Transient Response Functions for Complex Graph Similarity Testing

Weibo Gong
Department of Electrical and Computer Engineering,
University of Massachusetts, Amherst, MA0002, U.S.A.

Abstract: Similarity testing for complex graphs has applications in a wide range of problems in network science, computer vision, artificial intelligence and neuroscience, and other engineering areas. Graphs are combinatorial structures and similarity testing poses serious challenges to computation feasibility. In this paper we propose to develop an algorithm analogues to linear system identification based on the transient behavior of the initial condition responses. We propose to collect data from intrinsically parallel random walks to form a graph response function as an effective measure of graph similarity.

Key Words: Monte-Carlo simulation, transient analysis, random walk, graph

1 Introduction

Complex graphs can be used to represent many real world signals such as images, networks, texts, audio pieces and even abstract concepts in the brain. Therefore how to check if two or more such complex graphs are similar is of significance in many areas of engineering and sciences. In this paper we develop an algorithm for complex graph similarity testing. Our algorithm is motivated by the theory of diffusion on manifolds and graphs, is very fast, and is trivially parallel. Our algorithm uses the transient response of a graph to the initial distribution of random walkers to characterize the graph structure. This is analogous to the transient response description of linear dynamic systems.

Graph similarity has been an active research area, various definitions of graph similarity have been proposed (see [7] and references therein). Our definition is different from all previous works in that it is stemmed out from neuroscience motivations. We propose that to compare two complex neuronal networks featured by their connectivity structures the algorithm should be based on random walks that models the neuronal spike motions. Since the random walk motion modes are governed by the spectral parameters of the graphs, it is meaningful to define a similarity criterion using their eigen structure information.

Random walk over complex graphs is analogues to diffusions on manifolds in some aspects. The latter has a rich history and some important results there motivated our algorithm development. Although the approximations between the random walks on continuous manifold and on discrete graph can only be made precise when the graphs are grid-like, many results in continuous cases have similar counterparts in general graphs. In the following we try to view a complex graph with a continuous analogue and try to gain insights into the nature of the problem. For example one could approximate the behavior of a two dimensional planar graph by a real positive function $f(x, y)$ in a bounded domain in $\mathbb{R}^2$ and use the values of this function to approximate the denseness of the connectivity in a neighborhood. Then one can try to develop methods that can effectively test the similarity of two functions $f(x, y)$ and $g(x, y)$. A well-known method to carry out such testing is the Radon transform widely used in computer tomography(CT). In CT one collects the integrals of the test function along straight lines over the domain and apply Radon transform to reconstruct the original function. For similarity testing this is effective since if $f(x, y)$ and $g(x, y)$ differ significantly the line integral data would tell the differences. Now the question is that if we can develop a similar algorithm for complex graphs.

Integration of $f(x, y)$ along a straight line does not seem to have a direct analogue for a graph. However we can think that the normalized $f(x, y)$ values represent annihilation probability (field) of random walk particles. One can run random walk paths through the domain and collect the transient behavior of the random walks, hoping such transient behavior data would contain enough information about the function $f(x, y)$. In the case of graphs such random walk could be discrete walk over the graph and the transient behavior of the random walks could reveal enough graph structure information for similarity testing purpose. In the continuous domain scenario the Feynman-Kac formula for the diffusion type partial differential equations tells us that random walk paths specified by certain stochastic differential equations can be used to find the solution of the corresponding partial differential equations (PDEs). The spectra information of the operators of the PDEs on the one hand governs the motion modes of the random walks, on the other hand is closely related to the geometric information of the domain and the field. In the next section we briefly review the Feynman-Kac formula with an emphasis on the intuitive picture that allows us to think of the analogue with the Radon transform mentioned above.

2 Motivation from continuous random walk and system transient analysis

Consider a smoothly closed domain in a 2D Euclidean space denoted as D with its smooth boundary ∂D. Suppose we have an annihilation field $V(x)$ on this domain and we would like to run a diffusion like random walk over the domain and collect the dynamic movement statistics in order to figure out the geometric features of $V(x)$. This task is not much different from the computer tomography sce-

This work is supported in part by the United States National Science Foundation Grant EFRI-0735974, CNS-1065133 and CNS-1239102, and Army Research Office Contract W911NF-08-1-0233 and W911NF-12-1-03.

nario we mentioned before, except that instead of using the straight lines like in the Radon transform case we try to use the zigzag random walk paths. Would the random walk paths work? Feynman's path integral approach could be applied here to give a hint. Suppose the random walks started from $x \in D$ at time t. The quantity of the random walker (now it is a positive real number so we call it the diffusive stuff) at (x,t) is denoted as $u(x,t)$. We want to know if the dynamics of $u(x,t)$ would tell us about the annihilation field $V(x)$. More specifically, suppose that $D \subset \mathbb{R}^d$ is a smooth bounded domain and that $u(x,t)$ solves the initial value problem

$$\begin{aligned} u_t &= \frac{1}{2}\Delta u - V(x)u, \quad x \in D, \quad t > 0 \quad (1) \\ u(x,0) &= f(x), \quad x \in D \quad (2) \\ u(x,t) &= 0, \quad x \in \partial D, \quad t > 0. \quad (3) \end{aligned}$$

The Feynman-Kac representation of the solution is

$$u(x,t) = E[f(X_t^x)\mathbb{I}_{\gamma^x > t} \exp(-\int_0^t V(X_s^x)ds)] \quad (4)$$

where $X_t^x = x + W_t$ and γ^x is the first time X_t^x hits the boundary.

Intuitive understanding of the path integral is a motivating source for our approach as we mentioned. We now discuss briefly the intuition of this formula. First assume that $V(x) = 0$. Although $f(x)$ is the initial "stuff" to be diffused along the paths generated by $X_t^x = x + W_t$, it can also be viewed as the terminal stuff to be diffused backwards in time. In other words the paths generated by the Wiener process W_t establish a path fabric in the domain using continuous paths. The random walking stuff can be seen as moving on this continuous path fabric either forward in time or backward in time. The weight of the paths remain the same regardless of whether the random walk is forward or backward in time. Thus when the random walking stuff reaches position x at time 0, the expected arrival stuff $E[f(X_t^x)\mathbb{I}_{\gamma^x > t}]$ gives the solution $u(x,t)$. The partial differential equation above can be subject to spectral decomposition that leads to the method of obtaining the spectral information of the domain, as we mentioned before. In fact people have studied the heat content function, defined as $g(t) = \int_D u(x,t)dx$, for obtaining the relevant information about $V(x)$ in, e.g., [10, 11].

The above formula provides several possibilities for graph similarity testing algorithms. We can either let $V(x) = 0$ and use the formula $u(x,t) = E[f(x + W_t)\mathbb{I}_{\gamma^x > t}]$ to suggest a pure diffusion over graph with initial data to reflect the graph connectivity information, or to use $V(x)$ to model the connectivity of the graph, or to combine both the initial condition and the potential function to model the graph connectivity. We will discuss some concrete computer algorithms in later sections.

Practical graph similarity testing algorithm needs to be very fast since complex graphs are computationally demanding for brute force type of comparisons. Can quickness be achieved via zigzag random walk paths? The so-called "small time asymptotics" results in [10, 11] provides enlightenment.

Some historical remarks are in order here. The famous mathematician Mark Kac gave a lecture in 1965 and subsequently a paper in 1966 with the same colorful title "Can one hear the shape of a drum?" [6]. The key observation in Mark Kac's paper is that for smoothly bounded domains in 2 dimensional Euclidean space, heat diffusion from Dirac delta sources could be used to evaluate important geometric parameters of the domain "immediately". More precisely Kac considered the following diffusion equation on a closed domain $D \subset E^2$ with a smooth boundary ∂D:

$$\frac{\partial \rho(y,x,t)}{\partial t} = \frac{1}{2}\frac{\partial^2 \rho(y,x,t)}{\partial y^2}. \quad (5)$$

The initial condition is $\rho(y,x,0) = \delta(y-x)$ and the boundary condition is $\rho(y,t) = 0$ for $y \in \partial D$.

His result is:

$$\sum_{k=1}^{\infty} e^{-\lambda_k t} \sim \frac{\|D\|}{2\pi t} - \frac{L}{4}\frac{1}{\sqrt{2\pi t}} + \frac{1}{6}(1-r) \quad as \quad t \to 0$$

with $\lambda_k, k = 1, 2, \cdots$ the eigenvalues of the diffusion operator, $\|D\|$ the area of the domain, L the length of the boundary ∂D and r the number of smooth holes inside the domain. This is referred to as a small time asymptotic expression ($\sim$). If we know any finite segment, in particular a small early segment, of the time function $\sum_{k=1}^{\infty} e^{-\lambda_k t}$ then we can determine $\|D\|, L, r$ and possibly other geometric parameters of the domain. We emphasize that this expression says that not only we can "hear" some important shape parameters of the "drum", we can hear them "immediately", whose practical implications have not been investigated much.

The fact that system structural information can be obtained "immediately" via a scalar function is actually a familiar one in control and systems theory. Consider a linear time invariant system

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u} \quad (6)$$

where $\mathbf{x} \in \mathbb{R}^n$ is the state vector and $\mathbf{u} \in \mathbb{R}^m$ is the control input vector. For the initial condition response ($\mathbf{u}(t) = 0$) we have (assuming that the eigenvalues of A are all distinct)

$$\mathbf{x}(t) = \mathbf{v}_1 e^{\lambda_1 t} + \mathbf{v}_2 e^{\lambda_2 t} + \cdots + \mathbf{v}_n e^{\lambda_n t} \quad (7)$$

where λ_i is the i-th eigenvalue of the system matrix A and $\mathbf{v}_i$ is the corresponding eigenvector. The initial condition response for

$$\mathbf{x}(0) = \alpha_1 \mathbf{v}_1 + \alpha_2 \mathbf{v}_2 \cdots + \alpha_n \mathbf{v}_n \quad (8)$$

is

$$\mathbf{x}(t) = \alpha_1 e^{\lambda_1 t}\mathbf{v}_1 + \alpha_2 e^{\lambda_2 t}\mathbf{v}_2 + \cdots + \alpha_n e^{\lambda_n t}\mathbf{v}_n. \quad (9)$$

If we let the system output be a scalar $y(t) = \mathbf{c}\mathbf{x}(t)$ then we have

$$y(t) = \alpha_1 e^{\lambda_1 t}\mathbf{c}\mathbf{v}_1 + \alpha_2 e^{\lambda_2 t}\mathbf{c}\mathbf{v}_2 + \cdots + \alpha_n e^{\lambda_n t}\mathbf{c}\mathbf{v}_n. \quad (10)$$

In principle one can recover all the eigenvalues and the coefficients $\alpha_1 \mathbf{c}\mathbf{v}_1$ from a small segment of the $y(t)$. In particular one can use an early segment for this purpose, leading to a quick identification of the critical system parameters. This is analogous to the above diffusion scenario. In fact in the separation of variable solution to the diffusion type of partial differential equations the "time eigenvalues" and "spatial

eigenvalues" are the same. Since the high frequency eigenvectors are not sensitive to the boundary conditions, the high frequency time components of $\int_D u(x,t)dx$ alone would reveal worthy information about the high spatial frequency behavior of the function $V(x)$.

Of course this method is not practical due to the observation noise. However when applied to random walk on complex graphs one can execute many independent walks simultaneously and the law of large numbers will help to battle the noises.

Under the assumption that the operator involved in the diffusion type of equations has distinct real eigenvalues the time function $\int_D u(x,t)dx = g(t)$ is an exponential sum of the form $\sum_{i=1}^{n} \alpha_i e^{-\lambda_i t}$. Thus in principle one can extract the information about the parameters $\alpha_i, \lambda_i, i = 1, \cdots, n$ from a small initial segment of $g(t)$.

We now turn to the description of a graph similarity testing algorithm that resembles the continuous situation where an uniform initial condition and a potential function are used.

3 A fast graph similarity testing algorithm

Our graph is denoted as $G(V, E, W)$ where V is the set of the graph vertex with $|V| = n = N^2$, E is the set of the edges and W is the set of the weights on the edges. Each vertex is representing a pixel in the image Π. We now consider the graph $G = (V, E, W)$ with a nonempty set of boundary vertexes. The adjacency matrix of the graph G is denoted as $A = [w_{v,u}]$ with $w_{v,u}$ being the weight of the edge between the vertexes u and v. The degree matrix is $D = diag\,[d_u]$ with $d_u = \sum_v w_{vu}$. If we let a discrete time random walker to start from a vertex u and move to vertex v with probability w_{vu}/d_u then the movements can be described by a Markov chain

$$P_{k+1} = MP_k \tag{11}$$

with

$$M = AD^{-1}. \tag{12}$$

We call this random walk on graph the "natural" random walk and M the "walk matrix". The boundary vertexes are corresponding to the absorbing states of the Markov chain.

Since our graph G is supposed to reflect realistic neuronal networks (at a high level) rather than a pure mathematical construction, we can assume that the eigenvalues of the matrix M are distinct from each other. Under such assumptions we know M is diagnolizable so we can write

$$M = \Phi\Lambda\Phi^{-1} \tag{13}$$

where Λ is a diagonal matrix with the eigenvalues of M as the diagonal entries. In fact we can write the above equation in the following "spectral decomposition" way:

$$M = \sum_{k=1}^{n} \lambda_k \phi_k \psi_k^T. \tag{14}$$

Here λ_k is the k-th smallest eigenvalue of M, ϕ_k is the corresponding column eigenvector and ψ_k^T is the corresponding row eigenvector. Note that since our matrix M is not symmetric, the eigenvalues and the eigenvectors might have complex conjugate pairs.

Suppose our random walk starts with a uniform initial condition, namely

$$p_0 = [1/n, 1/n, ..., 1/n]^T, \tag{15}$$

the probability vector for the Markov chain evolves as

$$p_{i+1} = Mp_i = \cdots = M^{i+1}p_0 \tag{16}$$

where the first element of p_{i+1} is the sum of the first row of M^{i+1}/n.

We now introduce a key concept in our algorithm development. We call the function

$$g_{i+1} = [r_1, \cdots, r_n]p_{i+1} \tag{17}$$

the *graph response (function)* with

$$r_u = \begin{cases} 1 & \text{if } u \in V/V_B \\ 0 & \text{if } u \in V_B \end{cases}. \tag{18}$$

In other words

$$r = [r_1, \cdots, r_n]^T \tag{19}$$

is a vector with its elements indicating whether the corresponding graph vertex belongs to the boundary vertex set V_B. The importance of graph response is due to

$$\begin{array}{ll} g_{i+1} & = \sum_{u\in V, v\in V} \sum_{k=1}^{n} \lambda_k^{i+1} \phi_k(u) \psi_k^T(v) \\ & = \sum_{k=1}^{n} \lambda_k^{i+1} \sum_{u\in V, v\in V} \phi_k(u) \psi_k^T(v) \\ & = \sum_{k=1}^{n} \alpha_k \lambda_k^{i+1} \end{array} \tag{20}$$

where $\alpha_k = \sum_{u\in V, v\in V} \phi_k(u)\psi_k^T(v)$ and $\phi_{L,k}(u)$ and $\psi_{L,k}^T(v)$)denote the vector components corresponding to vertex u and v, respectively. As can be seen if one has enough values of the function g_i then in principle one can recover the spectral parameters $\alpha_k, \lambda_k, k = 1, \cdots, n$ which describe the dynamics of the random walk and thus reflect the spectral structure of the graph. We emphasize that the values of the function g_i can be estimated from a simulation of the Markov chain with $M = AD^{-1}$. However such simulation is not efficient numerically. Firstly, at each time step of the simulation some vertexes may need to emit a large numbers of walkers. Secondly the functiong_i might drop very quickly in i before we harvest enough g_i values with sufficient accuracy. The well-known lazy walk scheme is suitable here for dealing with both issues.

The random walk that governed by the following transition matrix is called a lazy walk on graph:

$$M_L = (1-\delta)I + \delta M = (1-\delta)I + \delta AD^{-1}. \tag{21}$$

If we execute the lazy walk N steps in $[0, t)$ with the step size $\delta = t/N$ then we have

$$\begin{array}{rcl} M_L^N & = & [(1-\delta)I + \delta AD^{-1}]^N \\ & = & [I - \delta(I - AD^{-1})]^N \\ & = & [I - \frac{t}{N}(D-A)D^{-1}]^N. \end{array} \tag{22}$$

One can see that M_L^N would converge to a meaningful limit. For now we illustrate that the spectrum for the Markov chains M and M_L determines each other for a given step

size δ. Since AD^{-1} is diagnolizable, so is $I - \delta LD^{-1}$ and there exists an nonsingular matrix Φ_L (in fact $\Phi_L = \Phi$ but we keep the notion Φ_L as a reminder) such that

$$M_L^N = \Phi_L \Lambda_L^N \Phi_L^{-1}. \tag{23}$$

In fact

$$\begin{aligned} M_L &= [(1-\delta)I + \delta M] \\ &= [(1-\delta)I + \delta \Phi \Lambda \Phi^{-1}] \\ &= \Phi[(1-\delta)I + \delta\Lambda]\Phi^{-1}. \end{aligned} \tag{24}$$

It can be seen that Λ and Λ_L are uniquely related:

$$\Lambda_L = [(1-\delta)I + \delta\Lambda] = [I - \delta(I - \Lambda)]. \tag{25}$$

Furthermore (23) can be written as

$$M_L^N = \sum_{i=0}^{N} \lambda_{L,k}^i \phi_{L,k} \psi_{L,k}^T \tag{26}$$

where $\psi_{L,k}$ and $\psi_{L,k}^T$ are the column and row eigenvectors corresponding to eigenvalue $\lambda_{L,k}$ for the matrix M_L. It is clear that the information about $\{\alpha_k, k = 1, 2, \cdots\}$ in (20) can be read out from (26) and we have the graph response for the lazy walk $g_{L,i+1}$ as

$$\begin{aligned} g_{L,i+1} &= \textstyle\sum_{k=1}^n \lambda_k^{i+1} \sum_{u \in V, v \in V} \phi_{L,k}(u) \psi_{L,k}^T(v) \\ &= \textstyle\sum_{k=1}^n \alpha_{L,k} \lambda_{L,k}^{i+1}. \end{aligned} \tag{27}$$

The algorithm is to collect all the transitions of the random walkers from one vertex to another for all the vertex pairs in a Markov chain simulation. In other words our algorithm provides a Monte Carlo estimate of the graph response. Since each random walker walks independently in the algorithm, they each produce a sample of an independent random variable in the algorithm. As such the graph response estimate is simply the summation of all the random variable samples generated. When the number of the graph vertexes where random walkers are emitted is large then we are benefitting from the law of large numbers in terms of the variance of the Monte Carlo estimates of the graph response since we "normalize" the result by dividing the summation by the total number of the walkers emitted from all the vertexes. We note that our algorithm could run efficiently on parallel processors. We would also like to point out that when all the $\lambda_k, k = 1, \cdots, n$ are real then the graph response function is monotonic in time. This provide a possibility of dividing the graph into multiple sub-graphs and calculate the graph response function for each sub-graph. When the interactions of the heat flow between sub-graphs are taking into consideration the sum of the graph response functions of the sub-graphs will be the graph response function for the original total graph. Such division could be useful for graph similarity testing at different resolution scales.

4 Ridge factor considerations

The above basic algorithm can have the following generalized formation which is shown to improve the classification performance in experiments and also point to some interesting conjectures in neuroscience.

Suppose we would like to emphasize more the traffic from a heavily connected vertex to a less connected vertex. Here the connectivity of a vertex v is described by the total edge weight dv. It is reasonable to insert a multiplicative factor $(dv/du)^\eta, \eta > 0$ for the traffic from v to u. We call this factor the "ridge factor" since its purpose is to emphasize the connections from heavily connected regions to the less connected ones. In the matrix notations this insertion is amount to change the i step graph response expression from

$$g_i = rM^i p_0 \tag{28}$$

to

$$g_i = rD^{-\eta} M^i D^\eta p_0. \tag{29}$$

Since the degree matrix D is closely related to the transition matrix $M = AD^{-1}$, at a first glance it seems that by inserting such factors we are changing the eigenvalues that govern the motion modes of the random walkers. But this is not true. Note that in multiple step transitions $D^{-\eta}$ and D^η cancels each other so our algorithm simply collect the traffic as before in (27) and then insert the factor $(dv/du)^\eta$ only once:

$$\begin{aligned} g'_{L,i+1} & \\ &= \textstyle\sum_{k=1}^n \lambda_k^{i+1} \sum_{u,v \in V} \phi_{L,k}(u) \psi_{L,k}^T(v) \left(\frac{dv}{du}\right)^\eta \\ &= \textstyle\sum_{k=1}^n \alpha'_{L,k} \lambda_{L,k}^{i+1} \end{aligned} \tag{30}$$

where g' and α' denotes the graph response function and the α coefficients for the corresponding quantities in (27). The choice of η provides a tuning mechanism for classification since in some cases the traffic from the highly connected vertex to the lighter ones is very indicative of the image feature, such as in the case of identifying a photo and a sketch of a person.

The algorithm is scalable in the sense that at every step of the algorithm one updates each edge in the graph and that the computation effort increases linearly in the number of edges. Since the algorithm at each edge is the same and very simple, it is naturally suitable for parallel execution. In fact one of the motivating factor in our development is to have such parallel scheme that is more plausible in the brain than many other computer oriented algorithms.

5 Connections to related works

5.1 Scale space theory in image processing

When uses random walk to gather information of the graph structure it is also important to know which kinds of random walks are suitable in the sense that a random walk should not introduce new structural features into the signal. Previous research on scale space theory has clarified this issue. In the area of image processing it has been noted by Linderberg and others [9] that real world objects are composed of different structures at different spacial scales. As such it is important to generate representations at different spacial scales from the signals that originated from real world objects. Under a set of assumptions called the scale-space axioms that include linearity, shift invariance, semi-group structure, non-enhancement of local extreme, scale invariance and rotational invariance, it can be shown that the Gaussian kernel is the unique smoothing kernel that fits the bill.

Equivalently the scale-space elements can be defined as the diffusion equation

$$\frac{\partial}{\partial t}L(x,y;t) = \frac{1}{2}\nabla^2 L(x,y;t) \tag{31}$$

with the initial condition $L(x,y;0) = f(x,y)$ the given image. This is consistent with the random walk scheme in our proposed graph similarity testing algorithm. The adjust of ridge factor seems to be related to a position dependent diffusion coefficients $D(x)$. The impact to the scale space theory is under investigation.

5.2 Diffusion on graphs

There is an important connection to the line of works with heat diffusion over manifolds and graphs [2–4, 6, 10, 11]. It has a rich history and motivated our current research in quick recognition. To illustrate the connections to this line of works we tentatively assume that the adjacent matrix A is symmetric as in [2]. We denote the normalized graph Laplacian of G as $L_n = D^{-\frac{1}{2}}LD^{-\frac{1}{2}}$ where $L = D - A$ is the graph Laplacian. Now consider the heat flow matrix $h_t = [h_t(u,v)]$ defined by

$$\frac{\partial h_t}{\partial t} = -L_n h_t \tag{32}$$

with a non-empty set of boundary nodes which absorb all arriving heat. Initially the heat strength at every non boundary node is set to 1. In other words we have an heat e^{-tL_n} working with a Dirichlet boundary condition and an uniform initial condition.

Since L_n is a symmetric matrix we have the spectral representation of the solution for the above heat equation over graph:

$$h_t = \sum_{i=1}^{n} e^{-\lambda_i t}\varphi_i\varphi_i^T \tag{33}$$

with $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ the eigenvalues of L_n and $\varphi_i, i = 1, \cdots, n$ the corresponding eigenvectors. The component form of the expression is

$$h_t(u,v) = \sum_{i=1}^{n} e^{-\lambda_i t}\varphi_i(u)\varphi_i(v) \tag{34}$$

and the heat content is the sum of the heat flows between all pairs of the nodes:

$$Q(t) = \sum_{v\in V}\sum_{u\in V} h_t(u,v). \tag{35}$$

It is possible to calculate some leading coefficients in the series expansion of $Q(t)$ at $t = 0$ [11]. These coefficients have been shown to reveal more geometric information about the domain than the eigenvalues alone, and motivated our current study of using random walks in a Monte-Carlo style computation for graph similarity testing.

We note that in [1] an asymmetric heat kernel is used in a quite different way.

5.3 Fokker-Planck operator

There are also interesting connections to the Fokker-Planck equation which could also serve as a continuous analogue for random walk over graphs. Fokker-Planck equation describes the "heat density" $\rho(t,x)$ diffusion under a drifting potential $U(x), x \in \Omega$ with Ω a smoothly bounded domain. The equation is

$$\frac{\partial\rho(t,x)}{\partial t} = \langle\nabla, \rho(t,x)\nabla U(x) + \nabla\rho(t,x)\rangle \tag{36}$$

with a Dirichlet condition for a smooth boundary $\partial\Omega$. The corresponding sample path equation is

$$dx = -\nabla U(x) + \sqrt{2}dw \tag{37}$$

which describes a continuous random walk in the domain. Under suitable conditions the classical gauge transformation enables us to write out the eigendecompostion for the solution of (36) as [12]

$$\rho(t,x) = \sum_{i=0}^{\infty} a_i e^{-\lambda_i t}\phi_i(x) \tag{38}$$

where λ_i, ϕ_i are the i-th eigen pairs for the Fokker-Planck operator in (36). An analogue "graph response (function)" $g(t)$ would be

$$g(t) = \int_\Omega \rho(x)dx = \sum_{i=0}^{\infty} e^{-\lambda_i t}\int_\Omega a_i\phi_i(x)dx = \sum_{i=0}^{\infty}\gamma_i e^{-\lambda_i t} \tag{39}$$

which can be used to describe the eigen structural information of the domain.

We note that under very special circumstances like in [8] one can relate the drifting force and the diffusion term coefficients so that the Fokker-Planck operator becomes a pure diffusion operator in a Riemannian manifold with a metric calculated from the above terms. This suggests that the graph connectivity analogue in continuous random walks can either be the potential function or the Riemannian metric.

5.4 Dynamic system identification

The topic of recovering the shape of a domain from the Laplacian eigenvalues belongs to the spectral inversion problem and has a long history. In fact Kac [6] briefly reviewed this history. It is interesting for us in systems and control to note that originally Hendrik Lorentz formulated the problem as a wave problem in 1910. The issue concerned the formation of standing electromagnetic waves in an enclosure with a perfectly reflecting surface. The conjecture was that for a 2D domain (a membrane) the number of Laplacian eigenvalues less than λ would approach $\|D\|\lambda/2\pi$ when $\lambda \to \infty$ which was proved by Herman Weyl. In the 1960s Mark Kac treated the problem using diffusion theory, making use of the fact that both waves and diffusions are the acting of the same Laplacian operator. While in the wave approach the steady state behavior was the focus of analysis, in the diffusion approach of Kac the subject was the transient behavior of the diffused "stuff". In control engineering there are two main approaches to system identification for a linear time invariant system, namely either to use the sinusoidal input to obtain steady state frequency domain data, or to use a step input or the initial condition for transient time domain data. The latter is less practical in engineering practices due to the noises. However as we discussed earlier it turns out that the time domain transient method becomes quite useful in the

identification of a shape or a graph. The drawback of the transient behavior method can be effectively overcome with the parallel execution of the random walk algorithm on many paths of the domain or the graph.

It seems that much can be learned from the connections between the graph similarity testing algorithms and dynamic system identification. In particular, one notices that the heat diffusion equation on graph is a linear system with the graph Laplacian plays the role of the system dynamics matrix A in $\dot{x} = Ax + Bu$. Indeed we used the heat diffusion $\partial h_t/\partial t = -L_n h_t$ for collecting the initial condition response. We can also think of adding an excitation signal as input and discuss what kind of input is better.

6 The impact of the boundary setting

For a Markov chain M with $m_{ij} = \text{Prob}(j \to i)$ with some absorbing states we can renumber the states such that the transient states come first and to write the transition matrix as the following block matrix:

$$M = \begin{bmatrix} A & 0 \\ B & I \end{bmatrix}.$$

The graph vertexes corresponding to the columns of I are absorbing nodes. In other words the random walkers arrived at one of these nodes will be stuck and not moving any more. On the other hand a random walker start from nodes corresponding to the columns of A say j could move to either a non-absorbing node i by the transition probability a_{ij} or an absorbing node k by the transition probability b_{kj}. However once the random walker arrives at an absorbing node it goes to itself with probability one.

The selection of the boundary nodes in the above algorithm can be seen as the selection of the columns with self loops for the corresponding states. It can be seen that for a very large matrix such choices would not affect the spectral decomposition much. Specifically the eigenvalue interlacing theorems [5] promises that a few link changes would not affect the spectra of complex graphs much.

Absorbing nodes or boundary nodes are important for the graph response functions to pick up the transients that reflect the graph structure. One should select such nodes away from the center of the activities. For example if there are several densely connected clusters in the graph then the absorbing nodes should not be too close to them since the traveling paths of the random walkers should cover these clusters as much as possible. The choice of the absorbing nodes may not affect the eigenvalues which depends on the relative weights of nodes but may affect the eigenvectors and thus the α_ks. A good practical choice in image processing tasks would be to use the frames of the images for absorbing nodes when the image is turned into a graph.

7 Conclusion

One of the magic that our brain does is to recognize familiar objects very quickly. It is a natural consequence of evolution necessity but the engineering mechanism remain a mystery. On the other hand, we in control theory have always understood that an early segment of the initial condition response of a linear time-invariant dynamic system (LTI) is sufficient for the system matrix identification, although it is not a suitable engineering practice due to the noise in the data measurement. Since the brain consists of billions of neurons and synapses, it is possible that a kind of law of large numbers effect could be in action to sufficiently reduce the effect of the noises. Motivated by such thought we develop an algorithm that can quickly test the similarity of very large graphs, and it is intrinsically "embarrassingly" parallel.

References

[1] Fan Chung. The heat kernel as the pagerank of a graph. *Proceedings of the National Academy of Sciences*, 2007.

[2] F.R.K. Chung. *Spectral Graph Theory*. CBMS Lecture Notes. AMS Publication, 1997.

[3] S.-T. Yau F.R.K. Chung. A combinatorial trace formula. *Tsing Hua Lectures on Geometry and Analysis, International Press*, pages 107–116, 1997.

[4] P. Gilkey. Heat content, heat trace, and isospectrality. *Contemporary Math*, 491:115–124, 2009.

[5] WH Haemers. Interlacing eigenvalues and graphs. *Linear Algebra and its Applications*, 226-228:593–616, 1995.

[6] Mark Kac. Can one hear the shape of a drum? *American Mathematical Monthly*, 73(4, part2):1–23, 1966.

[7] George C. Verghese Laura A. Zager. Graph similarity scoring and matching. *Applied Mathematics Letters*, 21(1), 2008.

[8] Daniel Liberzon and Roger W. Brockett. Spectral analysis of fokker-planck and related operators arising from linear stochastic differential equations. *SIAM Journal on Control and Optimization*, 38:38–5, 2000.

[9] Tony Linderberg. On the axiomatic foundations of linear scale-space: Combining semi-group structure with causality vs. scale invariance. *Gaussian Scale-Space Theory: Proc. PhD School on Scale-Space Theory*, 1997.

[10] P. Gilkey M. van den Berg. Heat content asymptotics of a riemannian manifold with boundary. *J. Funct. Anal.*, 120:48–71, 1994.

[11] Patrick McDonald and Robert Meyers. Isospectral polygons, planar graphs and heat content. *Proceedings of the American Mathematical Society*, 131(11):3589–3599, 2003.

[12] H. Risken. *The Fokker-Planck Equation*. Springer, 1989.