, u_N^t]^T$ at each location $\tilde{\boldsymbol{s}}_i = (x_i, y_i), i = 1, 2, \ldots, N$ at a specific time index t on the isometric graph. Note that geodesic distances between spatial locations are preserved and these locations have the one-to-one correspondence from the isometric graph to the fractal surface. Therefore, reaction-diffusion variables $\boldsymbol{U}^t$ are then one-to-one mapped onto the fractal surface at locations $\boldsymbol{s}_i = (x_i, y_i, z_i)$ (see Fig. 4b). It may be noted that wave patterns are shown to be irregular and chaotic while they are mapped onto the high-dimensional fractal surface.

3.4 Spatiotemporal pattern recognition

New reaction-diffusion model in the reduced dimension provides an effective tool to simulate spatiotemporal dynamics on fractal surfaces, e.g., propagation and conduction of cardiac electrical activity. Next, it is imperative

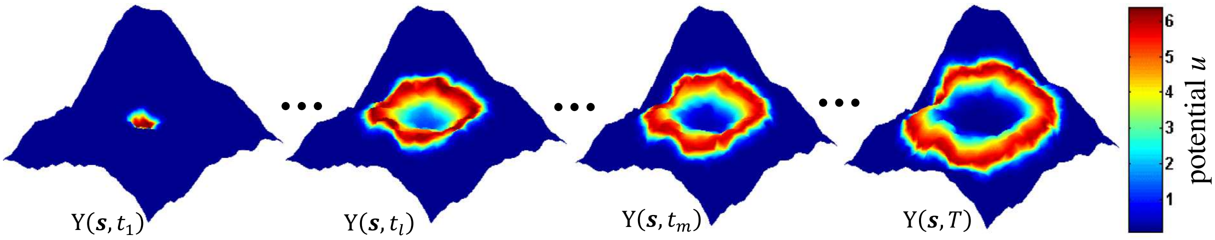


Fig. 8. Spatiotemporal dynamics on the fractal surface at different time indices.

to quantify such spatiotemporal patterns on irregular surfaces and investigate how fractal characteristics change these patterns. In this present paper, we propose manifold learning to extract low-dimensional basis for describing space-time dynamics in the high-dimensional space. This attractive technique leverages similarity or dissimilarity in the spatiotemporal data to make inference about system dynamics that are otherwise not apparent in the high-dimensional space. Manifold learning not only facilitates the visualization of high-dimensional dynamics but also provides statistical quantifiers of spatiotemporal patterns on fractal surfaces. Let $Y(\boldsymbol{s}_i,t), t = 1,\ldots,T$ denote the spatiotemporal data, where $\boldsymbol{s}_i$ is the spatial location $(x_i,y_i,z_i), i = 1,\ldots,N$, for N locations and T is the total number of snapshots forming an ensemble with time index t (see Fig. 8). The similarity or dissimilarity in the spatiotemporal data is quantified by the distance matrix D_T between different time indices.

$$D_T(l,m) = \left[\sum_{i=1}^{N} \left(Y(\boldsymbol{s}_i,t_l) - Y(\boldsymbol{s}_i,t_m)\right)^2\right]^{1/2}, \qquad (16)$$

where $D_T(l,m)$ is a hyper-distance denoting the spatial dissimilarity between two snapshots $Y(\boldsymbol{s}_i,t_l)$, $Y(\boldsymbol{s}_i,t_m)$ with time indexes t_l, t_m, respectively.

The hyper-distance matrix D_T quantifies the dissimilarity of spatial data over time. However, it cannot be directly used as statistical quantifiers of spatiotemporal patterns. Therefore, we propose to treat spatial data at a specific time point, e.g., $Y(\boldsymbol{s},t_l)$, as nodes in the network and D_T as the edge weight between nodes (i.e., adjacency matrix). Very little work has been done to derive the network structure from the hyper-distance matrix D_T. This is a new means to visualize high-dimensional dynamics in the low-dimensional network. In addition, traditional network statistics provide new and effective ways to quantify spatiotemporal patterns on the fractal surface. This present investigation is an extension from our previous works in self-organizing topology of recurrence networks and variable clustering [30,31].

Hence, network structure will be derived from the hyper-distance matrix D_T through the self-organizing process. Let $\boldsymbol{G} = \{\boldsymbol{V},\boldsymbol{E}\}$ be the directed and weighted network, where $\boldsymbol{V}$ is the set of nodes and $\boldsymbol{E}$ is the set of edges. We utilized the spring-electrical model to assign two forces (i.e., attractive and repulsive forces) between nodes. The repulsive force is defined as

$$f_r(l,m) = -\frac{1}{\|\boldsymbol{z}_l - \boldsymbol{z}_m\|^2} e^{D_T(l,m)} \qquad (17)$$

where $\boldsymbol{z}_l$ and $\boldsymbol{z}_m$ are the locations of network nodes l and m. It may be noted that the repulsive force is proportional to the distance between two points. This is because a large repulsive force is expected to separate the two points when they have a large distance. The attractive force is defined as

$$f_a(l,m) = \|\boldsymbol{z}_l - \boldsymbol{z}_m\|^2 e^{-D_T(l,m)}, l \leftrightarrow m. \qquad (18)$$

The attractive force only exists between two connected nodes and is inversely proportional to the distance between them, because a bigger attractive force will pull two nodes closer when they have a smaller distance. The combined force on a node l is the summation of all repulsive forces and attractive forces on the node:

$$f(l,\boldsymbol{z}) = \sum_{l \neq m} -\frac{e^{D_T(l,m)}}{\|\boldsymbol{z}_l - \boldsymbol{z}_m\|^3}(\boldsymbol{z}_l - \boldsymbol{z}_m) + \sum_{l \leftrightarrow m} \|\boldsymbol{z}_l - \boldsymbol{z}_m\| (\boldsymbol{z}_l - \boldsymbol{z}_m) e^{-D_T(l,m)} \qquad (19)$$

where $\boldsymbol{z}_l - \boldsymbol{z}_m$ is the force-directional vector, which is separated from $f_r(l,m)$ and $f_a(l,m)$ to define the direction of combined force $f(l,\boldsymbol{z})$.

The objective of self-organizing process is to optimize spatial locations of network nodes by minimizing the total network energy as: $\mathrm{Min}_{\boldsymbol{z}}\{\sum_{l=1,\ldots,T} f^2(l,\boldsymbol{z})\}$. As a result, the structure of the network is steady with the minimal energy. This, in turn, derives a unique geometry from the hyper-distance matrix D_T. Furthermore, network statistics provide new and effective means to quantify spatiotemporal patterns on the fractal surface. Consider the weighted network $\boldsymbol{G}$ formed by the set of vertices $\{\boldsymbol{z}_l\}_{l=1}^{T}$. Let $d(\boldsymbol{z}_l,\boldsymbol{z}_m)$ denote the shortest path between nodes l and m, which is calculated using the Dijkstra's algorithm [28]. If we assume $d(\boldsymbol{z}_l,\boldsymbol{z}_m) = 0$ for disconnected nodes l and m, the average path length l_G is

$$l_G = \frac{1}{T(T-1)} \sum_{l \neq m} d(\boldsymbol{z}_l,\boldsymbol{z}_m). \qquad (20)$$

This average path length l_G describes the average distance among all nodes in a weighted network. This measure can be used to compare networks with different ranges of weights.

4 Case study 1 – Fitzhugh-Nagumo (FHN) model

In this present investigation, we illustrated and evaluated the proposed methodology for numerical simulation of spiral wave dynamics and turbulent pattern on fractal surfaces. First, we studied the differences of reaction-diffusion dynamics on regular and fractal surfaces. Second, we varied the fractal dimension to generate different types of fractal surfaces. Then we investigated how fractal characteristics change spatiotemporal dynamic patterns on complex surfaces. Experimental design is detailed as follows.

4.1 Experimental design

In order to simulate spatiotemporal dynamics, we utilized nonlinear reaction-diffusion model, i.e., Fitzhugh-Nagumo (FHN) model [32], which is widely used to simulate cardiac electrical conduction on 2-dimensional tissues. Equation (4) is a general formulation of reaction-diffusion model. In this present paper, we specify the functions f and g in FHN kinetic model equation (4) as

$$\begin{aligned} f\,(u,v) &= c_1 u\,(1-u)\,(u-a) - c_2 uv \\ g(u,v) &= b(u-dv) \end{aligned} \tag{21}$$

If we substitute equation (21) into equation (4), the FHN model becomes

$$\begin{aligned} \frac{\partial u}{\partial t} &= c_1 u\,(1-u)\,(u-a) - c_2 uv + D_1 \nabla^2 u \\ \frac{\partial v}{\partial t} &= b\,(u-dv) + D_2 \nabla^2 v \end{aligned} \tag{22}$$

where $D_1 = 1$, $D_2 = 0$, $a = 0.13$, $b = 0.013$, $c_1 = 0.26$, $c_2 = 0.1$, $d = 1.0$ (also see Sect. 3.3), u is the membrane voltage, and v is the recovery variable. It may be noted that membrane voltage u is the dynamic variable used to simulate the electrical propagation and conduction on the fractal surfaces.

In addition to differentiating reaction-diffusion dynamics on regular and fractal surfaces, we generate different types of fractal surfaces with a variety of fractal dimensions (i.e., from 2.05 to 2.95, step size 0.05) to study how fractal characteristics change spatiotemporal dynamic patterns on complex surfaces (see Fig. 9). For each fractal dimension, we generated 10 replicates of surfaces so as to obtain the statistical distribution of modeling performance. Two stimulation protocols were used to generate spiral waves: (1) $S1$ protocol: A basic stimulation ($S1$) is applied at the center of a 33 × 33 grid for 20 time steps. (2) $S1$-$S2$ protocol: After the $S1$ stimulation, a second premature stimulation ($S2$) is applied after 4000 time steps and lasts for 500 time steps at 3 grid points away from the center. These two protocols generate two different types of spiral waves on the fractal surfaces.

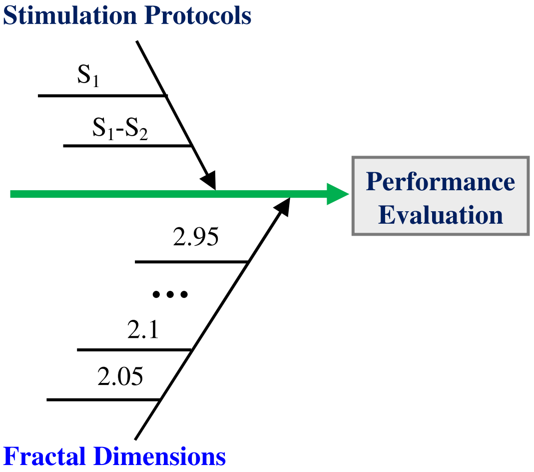


Fig. 9. Experimental design for reaction-diffusion modeling on fractal surfaces.

4.2 Spatiotemporal simulation on regular and fractal surfaces

As shown in Figure 10, we simulate the spiral formation and turbulent patterns on both regular and fractal surfaces. Figures 10a–10d show 2-D snapshots at different time indices of space-time dynamic variable u with spiral formation (which rotates over time) on the regular surface. It may be noted that reaction-diffusion dynamics on the regular surface exhibits the formation of spiral waves and periodic patterns. The reaction-diffusion dynamics starts with a semicircular arc amid the surface (see Fig. 10a), and then diffuses to the edges of the surface (see Fig. 10b). The membrane voltage vanishes at the top of the surface but re-circulate at the bottom (see Fig. 10c), and finally reforms a semicircle in the middle (see Fig. 10d). As a result, spiral waves periodically rotate on the regular surface.

On the other hand, Figures 10e–10h show reaction-diffusion dynamics on the fractal surface. It may be noted that spatiotemporal patterns in these snapshots are different due to the geometric complexity of the fractal surface. Although there are similar patterns as the regular surface, semicircular arcs show distinct patterns in the middle of the surface (see Fig. 10h). The width and length of spiral waves are different. After a sufficiently long time period, spiral wave patterns become more chaotic on the fractal surface that are vastly different from periodic patterns on the regular surface.

4.3 Spatiotemporal pattern recognition

In addition, we investigated how fractal characteristics of a surface impact the spatiotemporal patterns. Figure 11a shows an example of simulated fractal surfaces with the fractal dimension $D_F = 2.05$. As the fractal dimension is close to dimension 2, the simulated fractal surface is flat and similar to 2-D regular surface. We used the $S1$ stimulation protocol to generate reaction-diffusion dynamics on the fractal surface. Figure 11b is the 3-dimensional topology of self-organizing network. Each green node in

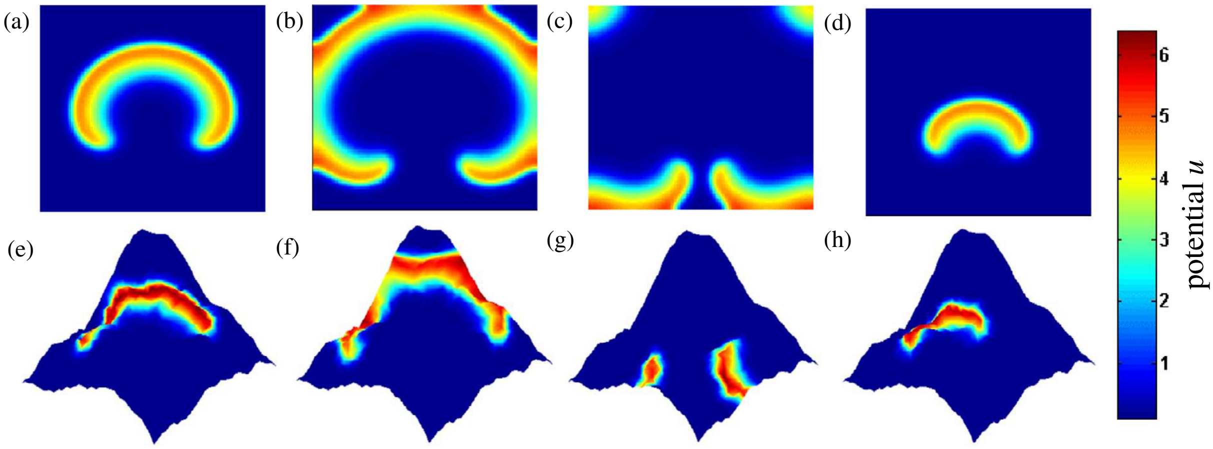


Fig. 10. Snapshots of reaction-diffusion dynamical patterns at different time indices on the regular surface (a)–(d), and the fractal surface (e)–(h).

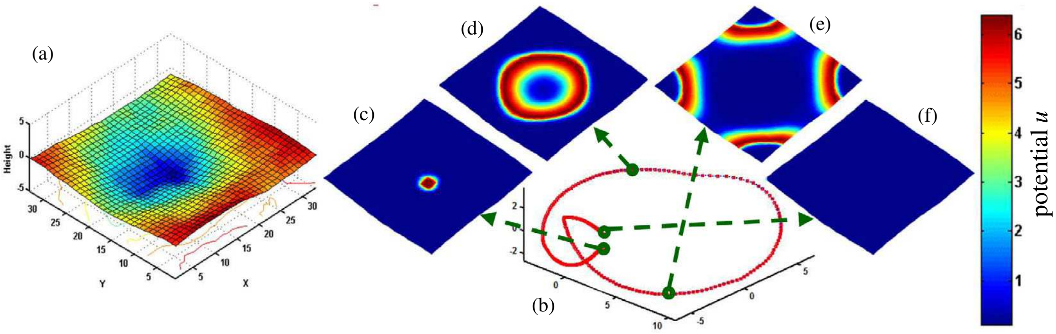


Fig. 11. A simulated fractal surface with fractal dimension $D_F = 2.05$. (b) 3-dimensional topology of self-organizing network. (c)-(f) Frames of spatiotemporal patterns on the fractal surface ($S1$ protocol). Note: Each green node in the network is mapped back to its associated frame.

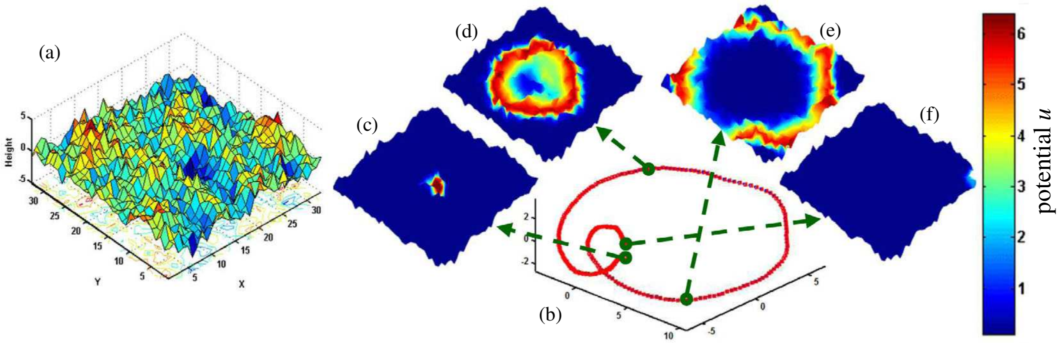


Fig. 12. As Figure 11 for fractal dimension D_F= 2.95.

the network is pertinent to a frame of spatiotemporal images (e.g., Figs. 11c–11f). The movement of nodes along the network trajectory reveals the frame-to-frame differences of spatiotemporal patterns on the fractal surface. The nodes corresponding to Figures 11c and 11f are very close in the network, because Figure 11c is the basic stimulation ($S1$) at the beginning and Figure 11f is pertinent to the vanish waves at the end. Figures 11d and 11e yield smooth waves in the simulation because the fractal dimension 2.05 is close to dimension 2 (i.e., a regular 2-D surface).

On the other hand, Figure 12 shows the results from the surface with fractal dimension D_F= 2.95. Because the fractal dimension is away from dimension 2 (i.e., a regular 2-D surface), the simulated fractal surface is uneven and irregular. As a result, spatiotemporal dynamic patterns in Figures 12c–12f are varied from Figures 11c–11f, although the same $S1$ protocol is used. Although the network trajectory in Figure 12b is visually similar to Figure 11b, there are differences in the beginning and end of the network trajectory. However, Figures 11 and 12 are only visual representations of spatiotemporal dynamics in

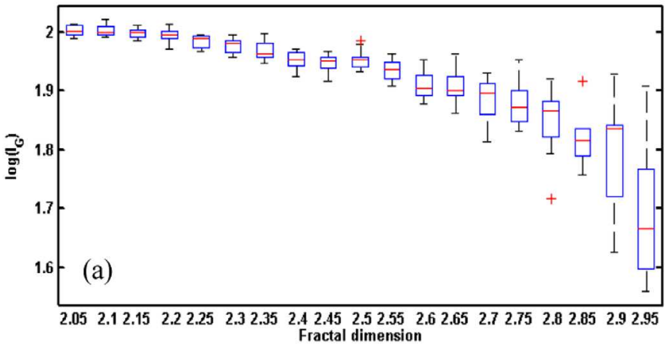


Fig. 13. The network statistics of average path length l_G for spatiotemporal dynamics ($S1$ protocol) on complex surfaces with various fractal dimensions.

the low-dimensional networks. It is necessary to further characterize and quantify dynamical properties of the underlying processes.

Figure 13 shows the distribution of network statistics, i.e., average path length l_G with respect to the variations of fractal dimensions. Notably, we generated 10 replicate surfaces for each fractal dimension. We took the logarithm of two network statistics as shown in the y-axis of the plot (see Fig. 12). Figure 13 shows a decreasing trend of the average path length l_G when the fractal dimension is increased from 2.05 to 2.95. As the fractal dimension gets bigger, the variance of the average path length l_G is also increasing. This is due to the fact that replicate surfaces for a bigger fractal dimension often have large variations. Our experiments evaluated a number of commonly used network measures, e.g., node degree, link density, average path length, diameter and clustering coefficient. It was found that the measure of average path length yields higher correlation with the fractal dimension. This is mainly because fractal dimension impacts the roughness and irregularity of surfaces, thereby impacting the average distances among all nodes in the weighted network. We further calculated the median path length in order to further corroborate the relationship between the path length and fractal dimension. The results showed a similar trend for the variations of the median path length when the fractal dimension is increasing.

In addition to the $S1$ protocol, we have also investigated reaction-diffusion dynamics with the $S1$-$S2$ stimulation protocol, i.e., a premature stimulation ($S2$) after the basic stimulation ($S1$). The fractal surfaces used are the same as in Figure 11a ($D_F = 2.05$) and Figure 12a ($D_F = 2.95$). However, $S1$-$S2$ protocol generates spiral waves that are more complex on the surfaces. Note that the premature stimulation ($S2$) is applied at the location that is 3 grid points away from the center (see Fig. 14b). As such, a semicircular arc appears in the middle of the surface (see Fig. 14c). This semicircular arc continues to propagate after the basic stimulation ($S1$) vanishes at the boundary (see Fig. 14d). However, the semicircular arc will not completely vanish at the boundary. Instead, it will rotate back to the surface to form continuous or self-circulating "spiral waves" on the surface (see Figs. 14e–14g). It is worth mentioning that the network trajectory (see Fig. 14a) is vastly different from the one with the $S1$ protocol in Figure 11b.

Figure 15 shows simulation results of $S1$-$S2$ protocol on the complex surface with fractal dimensions D_F= 2.95 (i.e., the same surface in Fig. 12a). Although Figures 15b–15g are taken at exactly the same time indices as Figures 14b–14g, wave patterns are significantly different with the same $S1$-$S2$ protocol. Note that the path of network trajectory in Figure 15a is varied from Figure 14a, particular in the end of network trajectory. Finally, we also calculated the distribution of network statistics for $S1$-$S2$ protocol, i.e., average path length l_G, when the fractal dimension is increased from 2.05 to 2.95. As show in Figure 16, average path length is decreasing monotonically with the increasing fractal dimension. The variance of network statistics is increasing as the fractal dimension gets bigger. The consistent results in Figures 13 and 16 show that network statistic of average path length l_G effectively characterize the complexity of reaction-diffusion dynamics on fractal surfaces. As such, the approach of self-organizing network provides an effective tool for pattern recognition of spatiotemporal dynamics on the fractal surfaces.

5 Case study 2 – whole heart modeling

5.1 Materials and experimental design

Furthermore, we studied the nonlinear modeling of spiral formation and turbulent patterns on a whole heart. Note that a healthy heart has near-periodic electrical impulses, while cardiac arrhythmia has rapid, disorganized and irregular electrical impulses. Specifically, this study focuses on the numerical simulation of electrical activities for healthy controls and arrhythmias. Recently, computer simulation of electrical conduction and propagation on a human heart is receiving increasing attentions, because it overcomes many practical and ethical limitations in real-world biomedical experiments. In addition, computer simulation offers greater flexibility for biomedical scientists to test their hypothesis and develop new hypotheses for cardiovascular research and knowledge discovery.

Cardiac electrical activity is a series of complex biochemical-mechanical reactions, which involves orchestrated transportation of large amounts of ions through various biological channels. Electrical activity will also propagate from cell to cell through the entire heart to maintain the vital living organism. In the literature, cellular automata and reaction-diffusion models were widely used to model cardiac electrical propagation and conduction processes [33–36]. A cellular automaton is a discrete model with a regular grid of cells, each has a finite number of states. Every cell has the same updating rule based on its neighboring states. Because of its simplicity and superior computational speed, cellular automata was popular in whole heart simulation. However, simplistic assumptions and rules are limited in their ability to model cardiac electrical activity, especially on the irregular surfaces. Reaction-diffusion models describe how dynamic variables

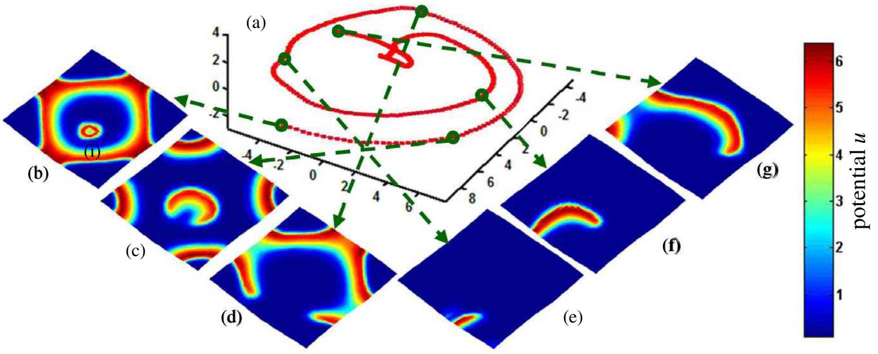


Fig. 14. (a) 3-dimensional topology of self-organizing network. (b)–(g) Frames of spatiotemporal patterns with the $S1$–$S2$ protocol on the surface with fractal dimension $D_F = 2.05$ in Figure 11a. Note: Each green node in the network is mapped back to its associated frame.

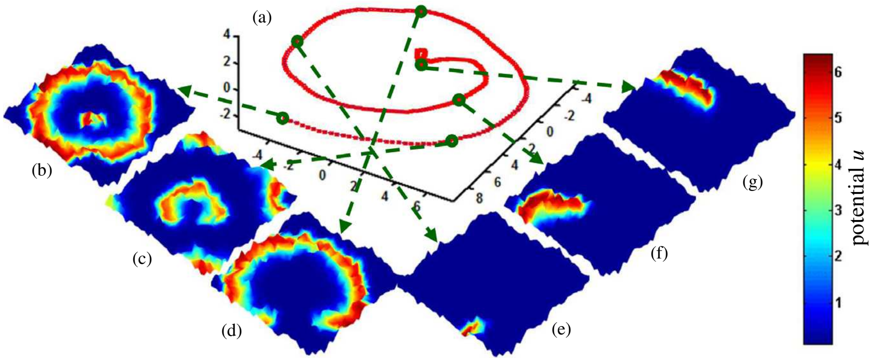


Fig. 15. As Figure 14 for fractal dimension D_F= 2.95.

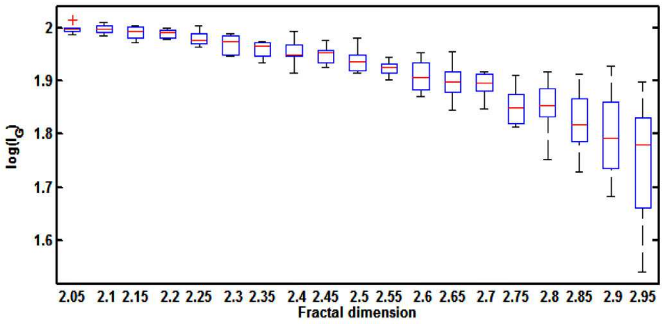


Fig. 16. The network statistic of average path length l_G for spatiotemporal dynamics ($S1$-$S2$ protocol) on complex surfaces with various fractal dimensions.

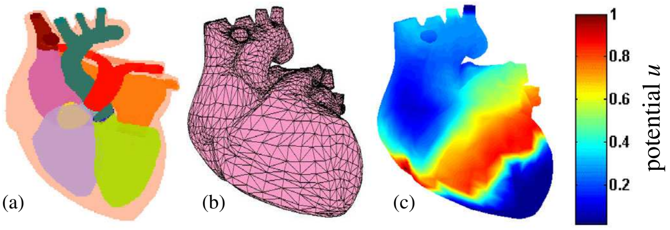


Fig. 17. Whole heart models: (a) colored components of the human heart, (b) finite element meshes, (c) electrical conduction on the surface of the human heart.

change the distribution in space and time under two processes, i.e., (1) *reaction process*: dynamic variables are interacting with each other for conversion, and (2) *diffusion process*: dynamic variables spread out in space. Although reaction-diffusion models provide more realistic simulation of electrical activity through the whole heart, they are more difficult to model on irregular surfaces and more computationally expensive than cellular automata.

This present investigation provides a practical solution to simulate electrical propagation and conduction through the heart. However, a significant challenges resides in the representation of complex geometry of the heart. This representation must not only characterize geometric complexities but also yield sufficient resolution to capture activation wavefronts and cell-to-cell propagation dynamics. As shown in Figure 17, this present investigation utilized an anatomically realistic heart geometry for reaction-diffusion modeling of cardiac electrical activity. However, traditional finite-difference methods (FDM) are effective on regular surfaces with orthogonal and regular grids by discretizing the diffusion tensor in the domain. The complex geometry and high dimensionality of heart (see Fig. 17b) pose great challenges for FDMs. Therefore, we proposed reaction-diffusion modeling in the reduced dimension (see Sect. 3.3) that is designed and developed for irregular grids of fractal surfaces. This new methodology will be demonstrated and evaluated for numerical simulation of space-time electrical dynamics on the complex geometry of a 3-D heart (see Fig. 17c).

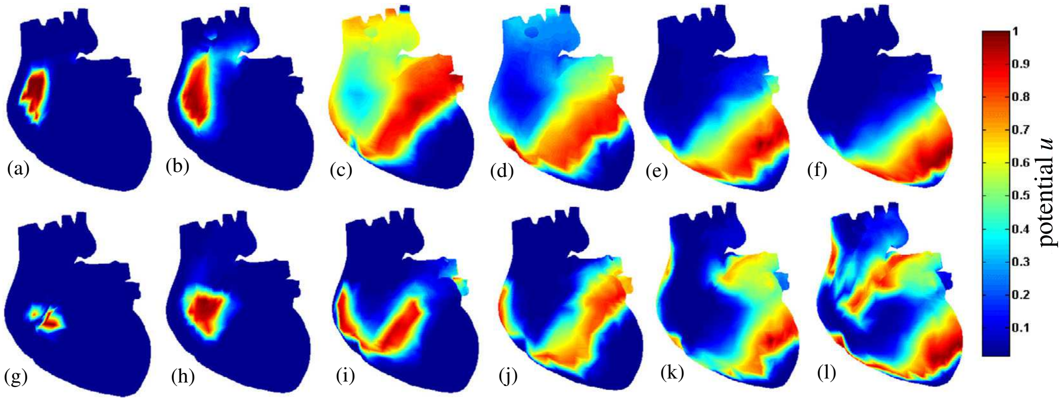


Fig. 18. Spatiotemporal dynamics of electrical conduction at different time indices on the 3-D heart that is healthy (a)–(f), or is with cardiac arrhythmia (g)–(l).

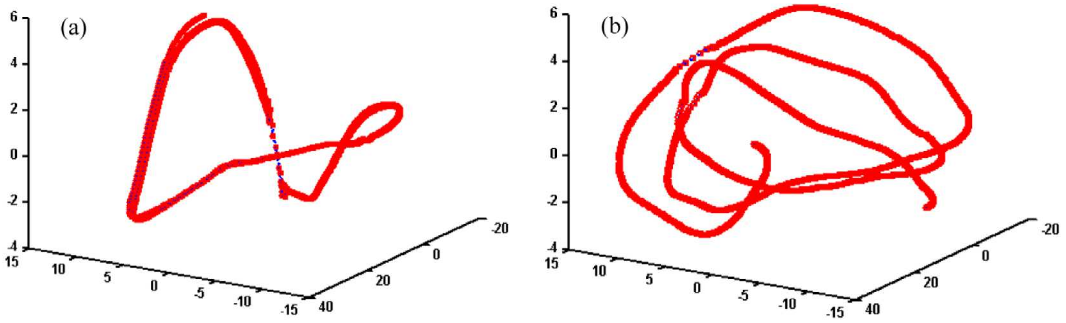


Fig. 19. Self-organizing topology of low-dimensional networks derived from spatiotemporal electrical conduction and propagation of (a) a healthy heart and (b) a heart with arrhythmia.

5.2 Spatiotemporal dynamics of electrical conduction on the 3-D heart surface

Further, we investigated the excitation and propagation of cardiac electrical activity on the 3-D heart surface. Because of the complexity of heart geometry, we leveraged the proposed methodology to address the issues of irregular node spacing, complex boundaries, and regionally dependent conductivities in the 3-D heart model. As shown in Figure 18, the propagation and conduction of electrical activity were simulated and compared between a healthy heart (see Figs. 18a–18f) and a heart with arrhythmia (see Figs. 18g–18l). It may be noted that the healthy heart yields regular and periodic electrical impulses. In the healthy heart, electrical activity is near-periodically paced by the sinoatrial node in the right atrium in each cardiac cycle (see Fig. 18a). Cardiac electric activity propagates throughout the whole heart (see Figs. 18b–18e) and eventually vanishes (see Fig. 18f), then the next cycle starts.

However, electrical excitation does not begin from the sinoatrial node in the simulation of cardiac arrhythmia. Instead, cardiac cycle initiates from another site of the right atrium or muscle cells in the nearby pulmonary veins. As such, the rapid and disorganized electrical impulses occur around atria (see Figs. 18g–18l). As shown in Figure 18, electrical conduction is different between the healthy heart and the arrhythmia one. The snapshots in Figures 18g–18l are taken at the same time indices as Figures 18a–18f. However, traveling patterns of spatiotemporal electrical activity are different. It may be noted that space-time patterns in the arrhythmia heart are more disorganized than the healthy heart. In addition, electrical waves can rotate and self-circulate in the arrhythmia heart without the second stimulus. As such, this causes the heart to fibrillate (commonly known as atrial fibrillations) (see Fig. 18l). Next section will further details the pattern recognition of space-time electrical activities in the whole heart between healthy control and cardiac arrhythmia.

5.3 Spatiotemporal pattern recognition

Figure 18 shows the frames of electrical conduction on the 3-D heart surface over time. As aforementioned, we utilized a hyper-distance matrix D_T to quantify the frame-to-frame dissimilarity. Each frame, e.g., $Y(\boldsymbol{s},t_l)$, is treated as nodes in a network and D_T as the edge weight between nodes (i.e., adjacency matrix). Then, a self-organizing approach is used to derive a stable network topology by iteratively minimizing the energy of the network. Figure 19 shows geometric structures of low-dimensional networks derived from spatiotemporal dynamics of the healthy heart (Figs. 18a–18f) and a heart with arrhythmia (Figs. 18g–18l).

As shown in Figure 19a, network structure of the healthy heart shows a regular and periodic pattern. However, network topology of the arrhythmia heart is disorganized and irregular (see Fig. 19b). As such, the proposed approach of self-organizing network provides a new means to visualize spatiotemporal dynamics in low-dimensional networks, thereby facilitating the characterization and quantification of dynamical properties of the underlying complex processes.

6 Conclusions

Numerical simulation provides a better understanding of complex phenomena, and thereby enables the prediction of system behaviors and optimal control of system operations. However, many engineered and natural systems involve irregular and self-similar geometric forms (i.e., fractal geometry). Very little has been done to investigate the simulation model of spatiotemporal dynamics (e.g., cardiac electrical excitation and conduction) on the fractal geometry. Most of previous simulation models were developed for space-time dynamics (e.g., reaction, diffusion and propagation) on the Euclidean geometry, e.g., flat planes and rectangular volumes. This brings inaccurate approximation of real-world dynamics on fractal surfaces, due to the sensitive dependence of nonlinear dynamical systems on initial conditions. There is an urgent need to investigate the modeling differences between fractal and Euclidean geometry. This present paper developed a suite of methods and tools for numerical simulation of spatiotemporal dynamics on fractal surfaces, including fractal surface simulation, isometric graphing for surface characterization, dimensionality reduction for reaction-diffusion modeling and spatiotemporal pattern recognition.

We illustrated and evaluated the proposed methodology for numerical simulation of spatiotemporal dynamics on fractal surfaces. *First*, we compared the differences of reaction-diffusion dynamics on regular and fractal surfaces. As the geometric complexity of the fractal surface is much higher, spatiotemporal patterns on the fractal surface are vastly different from periodic patterns on the regular surface. *Second*, we generated different types of fractal surfaces to investigate how fractal characteristics change spatiotemporal dynamic patterns on complex surfaces. We leveraged manifold learning to extract low-dimensional basis for describing space-time dynamics on the fractal surface. Each frame is treated as a network node. We utilized a self-organizing approach to derive the network topology from a hyper-distance matrix D_T of the frame-to-frame dissimilarity between nodes (i.e., adjacency matrix). We showed that the distribution of network statistics, i.e., average path length l_G, yields a decreasing trend when the fractal dimension is increased from 2.05 to 2.95. The decreasing trend is consistent for two different types of spatiotemporal dynamics from $S1$ and $S1$-$S2$ protocols. This experiment showed that self-organizing network provides an effective tool for pattern recognition of spatiotemporal dynamics on the fractal surfaces. In addition, network statistics such as average path length l_G effective characterize the complexity of reaction-diffusion dynamics on fractal surfaces.

Further, we developed the whole-heart simulation model to investigate the excitation and propagation of cardiac electrical activity between a healthy heart and a heart with arrhythmia. The proposed simulation methodology effectively addresses the complexity of heart geometry such as irregular node spacing, complex boundaries, and regionally dependent conductivities. Experimental results showed that space-time patterns in the arrhythmia heart are more disorganized than the healthy heart. In addition, electrical waves can rotate and self-circulate in the arrhythmia heart without the second stimulus. This research demonstrated that the proposed methodology outperforms traditional modeling approaches based on the Euclidean geometry, and provide effective tools to model and characterize space-time dynamics on fractal surfaces of complex systems.

Finally, it may be noted that this present investigation focuses on the space-time dynamics on the fractal surface, thus is applicable for real-world case studies such as wild fire on the mountains, heat transfer on the sheet metal, electrical waves on 2-D tissues. However, new models are necessary to investigate space-time dynamics beyond the surface, e.g., pollution that impacts the deep sea and shores. In addition, if the fractal surface is fine-grained with a large number of nodes, the computational complexity will significantly increase. There are 1069 nodes in the fractal surface and 3648 nodes in the heart surface in this present paper. Therefore, the computation can be readily handled by a laptop computer. In general, the computational complexity depends on both surface properties and simulation time, i.e., $O\left(N^2\right) + O(T^2)$, where N is number of total nodes and T is the number of time steps. Our ongoing investigation makes an attempt to extend the methodology to simulate space-time dynamics in the 3-D heart with 728 321 nodes [35]. As a result, real-time computing and visualization become a significant challenge. Therefore, massive parallel computing can be leveraged for large-scale simulation. Last but not least, algorithmic stability is critical to the self-organizing network construction and may depend on the sampling-time of space-time dynamics. However, this present investigation does not encounter stability issues in two representative sets of simulation studies. In the future work, this topic of stability will be further studied analytically and experimentally.

This work is supported in part by the National Science Foundation (CMMI-1646660, CMMI-1617148, CMMI-1619648, and IOS-1146882). The authors also thank Harold and Inge Marcus Career Professorship (HY) for additional financial support.

Appendix A

Corollary: The Laplacian operator of u_i in the domain Ω with the triangular mesh is

$$\nabla^2 u_i = -\sum_{j=1}^{N} u_j \cdot \left(\int_\Omega \phi_i\left(\tilde{\boldsymbol{s}}\right) d\Omega \right)^{-1} \times \left(\int_\Omega \nabla\phi_i\left(\tilde{\boldsymbol{s}}\right) \cdot \nabla\phi_j\left(\tilde{\boldsymbol{s}}\right) d\Omega \right) = \boldsymbol{\nabla}^2_{u_i} \cdot \boldsymbol{U}$$

where $\boldsymbol{U} = \left[u_1, u_2, \ldots, u_N\right]^T$.

Proof: If $\Phi, u \in H(\Omega)$, where $H(\Omega) = \{f: \Omega \rightarrow R, \int_\Omega f^2 d\Omega < \infty\}$ is a function space that all the functions are bounded (i.e., quadratic integrable). The Chain rule from calculus gives that

$$\nabla(\Phi\nabla u) = \nabla\Phi\nabla u + \Phi\nabla^2 u. \tag{A.1}$$

Furthermore, using Gauss's theorem on $\nabla(\Phi\cdot\nabla u)$ we have

$$\int_\Omega \nabla(\Phi\cdot\nabla u)\,d\Omega = \int_{\partial\Omega} (\Phi\cdot\nabla u)\cdot\vec{\mathbf{n}}dS \tag{A.2}$$

where $d\Omega = dx\cdot dy$ is a surface element in Ω, $\vec{\mathbf{n}}$ is the unit normal direction pointing outward at the boundary $\partial\Omega$ with line element dS. Then, we integrate equation (A.1) on both sides and apply equation (A.2) to get

$$\begin{aligned}\int_\Omega \nabla(\Phi\cdot\nabla u)\,d\Omega &= \int_\Omega \Phi\cdot\nabla^2 u d\Omega + \int_\Omega \nabla\Phi\cdot\nabla u d\Omega\\ &= \int_{\partial\Omega} (\Phi\cdot\nabla u)\cdot\vec{\mathbf{n}}dS.\end{aligned} \tag{A.3}$$

Moreover, in our presented research, we assume that the derivatives of concentration variables u and v vanish at the boundary, i.e., $\nabla u = \nabla v = 0$ at $\partial\Omega$. Therefore $\int_{\partial\Omega}(\Phi\cdot\nabla u)\cdot\vec{\mathbf{n}}dS = 0$ and equation (A.3) becomes to

$$\int_\Omega \Phi\nabla^2 u d\Omega = -\int_\Omega \nabla\Phi\nabla u d\Omega. \tag{A.4}$$

Let $\Phi(\tilde{\boldsymbol{s}}) = \phi_i(\tilde{\boldsymbol{s}})$, then equation (A.4) can be rewritten as

$$\begin{aligned}\int_\Omega \phi_i(\tilde{\boldsymbol{s}})\cdot\nabla^2 u(\tilde{\boldsymbol{s}})\,d\Omega &= -\int_\Omega \nabla\phi_i(\tilde{\boldsymbol{s}})\cdot\nabla u(\tilde{\boldsymbol{s}})\,d\Omega\\ &= -\int_\Omega \nabla\phi_i(\tilde{\boldsymbol{s}})\cdot\nabla\left(\sum_{j=1}^N u_j\phi_j(\tilde{\boldsymbol{s}})\right)d\Omega\\ &= -\sum_{j=1}^N u_j\cdot\left(\int_\Omega \nabla\phi_i(\tilde{\boldsymbol{s}})\cdot\nabla\phi_j(\tilde{\boldsymbol{s}})\,d\Omega\right).\end{aligned} \tag{A.5}$$

It may be noted that $\phi_i(\tilde{\boldsymbol{s}}_j) = 0$ for all $j \neq i$. Then, $\nabla^2 u_i$ from equation (A.5) becomes

$$\begin{aligned}\nabla^2 u_i = -\sum_{j=1}^N u_j\cdot\left(\int_\Omega \phi_i(\tilde{\boldsymbol{s}})\,d\Omega\right)^{-1}\\ \times\left(\int_\Omega \nabla\phi_i(\tilde{\boldsymbol{s}})\cdot\nabla\phi_j(\tilde{\boldsymbol{s}})\,d\Omega\right) = \nabla^2_{u_i}\cdot\boldsymbol{U}\end{aligned} \tag{A.6}$$

where $\boldsymbol{U} = [u_1, u_2, \ldots, u_N]^T$.

Appendix B

Numerical calculation of the *diagonal* lumped mass matrix, $\boldsymbol{M} = [M_{ii}]_{i=1}^N = \left[\int_\Omega \phi_i d\Omega\right]_{i=1}^N$, and the *symmetric* stiffness matrix, $\boldsymbol{K} = [K_{ij}]_{i,j=1}^N = \left[\int_\Omega \nabla\phi_i\nabla\phi_j d\Omega\right]_{i,j=1}^N$.

First, the integration on domain Ω is approximated by the sum of integrations on each nonoverlap triangulation Γ_l, i.e., $M_{ii} = \int_\Omega \phi_i d\Omega = \sum_{l=1}^L \int_{\Gamma l}\phi_i d\Omega$ and $K_{ij} = \int_\Omega \nabla\phi_i\nabla\phi_j d\Omega = \sum_{l=1}^L \int_{\Gamma l}\nabla\phi_i\nabla\phi_j d\Omega$.

Second, we illustrate the integration of a triangle Γ_l with vertices $\tilde{\boldsymbol{s}}_i = (x_i, y_i)$, $\tilde{\boldsymbol{s}}_j = (x_j, y_j)$ and $\tilde{\boldsymbol{s}}_k = (x_k, y_k)$ as shown in Figure 6b. First, the basis function $\phi_i(\cdot)$ associated with vertex $\tilde{\boldsymbol{s}}_i$ can be expressed as

$$\phi_i(\tilde{\boldsymbol{s}}) = \frac{\psi_{jk}(\tilde{\boldsymbol{s}})}{\psi_{jk}(\tilde{\boldsymbol{s}}_i)}, \tag{B.1}$$

where $\psi_{jk}(\tilde{\boldsymbol{s}}) = (x - x_k)(y_j - y_k) - (y - y_k)(x_j - x_k)$, and the basis functions for vertices $\tilde{\boldsymbol{s}}_j$ and $\tilde{\boldsymbol{s}}_k$ are defined analogously. Therefore, we have elements of two $N\times N$ sparse matrices $\boldsymbol{M}(\Gamma_l) = Sparse\{M_{pp}(\Gamma_l)\}_{p=i,j,k}$ and $\boldsymbol{K}(\Gamma_l) = Sparse\{K_{pq}(\Gamma_l)\}_{p,q=i,j,k}$ as

$$\begin{aligned}M_{pp}(\Gamma_l) &= \int_{\Gamma_l}\phi_p d\Omega = \frac{|\Gamma_l|}{3},\ p = i,j,k\\ K_{pq}(\Gamma_l) &= \int_{\Gamma_l}\nabla\phi_p\nabla\phi_q d\Omega = \nabla\phi_p\nabla\phi_q|\Gamma_l|,\ \ p,q = i,j,k\end{aligned} \tag{B.2}$$

where $|\Gamma_l| = |x_jy_k - x_ky_j - x_iy_k + x_ky_i + x_iy_j - x_jy_i|/2$ is the area of the triangle Γ_l. And elementary calculations yield

$$\nabla\phi_k\cdot\nabla\phi_i = \frac{(y_j - y_i)(y_k - y_j) - (x_i - x_j)(x_j - x_k)}{\psi_{ji}(\tilde{\boldsymbol{s}}_k)\psi_{kj}(\tilde{\boldsymbol{s}}_i)}, \tag{B.3}$$

with similar expressions obtained for $\nabla\phi_k\nabla\phi_j$, $\nabla\phi_i\nabla\phi_j$, $|\nabla\phi_i|^2$, $|\nabla\phi_k|^2$ and $|\nabla\phi_j|^2$.

Finally, the matrices $\boldsymbol{M}$ and $\boldsymbol{K}$ can be obtained by summing matrices through all triangles $\{\Gamma_l\}_{l=1}^L$ as $\boldsymbol{M} = \sum_{l=1}^L \boldsymbol{M}(\Gamma_l)$ and $\boldsymbol{K} = \sum_{l=1}^L \boldsymbol{K}(\Gamma_l)$.

References

1. U. Phatak, S.T.S. Bukkapatnam, Z. Kong, R. Komanduri, Int. J. Machine Tools Manufact. **49**, 171 (2009)
2. D. Du, H. Yang, S. Norring, E. Bennett, IEEE J. Biomed. Health Inform. **18**, 205 (2013)
3. American Heart Association Writing Group, Circulation **129**, e28 (2014)
4. P.C. Ivanov, L.A.N. Amaral, A.L. Goldberger, S. Havlin, M.G. Rosenblum, Z.R. Struzik, H.E. Stanley, Nature **399**, 461 (1999)
5. A.L. Goldberger, D.R. Rigney, B.J. West, Scientific American **262**, 42 (1990)
6. T.B. Tenenbaum, V. de Silva, J.C. Langford, Science **290**, 2319 (2000)
7. Y. Chen, H. Yang, IEEE Trans. Automat. Sci. Eng. **13**, 215 (2016)
8. Y. Chen, G. Liu, H. Yang, in *Proceedings of 2014 IEEE International Conference on Automation Science and Engineering (CASE), Taipei, Taiwan*, pp. 626–631

9. Y. Ding, E.A. Elsayed, S. Kumara, J.C. Lu, F. Niu, J. Shi, IEEE Trans. Automat. Sci. Eng. **3**, 344 (2006)
10. N. Serban, Ann. Appl. Stat. **5**, 1699 (2011)
11. B.B. Mandelbrot, *The fractal Geometry of Nature* (Freeman, New York, 1982)
12. C.K. Peng, S.V. Buldyrev, S. Havlin, M. Simons, H.E. Stanley, A.L. Goldberger, Phys. Rev. E. **49**, 1685 (1994)
13. J.M. Haussdorff, Y. Ashkenazy, C.-K. Peng, P.C. Ivanov, H.E. Stanley, A.L. Goldberger, Physica A **302**, 138 (2001)
14. W. Dierking, IEEE Trans. Geosci. Remote Sensing **37**, 2397 (1999)
15. X. Yang, X. Ning, J. Wang, Physica A **384**, 413 (2007)
16. A. Turiel, C.J. Perez-Vicente, Physica A **322**, 629 (2003)
17. S. Blacher, F. Brouers, G. Ananthakrishna, Physica A **185**, 28 (1992)
18. N.C. Kenkel, D.J. Walker, Coenoses **11**, 77 (1996)
19. N.S. Lam, H. Qiu, D.A. Quattrochi, C.W. Emerson, Cartography and Geograpghyic Information Science **29**, 25 (2002)
20. A. Fournier, D. Fussel, L. Carpenter, Commun. ACM **25**, 371 (1982)
21. D. Saupe, Algorithms for random fractals, in *The Science of Fractal Images,* edited by H.O. Peitgen, D. Saupe (Spriger-Verlag, New York, 1988), pp. 71–136
22. D.C. Lin, R.L. Hughson, IEEE Trans. Biomed. Eng. **49**, 97 (2002)
23. D.C. Lin, R.L. Hughson, Phys. Rev. Lett. **86**, 1650 (2001)
24. I.T. Jolliffe, *Principal Component Analysis* (Springer-Verlag, New York, 1989)
25. T. Cox, M. Cox, *Multidimensional Scaling* (Chapman & Hall, London, 1994)
26. T. Kohonen, *Self-Organizing Maps* (Springer, New York, 1997)
27. H. Yang, Y. Chen, F. Leonelli, Characterization and Monitoring of Nonlinear Dynamics and Chaos in Complex Physiological Systems, in *Healthcare Analytics: From Data to Knowledge to Healthcare Improvement*, edited by H. Yang, E.K. Lee (Wiley, New York, 2016)
28. E.W. Dijkstra, Numerische Mathematik **1**, 269 (1959)
29. J.M. Rogers, A.D. McCulloch, IEEE Trans. Biomed. Eng. **41**, 743 (1994)
30. H. Yang, G. Liu, Chaos **23**, 043116 (2013)
31. G. Liu, H. Yang, in *IEEE International Conference on Automation Science and Engineering (CASE)* (Gothenburg, Sweden, 2015), pp. 1084–1089
32. A.M. Pertsov, J.M. Davidenko, R. Salomonsz, W.T. Baxter, J. Jalife, Circulation Res. **72**, 632 (1993)
33. A.J. Pullan, M.L. Buist, L.K. Cheng, *Mathematically modelling the electrical activity of the heart: from cell to body surface and back again* (World Science, Singapore, 2005)
34. F.B. Sachse, *Computational cardiology: modeling of anatomy, electrophysiology, and mechanics* (Springer, Berlin, 2004)
35. D. Yu, D. Du, H. Yang, Y. Tu, in *Proceedings of Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE, Chicago, IL, 2014*, pp. 4315–4319
36. Y. Zhang, C. Bajaj, Comput. Methods Appl. Mech. Eng. **195**, 942 (2006)