Research Paper
Computational Methods for Differential Equations
http://cmde.tabrizu.ac.ir
Vol. 14, No. 3, 2026, pp. 1025-1039
DOI:10.22034/cmde.2025.63178.2816

Introduction to quantum graphs and their stability analysis

Sovan Samanta[1,2,*] **and Tofigh Allahviranloo**[2,3]

[1]Department of Mathematics, Tamralipta Mahavidyalaya, Tamluk, W.B-721636, India.

[2]Research Center of Performance and Productivity Analysis, Istinye University, Istanbul, Türkiye.

[3]Department of Mathematics, Science and Research Branch, Islamic Azad University, Tehran, Iran.

Abstract

Quantum graphs have dynamic vertices and edges. The vertices and edges are assigned by functions. These types of representations of graphs can represent time-dependent network structures. The network characteristics can be visible at any point in time. The networks, such as brain networks or biological disease networks, can be represented by quantum graphs, and hence the characteristics and parameters of quantum graphs are an important research area. This study discusses the properties of quantum graphs. Additionally, real examples and the area of applications are provided.

Keywords. Quantum Graphs, Dynamical Systems, Quantum Networks, Stability Analysis.

2010 Mathematics Subject Classification. 05C22, 81-00,37-02.

1. Introduction

Superposition and Entanglement are two major properties of quantum theory. Superposition represents the dynamical nature of particles, where outsiders can not locate the particles on the quantum level. Entanglement is a connection between quantum particles where the status of one particle is confirmed if the status of another entangled particle is known. These two fundamental properties, Superposition and entanglement, are implemented in the quantum graphs to represent the vertex property and the edge property, respectively. The definition of quantum graphs is given by Allahviranloo and Samanta [3]. Previously, the quantum networks were analysed by several researchers. Quantum random networks [23] are formed based on the principles of quantum mechanics to enhance communication and computation, offering secure and efficient information processing. It is based on random graphs and probability. These networks promise significant advancements in quantum cryptography, teleportation, and superdense coding, though scalability and practical implementation remain challenging. Kottos and Smilansky [17] investigated the spectral properties of graphs with chaos, demonstrating that graphs could be used to study quantum chaos and wave dynamics in complex systems. More recently, Razavinia and Haghighatdoost [24] introduced technical structures related to quantum graphs and quantum computing principles, highlighting the potential of quantum graphs in quantum computing. Tang et al. [31] provided a comprehensive survey on quantum graph computing and learning, emphasizing the advantages of quantum algorithms in solving graph-related problems.

The definition of quantum graphs by Allaviranloo and Samanta [3] can be represented in real networks where vertices and edges are correlated. The vertices also follow some patterns during any interval of time. That study was enough to understand the purpose, still the characteristics of the graphs were not provided. This study focuses on proving some important properties of quantum graphs and analyzing them for possible applications.

Received: 24 August 2024; Accepted: 30 May 2025.
∗ Corresponding author. Email: ssamantavu@gmail.com.

1.1. **Basic properties of quantum theory: Superposition and entanglement.** Quantum mechanics, the foundation of quantum theory, is characterized by several unique properties that distinguish it from classical physics. Two of the most important properties are superposition and entanglement.

Superposition refers to the ability of a quantum system to exist in multiple states simultaneously. This principle is famously illustrated by Schrödinger's cat [29], a thought experiment in which a cat in a box is simultaneously alive and dead until observed. Mathematically, if a quantum system can be in states $|0\rangle$ and $|1\rangle$, it can also be in any linear combination of these states, such as $\alpha|0\rangle + \beta|1\rangle$, where α and β are complex numbers [30].

Entanglement is another fundamental property of quantum mechanics, describing a situation in which the states of two or more particles are so interconnected that the state of one particle cannot be described independently of the others. When particles are entangled, measuring the state of one particle instantaneously determines the state of the others, regardless of the distance between them. This phenomenon, which Einstein famously referred to as 'spooky action at a distance,' [6] has been experimentally verified and is a cornerstone of quantum information theory.

1.2. **Framework of this study.** This paper begins with an introduction to the concept of quantum graphs, highlighting their significance and the motivation for their development. The study includes a review of literature (section 2). The paper discusses the mathematical definition and properties of quantum graphs (section 3), stability of quantum graphs (section 4), a small case study (section 5), quantum networks (section 6), concluding with a summary of findings and future research directions in the last section. The symbols and notations are given in Table 1.

Table 1. List of symbols and notations.

Symbol	**Description**
$\|0\rangle, \|1\rangle$	Basis states in quantum theory
α, β	Complex coefficients in superposition
$\alpha\|0\rangle + \beta\|1\rangle$	Superposition state
$g_{ij}(t)$	Correlation function between vertices v_i and v_j
$h(x, y)$	Function defining the nature of entanglement
$\text{dweight}_i(t)$	Dynamic weight of vertex v_i at time t
ω_i, ϕ_i	Frequency and phase shift for dynamic weight
$A_{ij}(t)$	Adjacency matrix element at time t
$G = (V, E)$	Graph with vertices V and edges E
$d_i(t)$	Degree of node v_i at time t

2. Literature review

These quantum properties have inspired the development of quantum graphs, which extend classical graph theory into the quantum domain. Quantum graphs incorporate dynamic vertices and potentially entangled edges, providing a framework to model quantum systems more accurately [8, 17].

Kottos and Smilansky (1997) [17] started the basics of spectral properties of graphs with chaos. The paper "Cambridge Quantum Network" by Dynes et al. (2019) [10] presents a significant advancement in the dynamics of QKD network.

Graphs have a wide range of applications, from quantum computing to complex systems. In quantum computing, graphs can model the interactions between qubits, providing insights into the behavior of quantum algorithms and error correction techniques [18]. In quantum communication, graphs help understand entanglement distribution and information transfer in quantum networks. By modeling the dynamic and entangled nature of quantum communication channels, the authors developed efficient and secure quantum communication protocols [5].

Entangled edges in quantum graphs can model the correlated activity between different brain regions. In the brain, neural connections are not static; they can strengthen or weaken over time, and their activity can be highly correlated, and the details can be found in [7]. Bassett and Bullmore's paper [4] on small-world brain networks provides a comprehensive analysis of the structural and functional connectivity in the brain, highlighting the efficiency of small-world properties.

Graphs with nodes and edges having membership values, termed as fuzzy graphs, are perfect to represent any uncertain networks. Das et al. [9] analysed link prediction within co-authorship networks under conditions of fuzziness, applying these concepts to biomedical analysis, thereby showcasing the dynamic nature of weighted graphs in academic collaborations. Samanta et al. [26] presented a mathematical framework for representing competitions through competition cluster hypergraphs, illustrating the dynamic and weighted interactions within competitive environments. Samanta and Sarkar [25] investigated generalized fuzzy graphs, emphasizing their dynamic and weighted properties, which are crucial for modeling complex systems with varying degrees of uncertainty. Sarkar and Samanta [28] explored the structure and characteristics of generalized fuzzy trees, highlighting their dynamic and weighted aspects in representing hierarchical relationships. Mandal et al. [19] developed a decision-making model for advanced ovarian cancer treatment, leveraging social network trust relationships and incorporating dynamic, weighted graphs to reflect varying levels of expert confidence. Samanta et al. [27] analyzed semi-directed graphs within social media networks, focusing on their dynamic and weighted nature to better understand the flow and influence of information. But these graphs can not represent the dynamic nature of nodes and links in networks directly.

Generally, stability analysis is not done in fuzzy graphs due to their static nature. Still few studies are available for stability analysis by using graph theory. Hosseinzadeh Lotfi et al. [15] introduced a two-stage network based on ratio analysis models, emphasizing the dynamic nature of efficiency evaluation in data envelopment analysis. Allahviranloo et al. [2] applied artificial neural networks to solve fractional higher-order linear integro-differential equations, highlighting the stability analysis of these dynamic systems. Ghasemi et al. [13] developed an intelligent model using a fuzzy multilayer neural network to enhance credit risk management, focusing on the dynamic and uncertain conditions of financial environments. Eidinejad et al. [11] examined the attractiveness of pseudo almost periodic solutions for delayed cellular neural networks, utilizing fuzzy measure theory to analyze the stability of these dynamic systems in matrix-valued fuzzy spaces. Adeniji et al. [1] focused on a mathematical model analyzing migration and unemployment among PhD graduates and postdocs in South Africa, using stability analysis to understand the dynamics between academia and industry. Olutimo et al. [22] investigated the stability and boundedness of solutions for nonlinear delay differential equations, employing Lyapunov's methods to establish conditions for stability. Murali et al. [21] worked on stability analysis using differential equations for dynamical systems. Jayapriya et al. [16] explored the application of a transformation to establish the stability for nth order linear differential equations. These studies collectively highlight the importance of stability analysis in understanding and predicting the behavior of complex systems in various fields.

The study by Ghaderi et al. [12] and Ghosh et al.[14] applied synchronization in various real-world systems, emphasizing the importance of understanding these dynamics for applications in physics, biology, and social sciences. The comparison of a few related works is shown in Table 2.

3. Quantum graphs

Quantum graphs are a generalised version of graph theory with dynamic vertices and entangled edges, defined as follows:
A graph $G = (V, E)$, where V is the set of vertices and E is the set of edges, is said to be a quantum graph if each vertex v_i is assigned a dynamic weight $\text{dweight}_i(t)$ that changes over times. The dweight is also assumed as:

$$0 \leq \text{dweight}_i(t) \leq 1.$$

The dweight is represented by any function of time (see examples in the subsection 3.1) such as:

$$\text{dweight}_i(t) = \frac{1}{2}\left(1 + \sin(\omega_i t + \phi_i)\right),$$

where ω_i is the frequency and ϕ_i is the phase shift. Edges e_{ij} between vertices v_i and v_j are entangled or not. If entangled, the edge is represented by a correlation function $g_{ij}(t)$ that describes the relationship between the dweights of the connected vertices:

$$g_{ij}(t) = h(\text{dweight}_i(t), \text{dweight}_j(t)),$$

TABLE 2. Comparison of previous works with proposed work.

Authors	**Dynamic Vertices**	**Entangled Edges**	**Stability Analysis**	**Synchronization Synchronization**	**Quantum Communication**	**Quantum Computing**
Kottos and Smilansky (1997)[17]	×	×	×	×	×	✓
Razavinia and Haghighatdoost (2024)[24]	×	✓	×	×	×	✓
Tang, Yan, and Hancock (2022)[31]	×	×	×	×	×	✓
Mernyei, Meichanetzidis and Ceylan (2022)[20]	✓	×	×	×	×	✓
Proposed Work	✓	✓	✓	✓	✓	✓

where h is a function defining the nature of the entanglement or correlation.

Remark 3.1. Dynamic Nature (Superposition Behavior) The dynamic nature of the vertices means that the weights $\text{dweight}_i(t)$ change over time.

Remark 3.2. The phase shift (ϕ) in the dynamic weight function of a node affects the timing of the oscillations of the node's weight. The dynamic weight function is typically given by:

$$\text{dweight}_i(t) = \frac{1}{2}\left(1 + \sin(\omega_i t + \phi_i)\right).$$

The phase shift determines the initial position of the sine wave at (t = 0).

Example 3.3. Let us consider the range of ω to be $[0, 2\pi]$, $\phi = \frac{\pi}{4}$, and t to range from 0 to 10 with a step size of 0.01. The number of vertices is 5 and the number of edges is 8. Let us denote the vertices as v_1, v_2, v_3, v_4, v_5, and the edges as $e_{12}, e_{13}, e_{14}, e_{15}$, $e_{23}, e_{24}, e_{25}, e_{34}$.

The dynamic weight functions for the vertices are:

$$dweight_1(t) = 0.5 \times (1 + \sin(\omega_1 \times t + \phi)),$$
$$dweight_2(t) = 0.5 \times (1 + \sin(\omega_2 \times t + \phi)),$$
$$dweight_3(t) = 0.5 \times (1 + \sin(\omega_3 \times t + \phi)),$$
$$dweight_4(t) = 0.5 \times (1 + \sin(\omega_4 \times t + \phi)),$$
$$dweight_5(t) = 0.5 \times (1 + \sin(\omega_5 \times t + \phi)).$$

The correlation functions for the edges are:

$$g_{12}(t) = h(dweight_1(t), dweight_2(t)),$$
$$g_{13}(t) = h(dweight_1(t), dweight_3(t)),$$
$$g_{14}(t) = h(dweight_1(t), dweight_4(t)),$$
$$g_{15}(t) = h(dweight_1(t), dweight_5(t)),$$
$$g_{23}(t) = h(dweight_2(t), dweight_3(t)),$$
$$g_{24}(t) = h(dweight_2(t), dweight_4(t)),$$
$$g_{25}(t) = h(dweight_2(t), dweight_5(t)),$$
$$g_{34}(t) = h(dweight_3(t), dweight_4(t)).$$

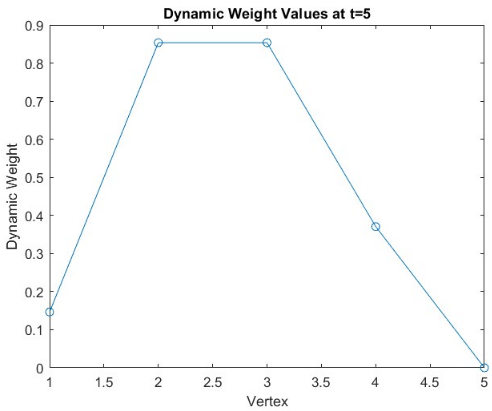

FIGURE 1. The dweights of nodes of quantum graph.

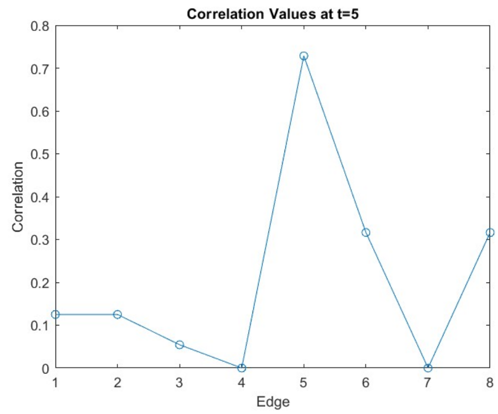

FIGURE 2. Correlation values of edges of quantum graph.

The adjacency matrix $A(t)$ of the graph is defined such that:

$$A(t) = \begin{bmatrix} 0 & g_{12}(t) & g_{13}(t) & g_{14}(t) & g_{15}(t) \\ g_{12}(t) & 0 & g_{23}(t) & g_{24}(t) & g_{25}(t) \\ g_{13}(t) & g_{23}(t) & 0 & g_{34}(t) & 0 \\ g_{14}(t) & g_{24}(t) & g_{34}(t) & 0 & 0 \\ g_{15}(t) & g_{25}(t) & 0 & 0 & 0 \end{bmatrix}.$$

This matrix represents the weights of the edges at time t.

Now, let us assume, the dynamic weight values and correlation values for $h(x, y) = x \times y$ with $\omega_1 = \pi$, $\omega_2 = \pi/2$, $\omega_3 = 2\pi$, $\omega_4 = \pi/3$, $\omega_5 = \pi/4$, $\phi = \pi/4$, and $t = 5$.

The dynamic weight values at $t = 5$ are:

$$dweight_1(5) = 0.15, dweight_2(5) = 0.85, \quad dweight_3(5) = 0.85, dweight_4(5) = 0.37, \quad dweight_5(5) = 0.0.$$

The correlation values at $t = 5$ are:

$$g_{12}(5) = 0.12, \quad g_{13}(5) = 0.12, \quad g_{14}(5) = 0.05, \quad g_{15}(5) = 0.0,$$
$$g_{23}(5) = 0.72, \quad g_{24}(5) = 0.32, \quad g_{25}(5) = 0.0, \quad g_{34}(5) = 0.32.$$

These values represent the weights of the vertices and edges at time $t = 5$. If there is no edge between two vertices, the corresponding entry in the matrix is 0. The dweights of nodes are shown in Figure 1. In Figure 2, the correlation values of edges are shown and the comparison of dweights of nodes and correlation values of edges over time have been shown in Figure 3.

3.1. **Few weight functions of nodes and correlations.** The choice of the weight function depends on the specific properties and behaviors we want to model in quantum graphs. A few examples of weight functions are given as follows.

1. Cosine Function:

The cosine function can be adjusted similarly to the sine function:

$$\text{dweight}_i(t) = \frac{1}{2}\left(1 + \cos(\omega_i t + \phi_i)\right).$$

2. Exponential Decay:

For weights that decrease over time, we can use an exponential decay function.

$$\text{dweight}_i(t) = e^{-\lambda_i t},$$

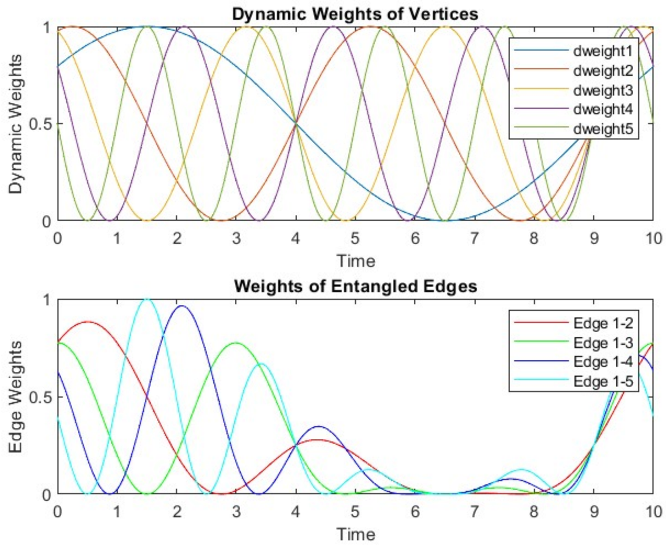

FIGURE 3. Comparison of dweights of nodes and correlation values of edges over time.

where λ_i is chosen such that $0 \leq e^{-\lambda_i t} \leq 1$.

3. Gaussian Function:

For weights that peak at a certain time and then decrease, we can use a Gaussian function.

$$\text{dweight}_i(t) = e^{-\frac{(t-\mu_i)^2}{2\sigma_i^2}},$$

where the parameters μ_i and σ_i are chosen in such a way as to fit the values between 0 and 1.

4. Linear Function:

For weights that change linearly over time, we can use a linear function. A linear function can be scaled and shifted to fit within the range.

$$\text{dweight}_i(t) = \frac{a_i t + b_i}{c_i},$$

where a_i, b_i, and c_i are chosen such that $0 \leq \frac{a_i t + b_i}{c_i} \leq 1$.

5. Periodic Functions:

Few periodic functions other than sine function are given as follows

$$\text{dweight}_i(t) = \frac{1}{2}\left(1 + \text{triangular}(\omega_i t + \phi_i)\right),$$

or

$$\text{dweight}_i(t) = \frac{1}{2}\left(1 + \text{square}(\omega_i t + \phi_i)\right).$$

6. Polynomial Functions:

For more complex behaviors, polynomial functions can be used. Polynomial functions can be normalized:

$$\text{dweight}_i(t) = \frac{a_i t^2 + b_i t + c_i}{d_i},$$

where a_i, b_i, c_i, and d_i are chosen to fit the values between 0 and 1.

7. Custom Functions:

It can also be defined custom functions based on the specific requirements of the model. For example, a combination of different functions: Custom functions can be designed to fit within the range by combining and normalizing different functions.

$$\text{dweight}_i(t) = \frac{1}{2}\left(1 + \sin(\omega_i t + \phi_i)\right) + \frac{e^{-\lambda_i t}}{2}.$$

This ensures that the dynamic weights stay within the desired range of 0 to 1.

3.1.1. *Correlations among nodes.* Some examples of correlation functions h that are used to define the entanglement or correlation between the dynamic weights of vertices in quantum graphs are as follows:

1. Linear Correlation:

$$h(\text{dweight}_i(t), \text{dweight}_j(t)) = \text{dweight}_i(t) \cdot \text{dweight}_j(t).$$

2. Cosine Similarity:

$$h(\text{dweight}_i(t), \text{dweight}_j(t)) = \cos(\pi \cdot (\text{dweight}_i(t) - \text{dweight}_j(t))).$$

3. Exponential Decay:

$$h(\text{dweight}_i(t), \text{dweight}_j(t)) = e^{-\left|\text{dweight}_i(t) - \text{dweight}_j(t)\right|}.$$

4. Gaussian Correlation:

$$h(\text{dweight}_i(t), \text{dweight}_j(t)) = \exp\left(-\frac{(\text{dweight}_i(t) - \text{dweight}_j(t))^2}{2\sigma^2}\right).$$

where σ is a parameter that controls the width of the Gaussian function.

5. Sigmoid Function:

$$h(\text{dweight}_i(t), \text{dweight}_j(t)) = \frac{1}{1 + \exp(-k(\text{dweight}_i(t) - \text{dweight}_j(t)))}.$$

where k is a parameter that controls the steepness of the sigmoid function.

These functions h provide different ways to model the relationship between the dynamic weights of the vertices, capturing various types of correlations or entanglements.

3.2. **Degree of nodes.** The degree of a node v_i in a quantum graph is defined as the sum of all edge weights adjacent to that node. Mathematically, if $g_{ij}(t)$ represents the weight of the edge between nodes v_i and v_j, the degree $d_i(t)$ of node v_i at time t is given by:

$$d_i(t) = \sum_j g_{ij}(t).$$

For simplicity, let's assume h is a product function:

$$g_{ij}(t) = \text{dweight}_i(t) \cdot \text{dweight}_j(t).$$

The degree $d_i(t)$ of node v_i is the sum of all $g_{ij}(t)$ values. If there are n nodes in the graph, the minimum degree is 0 (if all $g_{ij}(t)$ are 0) and the maximum degree is $n-1$ (if all $g_{ij}(t)$ are 1).

Using the sinusoidal function for $\text{dweight}_i(t)$:

$$\text{dweight}_i(t) = \frac{1}{2}\left(1 + \sin(\omega_i t + \phi_i)\right).$$

The correlation function $g_{ij}(t)$ becomes:

$$g_{ij}(t) = \left(\frac{1}{2}\left(1 + \sin(\omega_i t + \phi_i)\right)\right) \cdot \left(\frac{1}{2}\left(1 + \sin(\omega_j t + \phi_j)\right)\right).$$

Simplifying, we get:

$$g_{ij}(t) = \frac{1}{4}\left(1 + \sin(\omega_i t + \phi_i)\right)\left(1 + \sin(\omega_j t + \phi_j)\right).$$

TABLE 3. Comparison between crisp graphs and quantum graphs.

Aspect	Crisp Graphs	Quantum Graphs
Definition	Static graphs with fixed edges and weights	Dynamic graphs with time-varying weights
Structure	Fixed structure	Dynamic structure
Mathematical Representation	Adjacency matrix or list	Dynamic matrices with time-dependent entries
Applications	Network analysis, shortest path algorithms, social network analysis	Quantum computing, dynamic network analysis
Example	Road network with fixed distances	Quantum communication network with changing connection strengths

Thus, the degree $d_i(t)$ in terms of the sinusoidal functions is:

$$d_i(t) = \sum_j \frac{1}{4} \left(1 + \sin(\omega_i t + \phi_i)\right) \left(1 + \sin(\omega_j t + \phi_j)\right).$$

3.3. **Degree of a node with sigmoid correlation.** The degree of a node v_i is simplified with a sigmoid correlation functions:

$$h(\text{dweight}_i(t), \text{dweight}_j(t)) = \frac{1}{1 + \exp(-k(\text{dweight}_i(t) - \text{dweight}_j(t)))}.$$

Given the dynamic weight function:

$$\text{dweight}_i(t) = \frac{1}{2} \left(1 + \sin(\omega_i t + \phi_i)\right).$$

The correlation function $g_{ij}(t)$ becomes:

$$g_{ij}(t) = \frac{1}{1 + \exp\left(-k\left(\frac{1}{2}\left(1 + \sin(\omega_i t + \phi_i)\right) - \frac{1}{2}\left(1 + \sin(\omega_j t + \phi_j)\right)\right)\right)}.$$

Simplifying the argument of the exponential function:

$$g_{ij}(t) = \frac{1}{1 + \exp\left(-k\left(\frac{1}{2}\left(\sin(\omega_i t + \phi_i) - \sin(\omega_j t + \phi_j)\right)\right)\right)},$$

$$g_{ij}(t) = \frac{1}{1 + \exp\left(-\frac{k}{2}\left(\sin(\omega_i t + \phi_i) - \sin(\omega_j t + \phi_j)\right)\right)}.$$

Now, the degree $d_i(t)$ of node v_i is the sum of all $g_{ij}(t)$ values:

$$d_i(t) = \sum_j g_{ij}(t).$$

Substituting the simplified correlation function:

$$d_i(t) = \sum_j \frac{1}{1 + \exp\left(-\frac{k}{2}\left(\sin(\omega_i t + \phi_i) - \sin(\omega_j t + \phi_j)\right)\right)}.$$

The degree $d_i(t)$ will vary over time depending on the values of ω_i, ϕ_i, ω_j, ϕ_j, and the parameter k.

3.4. **Comparison with weighted graphs.** There is a significant difference between weighted graphs and quantum graphs structurally and in applications. In weighted graphs, fixed values are assigned to vertices and edges as weights such as cost or distance, making them suitable for applications in network routing, shortest path algorithms, resource allocation, etc. Conversely, quantum graphs feature dynamic vertices with weights that change over time, often following patterns like sinusoidal functions, and edges that can be entangled, meaning their relationships are described by time-dependent correlation functions. This dynamic and complex nature of quantum graphs makes them applicable in advanced fields like quantum computing, where they can model quantum states and entanglement, and in dynamic systems analysis, where changing relationships over time are crucial (see Figures 4 and 5 and Table 3).

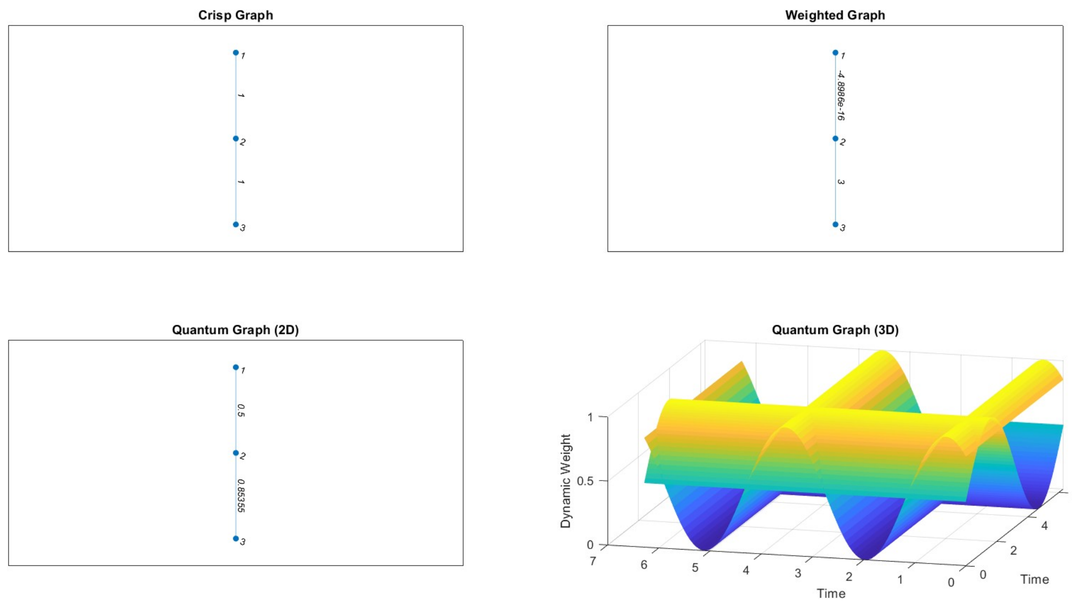

FIGURE 4. Comparison of graphs with quantum graphs.

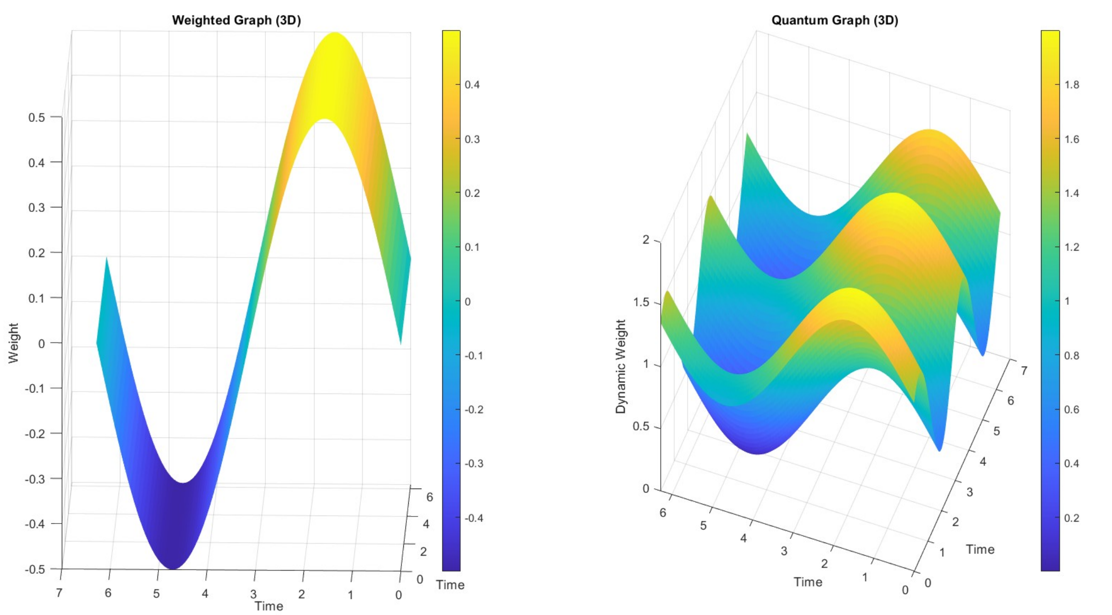

FIGURE 5. Comparison of weighted graphs with quantum graphs.

4. Stability of quantum graphs

In this section, we will perform a stability analysis of the quantum graph with dynamic vertices and potentially entangled edges. The goal is to determine the conditions under which the system remains stable, meaning that small perturbations in the system's state will decay over time, leading the system back to its equilibrium state.

4.1. **Linear stability analysis.** Linear stability analysis involves examining the behavior of small perturbations around an equilibrium point. For our quantum graph, the equilibrium state is defined by the dynamic weights of the vertices at a specific time t_0.

Let $\mathbf{x}(t) = [\text{dweight}_1(t), \text{dweight}_2(t), \ldots, \text{dweight}_n(t)]$ be the state vector representing the dynamic weights of the vertices. The time evolution of the state vector is governed by a set of differential equations:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{f}(\mathbf{x}(t), t),$$

where $\mathbf{f}$ is a vector function that describes the time evolution of the state variables.

To perform linear stability analysis, we linearize the system around the equilibrium point $\mathbf{x}_0$. The linearized system can be written as:

$$\frac{d\mathbf{y}(t)}{dt} = J(\mathbf{x}_0, t)\mathbf{y}(t),$$

where $\mathbf{y}(t) = \mathbf{x}(t) - \mathbf{x}_0$ is the perturbation vector, and $J(\mathbf{x}_0, t)$ is the Jacobian matrix of the system evaluated at the equilibrium point $\mathbf{x}_0$:

$$J(\mathbf{x}_0, t) = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_0}.$$

The stability of the system depends on the eigenvalues of the Jacobian matrix $J(\mathbf{x}_0, t)$. If all eigenvalues have negative real parts, the perturbations will decay over time, and the system is stable.

4.2. **Eigenvalue analysis.** To analyze the stability of the system, we need to compute the eigenvalues of the Jacobian matrix $J(\mathbf{x}_0, t)$. The eigenvalues λ_i are solutions to the characteristic equation:

$$\det(J(\mathbf{x}_0, t) - \lambda I) = 0,$$

where I is the identity matrix.

For our quantum graph, the Jacobian matrix $J(\mathbf{x}_0, t)$ can be derived from the adjacency matrix $A(t)$ and the dynamic weights of the vertices. The entries of the Jacobian matrix are given by:

$$J_{ij}(\mathbf{x}_0, t) = \left.\frac{\partial f_i}{\partial x_j}\right|_{\mathbf{x}=\mathbf{x}_0},$$

where f_i is the i-th component of the vector function $\mathbf{f}$.

5. A small case study

Four vertices with different frequencies and phases

(1) Graph Structure: Vertices: v_1, v_2, v_3, v_4, Edges: $e_{12}, e_{23}, e_{34}, e_{41}$ (all entangled).
(2) State Variables: $\mathbf{x}(t) = [\text{dweight}_1(t), \text{dweight}_2(t), \text{dweight}_3(t), \text{dweight}_4(t)]$.
(3) Time Evolution: $\text{dweight}_i(t) = \frac{1}{2}(1 + \sin(\omega_i t + \phi_i))$ for $i = 1, 2, 3, 4$.
(4) Interaction: $g_{ij}(t) = h(\text{dweight}_i(t), \text{dweight}_j(t))$ for $(i, j) \in \{(1, 2), (2, 3), (3, 4), (4, 1)\}$.
(5) Adjacency Matrix: $A(t) = \begin{pmatrix} 0 & g_{12}(t) & 0 & g_{41}(t) \\ g_{12}(t) & 0 & g_{23}(t) & 0 \\ 0 & g_{23}(t) & 0 & g_{34}(t) \\ g_{41}(t) & 0 & g_{34}(t) & 0 \end{pmatrix}.$

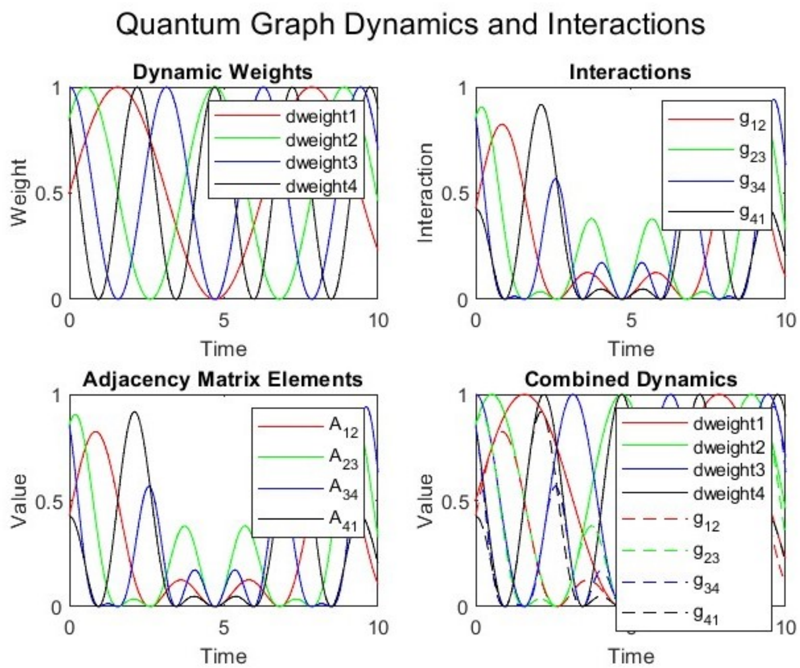

FIGURE 6. Stability analysis of quantum graph.

5.1. Analysis for $t \in [0, 10]$.

(1) Stability: Linear stability analysis by examining the eigenvalues of $A(t)$ over the interval $t \in [0, 10]$ can be performed. Lyapunov functions can be used to further analyze stability.
(2) Synchronization: The dynamic weights synchronize over time and can be checked. This involves analyzing the phase differences and the interaction functions h.

5.2. Results.

(1) Stability: The system's stability depends on the eigenvalues of the adjacency matrix and the nature of the interaction functions. Over the interval $t \in [0, 10]$, observe how the eigenvalues change and determine if the system remains stable.
(2) Synchronization: Synchronization is more complex in this case. If all ω_i and ϕ_i are equal, perfect synchronization can occur. Otherwise, the system may exhibit partial synchronization or complex synchronization patterns.

The dynamics of a quantum graph with four vertices, each having different frequencies and phases, is shown in Figure 6. The dynamic weights of the vertices are defined as sinusoidal functions of time, representing oscillatory behavior. The interaction between vertices is modeled as the product of their dynamic weights. The code calculates these interactions over a specified time interval and plots the dynamic weights, interactions, and elements of the adjacency matrix over time. The results show how the weights and interactions evolve, providing insights into the system's behavior. The combined dynamics plot illustrates the interplay between individual vertex dynamics and their interactions, highlighting potential synchronization or complex patterns in the system.

The provided Figure 7 demonstrates phase synchronization by plotting the dynamic weights of four vertices over time, each with different frequencies and phase shifts. The dynamic weights are calculated using sinusoidal functions, which inherently oscillate. By plotting these weights on the same graph, we can visually inspect whether the oscillations of the vertices align or maintain a consistent phase difference. If the vertices are phase-synchronized, their dynamic

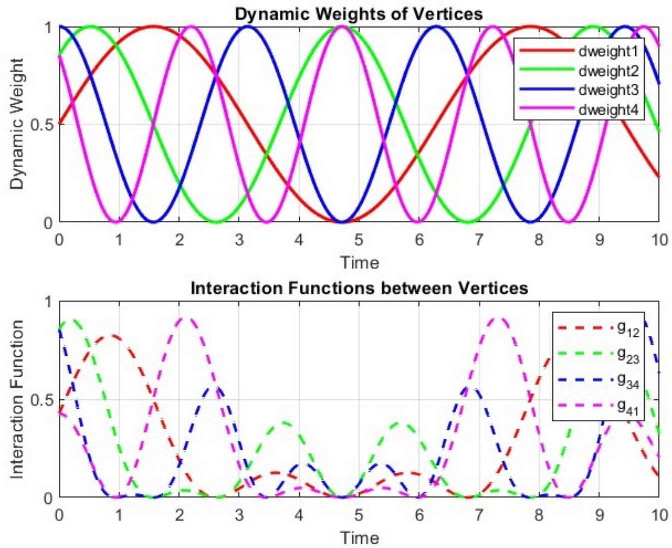

FIGURE 7. Synchronisation of phases in quantum graph.

weights will show a consistent pattern of oscillation relative to each other, either oscillating in unison or maintaining a constant phase difference. The interaction functions, plotted in the second subplot, further illustrate the relationships between the vertices. These functions are products of the dynamic weights of connected vertices, and their consistent patterns over time indicate synchronized interactions. Together, these plots provide a clear visual representation of how the phases of the vertices' oscillations are synchronized within the quantum graph.

6. Quantum network

Quantum graphs can be applied to quantum networks where nodes are connected by quantum channels.

6.1. **Key components of quantum networks.**

(1) Nodes of quantum networks are the qubits or qudits or any kind of quantum processors. Mathematically, V in a quantum graph $G = (V, E)$ is the set of such nodes. Examples are trapped ions, superconducting qubits, and photonic qubits. Nodes (processors) are connected by quantum channels.

(2) Entanglement: Each node v_i has a dynamic weight $\text{dweight}_i(t)$ that changes over time. The *dweights* represent the quantum state or the strength of its quantum processing capability. Correlation Function $g_{ij}(t)$: The edge weights $g_{ij}(t)$, the correlations represent the strength of the quantum channel between nodes v_i and v_j. Following is an example:

$$g_{ij}(t) = \frac{1}{4}\left(1 + \sin(\omega_i t + \phi_i)\right)\left(1 + \sin(\omega_j t + \phi_j)\right).$$

Adjacency Matrix $A(t)$: The adjacency matrix $A(t)$, where each element $A_{ij}(t)$ corresponds to the weight of the edge between nodes v_i and v_j is represented as follows:

$$A_{ij}(t) = \begin{cases} g_{ij}(t), & \text{if } e_{ij} \text{ exists,} \\ 0, & \text{otherwise.} \end{cases}$$

6.2. **Example: Quantum key distribution (QKD) network.** Quantum Key Distribution (QKD) is a secure communication method that uses quantum mechanics principles to enable two parties to generate a shared, random secret key, which can then be used to encrypt and decrypt messages. The security of QKD is based on the fundamental properties of quantum mechanics, such as the no-cloning theorem and the principle of measurement disturbance.

Let us consider a simple quantum network for QKD with three nodes v_1, v_2, and v_3.

(1) Nodes:
v_1: Ram's quantum processor. v_2: Ranjan's quantum processor. v_3: Romario's quantum processor.

(2) Edges:
e_{12}: Quantum channel between Ram and Ranjan. e_{13}: Quantum channel between Ram and Romario. e_{23}: Quantum channel between Ranjan and Romario.

(3) Dynamic Weights:
$\text{dweight}_1(t) = \frac{1}{2}\left(1 + \sin(\omega_1 t + \phi_1)\right),$
$\text{dweight}_2(t) = \frac{1}{2}\left(1 + \sin(\omega_2 t + \phi_2)\right),$
$\text{dweight}_3(t) = \frac{1}{2}\left(1 + \sin(\omega_3 t + \phi_3)\right).$

(4) Correlation Functions:
$g_{12}(t) = \frac{1}{4}\left(1 + \sin(\omega_1 t + \phi_1)\right)\left(1 + \sin(\omega_2 t + \phi_2)\right),$
$g_{13}(t) = \frac{1}{4}\left(1 + \sin(\omega_1 t + \phi_1)\right)\left(1 + \sin(\omega_3 t + \phi_3)\right),$
$g_{23}(t) = \frac{1}{4}\left(1 + \sin(\omega_2 t + \phi_2)\right)\left(1 + \sin(\omega_3 t + \phi_3)\right).$

(5) Adjacency Matrix:
The adjacency matrix $A(t)$ for this network is:

$$A(t) = \begin{pmatrix} 0 & g_{12}(t) & g_{13}(t) \\ g_{12}(t) & 0 & g_{23}(t) \\ g_{13}(t) & g_{23}(t) & 0 \end{pmatrix}.$$

7. Conclusion

In this study, the patterns of dynamic weights and correlation functions are analysed and exemplified. But there may arise a question that there may exist many systems where vertices do not follow any patterns over times. And edges also do not follow any correlations. In those cases, the functions which fit the vertex values most may be assumed. Suppose any vertex provides random values over times, the values may not follow any pattern, but the best fitted curve may be assumed to represent the dweight functions. In that way, any dynamic system can be represented by quantum graphs. In the future, dynamic systems such as brain networks and diseases networks will be represented by quantum graphs. And hence, the representation and early detection of diseases may be possible. These are referred to as the future scope of the study.

Acknowledgments:

The authors are thankful to honourable Editor-in-Chief, honourable Associate Editors, and honourable Reviewers of the journal for their comments to improve the article. Very few parts of the article were checked using AI (Copilot) for English correction only.

References

[1] A. A. Adeniji, M. O. Adeniyi, and M. Y. Shatalov, *Mathematical model and stability analysis of university-PhD-postdoc-industry migration and unemployment in South Africa*, J. Math. Comput. Sci., *31*(1) (2023), 81-94.

[2] T. Allahviranloo, A. Jafarian, R. Saneifard, N. Ghalami, S. Measoomy Nia, F. Kiani, and S. Noeiaghdam, *An application of artificial neural networks for solving fractional higher-order linear integro-differential equations*, Boundary Value Problems, *2023*(1) (2023), 74.

[3] T. Allahviranloo and S. Samanta, *Quantum Theory and Fuzzy Systems: Traversing Uncertainty in Group Decision-Making and Social Networks*, e-ISBN: 978-3-031-82058-8, Studies in Computational Intelligence, Springer, 2025, e-ISSN:1860-9503.

[4] D. S. Bassett and E. D. Bullmore, *Small-world brain networks*, The Neuroscientist, *12*(6) (2006), 512-523.

[5] C. H. Bennett and G. Brassard, *An update on quantum cryptography*, In Workshop on the theory and application of cryptographic techniques, Berlin, Heidelberg: Springer Berlin Heidelberg, (1984, August), 475-480.

[6] M. Brang, H. Franke, F. Greinert, M. S. Ubben, F. Hennig, and P. Bitzenbauer, *Spooky action at a distance? A two-phase study into learners' views of quantum entanglement*, EPJ Quantum Technology, *11*(1) (2024), 33.

[7] E. Bullmore and O. Sporns, *Complex brain networks: graph theoretical analysis of structural and functional systems*, Nature Reviews Neuroscience, *10*(3) (2009), 186-198.

[8] S. L. Cornish, M. R. Tarbutt, and K. R. Hazzard, *Quantum computation and quantum simulation with ultracold molecules*, Nature Physics, (2024), 1-11.

[9] K. Das, A. Maity, K. De, S. Mondal, S. Samanta, and T. Allahviranloo, *Link Prediction in Co-Authorship Network Under Fuzziness and Application in Biomedical Analysis*, Fuzzy Information and Engineering, *16*(2) (2024), 155-174.

[10] J. F. Dynes, A. Wonfor, W. S. Tam, A. W. Sharpe, R. Takahashi, M. Lucamarini, and et al., *Cambridge quantum network*, npj Quantum Information, *5*(1) (2019), 101.

[11] Z. Eidinejad, R. Saadati, and T. Allahviranloo, *Attractiveness of pseudo almost periodic solutions for delayed cellular neural networks in the context of fuzzy measure theory in matrix-valued fuzzy spaces*, Computational and Applied Mathematics, *41*(8) (2022), 369.

[12] A. H. Ghaderi, B. R. Baltaretu, M. N. Andevari, V. Bharmauria, and F. Balci, *Synchrony and complexity in state-related EEG networks: an application of spectral graph theory*, Neural Computation, *32*(12) (2020), 2422-2454.

[13] Z. Ghasemi, M. Afshar Kermani, and T. Allahviranloo, *Developing an Intelligent Model Based on Fuzzy Multilayer Neural Network for Improving Credit Risk Management under Uncertain Conditions*, Mathematical Problems in Engineering, *2023*(1) (2023), 4141140.

[14] D. Ghosh, M. Frasca, A. Rizzo, S. Majhi, S. Rakshit, K. Alfaro-Bittner, and S. Boccaletti, *The synchronized dynamics of time-varying networks*, Physics Reports, *949* (2022), 1-63.

[15] F. Hosseinzadeh Lotfi, T. Allahviranloo, W. Pedrycz, M. R. Mozaffari, and J. Gerami, *The Two-Stage Network Based on Ratio Analysis Models*, In Comparative Efficiency in Data Envelopment Analysis Based on Ratio Analysis, Cham: Springer International Publishing, (2023), 139-155.

[16] M. Jayapriya, A. Ganesh, S. S. Santra, R. Edwan, D. Baleanu, and K. M. Khedher, *Sawi transform and Hyers-Ulam stability of nth order linear differential equations*, *28*(4) (2022), 393–411.

[17] T. Kottos and U. Smilansky, *Quantum chaos on graphs*, Physical Review Letters, *79*(24) (1997), 4794.

[18] S. Lloyd, *Universal quantum simulators*, Science, *273*(5278) (1996), 1073-1078.

[19] P. Mandal, S. Samanta, M. Pal, and A. S. C. Ranadive, *Social network trust relationship environment based advanced ovarian cancer treatment decision-making model: An approach based on linguistic information with experts' multiple confidence levels*, Expert Systems with Applications, *229* (2023), 120407.

[20] A. Skolik, M. Cattelan, S. Yarkoni, T. Bäck, and V. Dunjko, *Equivariant quantum circuits for learning on weighted graphs*, npj Quantum Information, *9*(1) (2023), 47.

[21] R. Murali, A. P. Selvan, and S. Baskaran, *Stability of linear differential equation of higher order using Mahgoub transforms*, J. Math. Comput. Sci., *30* (2023), 1-9.

[22] A. L. Olutimo, A. Bilesanmi, and I. D. Omoko, *Stability and boundedness analysis for a system of two nonlinear delay differential equations*, J. Nonlinear Sci. Appl., *16*(2) (2023), 90-98.

[23] S. Perseguers, M. Lewenstein, A. Acín, and J. I. Cirac, *Quantum random networks*, Nature Physics, *6*(7) (2010), 539-543.

[24] F. Razavinia and G. Haghighatdoost, *A route to quantum computing through the theory of quantum graphs*, arXiv preprint arXiv:2404.13773.

[25] S. Samanta and B. Sarkar, *A study on generalized fuzzy graphs*, Journal of Intelligent and Fuzzy Systems, *35*(3) (2018), 3405-3412.

[26] S. Samanta, G. Muhiuddin, A. M. Alanazi, and K. Das, *A mathematical approach on representation of competitions: competition cluster hypergraphs*, Mathematical Problems in Engineering, *2020*(1) (2020), 2517415.

[27] S. Samanta, M. Pal, R. Mahapatra, K. Das, and R. S. Bhadoria, *A study on semi-directed graphs for social media networks*, International Journal of Computational Intelligence Systems, *14*(1) (2021), 1034-1041.
[28] B. Sarkar and S. Samanta, *Generalized fuzzy trees*, International Journal of Computational Intelligence Systems, *10*(1) (2017), 711-420.
[29] E. Schrödinger, *The present situation in quantum mechanics: A translation of Schrödinger's "Cat Paradox" paper*, Quantum theory and measurement, Princeton University Press, Princeton, New Jersey, (1983), 152-167.
[30] G. D. Scholes, *Quantum-like states on complex synchronized networks*, Proceedings of the Royal Society A, *480*(2295) (2024), 20240209.
[31] Y. Tang, J. Yan, and H. Edwin, *From quantum graph computing to quantum graph learning: A survey*, arXiv preprint arXiv:2202.09506, (2022).