Multivariate Time Series Forecasting With GARCH Models on Graphs

Junping Hong, Yi Yan, *Graduate Student Member, IEEE*, Ercan Engin Kuruoglu, *Senior Member, IEEE*, and Wai Kin (Victor) Chan, *Senior Member, IEEE*

***Abstract*—Data that house topological information is manifested as relationships between multiple variables via a graph formulation. Various methods have been developed for analyzing time series on the nodes of graphs but research works on graph signals with volatility are limited. In this article, we propose a graph framework of multivariate Generalized Autoregressive Conditional Heteroscedasticity (GARCH) models from the spectral perspective with the Laplacian matrix. We introduce three graphical GARCH models: one symmetric Graph GARCH model and two asymmetric models namely Graph Exponential GARCH and Graph GJR-GARCH. Assuming that graph signals and their residuals are *graph stationary*, this framework can decompose the multivariate GARCH models into a linear combination of several univariate GARCH processes in the graph spectral domain. Moreover, it is possible to reduce the number of parameters with the graph topology information and further reduce the estimation cost by utilizing the principal components of the graph signal in the frequency domain. These proposed models are tested on synthetic data and on two real applications for weather prediction and wind power forecasting. With the data and GARCH model residuals being graph stationary, the experiment results demonstrate that these three graphical models can make multi-step predictions more accurately than non-graph GARCH models and Graph Vector Autoregressive Moving Average model.**

***Index Terms*—Graph signal processing, laplacian matrix, multivariate time series forecasting, ARCH, GARCH, multivariate GARCH, EGARCH, GJR-GARCH, O-GARCH.**

Nomenclature

ARCH	Autoregressive Conditional Heteroskedasticity.
ARMA	Autoregressive Moving Average.
CCC	Constant Correlation Covariance.
EGARCH	Exponential Generalized Autoregressive Conditional Heteroskedasticity.
GARCH	Generalized Autoregressive Conditional Heteroskedasticity.
GJR-GARCH	Glosten-Jagannathan-Runkle GARCH.
GO-GARCH	Generalized Orthogonal GARCH.
G-VARMA	Graph Vector Autoregressive Moving Average.
O-GARCH	Orthogonal GARCH.
VARMA	Vector Autoregressive Moving Average.

Manuscript received 25 August 2022; revised 19 February 2023 and 15 May 2023; accepted 31 July 2023. Date of publication 10 August 2023; date of current version 25 August 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 71971127, in part by Shenzhen Science and Technology Innovation Commission under Grant JCYJ20210324135011030, in part by Guangdong Pearl River Plan under Grant 2019QN01X890, in part by High-end Foreign Expert Talent Introduction Plan under Grants G2021032021L and G2021032022L, and in part by Shenzhen International Graduate School, Tsinghua University under Grant JC2021004. The associate editor coordinating the review of this manuscript and approving it for publication was Miss Stefania Sardellitti. *(Corresponding authors: Wai Kin (Victor) Chan; Ercan Engin Kuruoglu.)*

The authors are with the Tsinghua-Berkeley Shenzhen Institute, Tsinghua Shenzhen International Graduate School, Shenzhen 518055, China (e-mail: jacob.hong17@gmail.com; y-yan20@mails.tsinghua.edu.cn; kuruoglu@sz.tsinghua.edu.cn; chanw@sz.tsinghua.edu.cn).

Digital Object Identifier 10.1109/TSIPN.2023.3304142

I. Introduction

NOWADAYS, Big Data structured as networks are ubiquitous in our daily lives, and they play an important role in modern society with continually increasing influence. These data include social network data, power grid data, sensor network data, etc. Volatility exists in various real networks and is often the cause of many physical and human/social factors, for example, fluctuations in energy demands in power grids [1], and vacillations in wind speed [2]. Recently, researchers have extended the traditional signal processing methods to the graph domain, leading to graph signal processing (GSP), which utilizes algebraic graph theory, spectral graph theory, linear algebra, and signal processing techniques [3]. The emergence of graph domain signal processing tools, such as graph shift operators, graph filters, etc., have appeared along with the introduction of GSP [4].

In time series analysis, the central topic is to forecast the future value from current and past observations. Such forecasting can provide consultation for economic and business planning and industrial process control, such as in energy plants. Well-known multivariate time series models are Vector Autoregressive (VAR) processes [5] and Vector Autoregressive Moving Average (VARMA) processes [5]. Despite their popularity in various applications, these models have the limitation of having a constant covariance, which cannot account for time-varying features, such as volatility. Volatility has been studied extensively in finance by various researchers. It has been observed in the study of asset return and risk that the inherent randomness is time-varying, small and large fluctuations tend to cluster together [6]. In the study of univariate data, Autoregressive Conditional Heteroscedasticity (ARCH) [6], and generalized ARCH (GARCH) [7] have been proposed. Subsequently, the exponential GARCH (EGARCH) model [8], and the GJR-GARCH model [9] have been introduced to study asymmetric volatility; for example, the stock market is more sensitive to bad news rather than good news [10]. There are various multivariate

GARCH models, such as Vector GARCH [11], matrix exponential GARCH [12], Orthogonal GARCH (O-GARCH) [13], generalized Orthogonal GARCH (GO-GARCH) [14], Constant Correlation Covariance (CCC) [15], etc. Among these multivariate GARCH models, the Vector GARCH and the matrix exponential GARCH utilize the $vech(\cdot)$ operator which stacks the lower triangular portion of the covariance matrix to a vector. The O-GARCH model, also known as principal component GARCH [16], assumes the multivariate GARCH processes can be a linear combination of a set of uncorrelated components through means of an orthogonal matrix. The GO-GARCH model is a natural generalization of the O-GARCH model which replaces the orthogonal matrix with the inverse matrix. The CCC model assumes a constant correlation relationship exists, which means that the corresponding entry of the covariance matrix equals the correlation coefficient times the square root of the multiply of two relevant univariate GARCH variances, and the Lagrange Multiplier test has been proposed to check whether a constant correlation exists or not [17]. A detailed review of the multivariate GARCH models can be found in [18]. Other than in finance, GARCH models have also been applied in wind forecasting [19], wind power prediction [20], weather prediction [21], solar energies [22], traffic [23], ECG prediction [24], etc.

In the past few years, time series forecasting techniques were extended to the processing of graph-structured processes, and have gained remarkable success. In the GSP field of study, a causal graph VAR model was proposed with the directed adjacency matrix in [25]. GSP adaptive algorithms for the signals on graphs that are based on the least mean squares (LMS) were introduced in [26], [27], [28]. The adaptive least mean pth power algorithms and the GSP Sign algorithm have further extended the GSP LMS algorithm onto estimating graph signals under impulsive noise [29], [30], [31]. A different time series model which considers the latent variable adopts the graph with low rank plus sparse decomposition was introduced [32]. A graph formulation of the VARMA (G-VARMA) model and graph polynomial-time series models were proposed based on the graph Laplacian matrix [33]. A network formulation of the VAR model [34] and a network multivariate GARCH model [35] were proposed based on the topology information of the adjacency matrix. Lastly, as neural network methods become popular, various researchers have combined graph and neural networks together and proposed the graph neural networks [36].

The existing graph-based time series forecasting models do not account for time-varying variance, nor do they consider the volatility through the neighborhood connection via the structure of the graph and will assign the same impact factor for each neighbor. However, for a given node, it may receive different amounts of influence from different node neighbors. In GSP, the graph Laplacian matrix denotes the diffusion process on graph nodes in the spatial domain [37], [38], and the eigendecomposition of the graph Laplacian gives another perspective on analyzing this diffusion in the graph spectral domain. Therefore, it is advantageous to explore the multivariate GARCH formulation from a graph perspective to utilize the intrinsic information of the graph topology. In this article, we propose a graph framework of multivariate GARCH through the graph Laplacian matrix. The contributions of our article are listed as the following:

1. We first propose a graph framework of multivariate GARCH which we characterize with a graph Laplacian matrix. Then, we introduce three variants of multivariate graphical GARCH models: the Graph GARCH (G-GARCH) model is for the symmetric case, the Graph Exponential GARCH (G-EGARCH) model, and the Graph GJR-GARCH (G-GJR-GARCH) model are for the asymmetric case. The three proposed models utilize the spectral tools of GSP that enable us to analyze multivariate GARCH from a graph perspective. We prove that these graphical GARCH models can be decomposed into a set of univariate GARCH in the graph spectral domain in the expectation sense if the graph signals and their residuals are graph stationary, and the number of estimated parameters will become O(N) rather than O(N^2) with a Laplacian matrix eigendecomposition cost.
2. We applied the three proposed graphical GARCH models to weather prediction and wind power forecasting. With the graph signals and their residuals nearly graph stationary, the proposed graphical GARCH models can achieve more accurate multi-step prediction than traditional GARCH models and the G-VARMA model.
3. Last but not least, this framework allows us to use the principal components to further reduce the estimation cost, which provides a balance selection between the forecasting performance and estimation cost. In the experiments, the framework can use only half the graph frequency components to obtain comparable performance with other time series models in multi-step prediction.

The article is organized as follows: Section II introduces the background information. Section III presents the multivariate GARCH for network modeling. Section IV presents the synthetic data results and the real-world data results. Section V draws conclusions and provides discussions.

II. Preliminaries

A. Graph Signal Processing

For a weighted and undirected graph $G = (V, E, \boldsymbol{W})$, V is the vertex (node) set with N vertices, E is the edge set, and $\boldsymbol{W}$ is the adjacency matrix. If vertices i and j are connected to each other, then E_{ij} belongs to the set E; the corresponding edge weight in the adjacency matrix is W_{ij}. *Graph signal* is a set of values on vertices that gives us a mapping from the vertex set to real numbers $\boldsymbol{x} : V \to R^N$. For instance, $\boldsymbol{x}_t \in R^N$ is a graph signal at time t, which is defined on the underlying graph with N vertices.

The underlying graph topology and edge weights are assumed to be time-invariant throughout this article. Although graphs in real networks might be time-varying, invariant graphs can give a good indication of the average behavior of the networks [39]. Among various choices, one possible way to generate the graph topologies is the k-nearest-neighbor (k-NN) algorithm [40] with Gaussian kernel weights, which is implemented by the GSP

toolbox [41]:

$$w_{ij} = \exp(-d_{ij}/d_{avg}), \tag{1}$$

where d_{ij} implicates the distance between node i and j, and d_{avg} denotes the average distance. Other weights can be used as well such as statistical correlations between variables.

The Laplacian matrix is defined as $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$, where $\boldsymbol{D}$ is a diagonal matrix defined as $D_{ii} = \sum_{j=1}^{N} W_{ij}$. Take $\boldsymbol{L}$ as the *graph shift operator*, since $\boldsymbol{L}$ is positive definite, the eigenvector decomposition of $\boldsymbol{L}$ is

$$\boldsymbol{L} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T, \tag{2}$$

where $\boldsymbol{U} = [\boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_{N-1}]$ is a unitary matrix that consists of N orthonormal eigenvectors, and $\boldsymbol{\Lambda}$ is a diagonal matrix of N eigenvalues on the diagonals. *Graph Fourier transform* (GFT) can transform the graph signals $\boldsymbol{x}_t$ from the spatial domain to the graph frequency (spectral) domain and is defined as

$$\widehat{\boldsymbol{x}}_t = \boldsymbol{U}^T\boldsymbol{x}_t, \tag{3}$$

where $\widehat{\boldsymbol{x}}_t$ denotes the value on graph frequency domain at time t. In GSP, the eigenvalues are sorted in increasing order $0 = \lambda_0 \leq \lambda_2 \leq \ldots \leq \lambda_{N-1}$, where low eigenvalues correspond to low graph frequencies, and vice versa [4]. The inverse GFT transforms $\widehat{\boldsymbol{x}}_t$ from the frequency domain back to the spatial domain and is defined as

$$\boldsymbol{x}_t = \boldsymbol{U}\widehat{\boldsymbol{x}}_t. \tag{4}$$

A *graph filter* h is a function of $\boldsymbol{L}$ and can be seen as a linear operator applied to the graph frequency domain [42], which has an eigenvector decomposition form

$$h(\boldsymbol{L})\boldsymbol{x} := \boldsymbol{U}h(\boldsymbol{\Lambda})\boldsymbol{U}^T\boldsymbol{x}. \tag{5}$$

Graph stationarity is an extension of the notion of stationarity to graphs, and we will use the graph wide-sense stationarity in this article. This requires the first and second moments of the graph signal to be preserved over the graph dimension, which means that the covariance matrix of $\boldsymbol{x}$ should be diagonalizable with respect to the GFT [43]:

$$E\left[\boldsymbol{x}\boldsymbol{x}^T\right] = \boldsymbol{U}\boldsymbol{\Lambda}_{\boldsymbol{x}}\boldsymbol{U}^T. \tag{6}$$

In addition, the sufficient condition for (6) is the graph frequency components of graph signals are uncorrelated [43].

B. VARMA and G-VARMA

The VARMA model is the vector form of the ARMA model, which can be expressed as

$$\boldsymbol{y}_t = \sum_{i=1}^{P} \boldsymbol{A}_i\boldsymbol{y}_{t-i} + \sum_{j=0}^{Q} \boldsymbol{B}_j\boldsymbol{\xi}_{t-j}, \tag{7}$$

where $\boldsymbol{y}_t$ denotes $N \times 1$ random variable, $\boldsymbol{A}_i$ and $\boldsymbol{B}_j$ are N by N matrices, $\boldsymbol{B}_0$ is an identity matrix, and $\boldsymbol{\xi}_t$ denotes the $N \times 1$ vector white noise at time t with *constant covariance*. P is the AR order, and Q is the MA order. Note that the vectors and matrices are expressed in bold.

The G-VARMA model requires the graph signals to be graph stationary and is defined with the underlying Laplacian matrix $\boldsymbol{L}$ [33] as

$$\boldsymbol{y}_t = \sum_{i=1}^{P} \boldsymbol{A}_i(\boldsymbol{L})\boldsymbol{y}_{t-i} + \sum_{j=0}^{Q} \boldsymbol{B}_j(\boldsymbol{L})\boldsymbol{\xi}_{t-j}, \tag{8}$$

where $\boldsymbol{y}_t$ is the time domain graph signals and $\boldsymbol{\xi}_t$ is the random component. In (8) the parameters $\boldsymbol{A}_i(\boldsymbol{L})$ and $\boldsymbol{B}_j(\boldsymbol{L})$ are the AR and MA parameters in the form of graph filters defined in the form of (5).

C. Univariate GARCH

A GARCH model is essentially an ARMA model with the variance being non-stationary and behaving as an ARMA model as well. The ARMA(P, Q) part can be written as

$$y_t = \sum_{i=1}^{P} \alpha_i y_{t-i} + \sum_{j=0}^{Q} \beta_j \xi_{t-j}, \tag{9}$$

where y_t is a random variable at time t, α_i is the AR parameter, β_j is the MA parameter, and ξ_t is the white noise with zero mean and *time-varying variance* σ_t^2.

The GARCH(p, q) part describes the time-varying variance of ξ_t:

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} \alpha_i \xi_{t-i}^2 + \sum_{j=1}^{p} \beta_j \sigma_{t-j}^2. \tag{10}$$

We will next briefly cover two asymmetric GARCH models, which are commonly used in the stock market. The first asymmetric GARCH model is the EGARCH model [8]. The variance of the EGARCH is

$$\ln \sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} \alpha_i \xi_{t-i} + \sum_{i=1}^{q} \gamma_i f(\xi_{t-i}) + \sum_{j=1}^{p} \beta_j \ln \sigma_{t-j}^2, \tag{11}$$

where

$$f(\xi_t) = (|\xi_t| - E(|\xi_t|)). \tag{12}$$

The asymmetry of the EGARCH model in (11) comes from the usage of the $\ln(\cdot)$ function and $f(\xi_t)$. Compared to the square in (10), the $\ln(\cdot)$ function allows negative values as output, which removes the non-negative restrictions of the residuals. Also, the expectation of $f(\xi_t)$ is zero while $f(\xi_t)$ is different for positive ξ_t and negative ξ_t, which could be used to describe the asymmetric effect.

The second asymmetric GARCH model with a specific parametric expression is the GJR-GARCH model [9]. The variance expression of the GJR-GARCH model is

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} \alpha_i \xi_{t-i}^2 + \sum_{i=1}^{q} \gamma_i g(\xi_{t-i}^2) + \sum_{j=1}^{p} \beta_j \sigma_{t-j}^2, \tag{13}$$

where

$$g(\xi_t^2) = \begin{cases} \xi_t^2 & if \quad \xi_t < 0 \\ 0 & \text{otherwise} \end{cases}.$$

One remark on the parameter function $g(\xi_t^2)$ is that its value is non-zero only for the negative components.

D. Multivariate GARCH

A multivariate GARCH model is essentially a VARMA model with the time-varying covariance behaving as a VARMA model as well. The VARMA(P, Q) can be written as

$$\boldsymbol{y}_t = \sum_{i=1}^{P} \boldsymbol{A}_i \boldsymbol{y}_{t-i} + \sum_{j=0}^{Q} \boldsymbol{B}_j \boldsymbol{\xi}_{t-j}. \tag{14}$$

Although this looks similar to the VARMA model, the main difference is that the covariance of white noise $\boldsymbol{\xi}_t$ is *time-varying*. The component $\boldsymbol{\xi}_t$ in (14) has the expression

$$\boldsymbol{\xi}_t = \boldsymbol{H}_t^{1/2} \boldsymbol{z}_t, \tag{15}$$

where $\boldsymbol{H}_t$ denotes the covariance of $\boldsymbol{\xi}_t$, and $\boldsymbol{z}_t$ is the vector form of independent and identically distributed (i.i.d.) white noise. Note that $\boldsymbol{H}_t^{1/2}$ is an N by N positive definite matrix.

The covariance of the O-GARCH model is described as

$$\boldsymbol{H}_t = \boldsymbol{Z} \boldsymbol{V}_t \boldsymbol{Z}, \tag{16}$$

where the associate matrix $\boldsymbol{Z}$ is a N by N orthogonal matrix, and $\boldsymbol{V}_t$ is a diagonal matrix.

III. Network Process Modeling

The VARMA part of the graph multivariate GARCH framework is the same as (8), and the time-varying covariance is

$$Var[\boldsymbol{\xi}_t | \boldsymbol{\xi}_u, u < t] = \boldsymbol{H}_t = \mathbb{E}\left[\boldsymbol{\xi}_t \boldsymbol{\xi}_t^T | \boldsymbol{\xi}_u, u < t\right]. \tag{17}$$

Utilize GFT, then

$$\boldsymbol{H}_t = \boldsymbol{U} \widehat{\boldsymbol{H}}_t \boldsymbol{U}^T, \tag{18}$$

where $\widehat{\boldsymbol{H}}_t$ denotes the covariance in the frequency domain.

From a linear combination perspective, this framework in (18) seems mathematically equivalent to the O-GARCH model in (16). However, the major difference is that we could utilize the graph topology a priori to reduce the estimation cost while the O-GARCH model needs to estimate the unconditional covariance matrix in advance, and endow these uncorrelated components with the meaning of graph frequency, which answers the question of why the associate matrix $\boldsymbol{Z}$ in the O-GARCH model should be orthogonal and avoid the weak correlation estimation problem through graphical settings [14].

A. Graph GARCH Model

The time-varying covariance of the G-GARCH model is computed through

$$\boldsymbol{H}_t = \boldsymbol{W}(\boldsymbol{L}) + \sum_{i=1}^{q} \boldsymbol{C}_i(\boldsymbol{L}) \boldsymbol{\xi}_{t-i} \boldsymbol{\xi}_{t-i}^T + \sum_{j=1}^{p} \boldsymbol{D}_j(\boldsymbol{L}) \boldsymbol{H}_{t-j}, \tag{19}$$

where $\boldsymbol{H}_t$ is the covariance of $\boldsymbol{\xi}_t$. The complete G-GARCH model is a combination of (8) and (19). Note that we use (P, Q, p, q) to denote the four parameters for the GARCH model and Graph GARCH model, where P denotes the AR parameter, Q is the MA parameter, p is the order parameter of the AR part of the GARCH part, and q is the order parameter of the MA part of the GARCH part.

Theorem 1: The G-GARCH model can be decomposed into N uncorrelated GARCH processes in the graph frequency domain in the expectation sense if $\boldsymbol{\xi}_t$ are graph stationary.

Proof: Let us first look at the G-GARCH model in (8) and (19). After applying the GFT to the G-GARCH model, based on [33], the VARMA part would be in the following form:

$$\widehat{\boldsymbol{y}}_t = \sum_{i=1}^{P} \boldsymbol{\Lambda}_{P,i}(\boldsymbol{L}) \widehat{\boldsymbol{y}}_{t-i} + \sum_{j=0}^{Q} \boldsymbol{\Lambda}_{Q,j}(\boldsymbol{L}) \widehat{\boldsymbol{\xi}}_{t-j}, \tag{20}$$

where $\boldsymbol{\Lambda}_{P,i}(\boldsymbol{L})$ and $\boldsymbol{\Lambda}_{Q,j}(\boldsymbol{L})$ are diagonal matrices. This means that (20) is essentially decomposing a VARMA graph signal into N univariate ARMA processes, and the residuals would be

$$\widehat{\boldsymbol{\xi}}_t = \boldsymbol{U}^T \boldsymbol{\xi}_t, \tag{21}$$

and

$$Var\left[\widehat{\boldsymbol{\xi}}_t | \widehat{\boldsymbol{\xi}_u}, u < t\right] = \widehat{\boldsymbol{H}}_t = \mathbb{E}\left[\widehat{\boldsymbol{\xi}}_t \widehat{\boldsymbol{\xi}}_t^T | \widehat{\boldsymbol{\xi}}_u, u < t\right]. \tag{22}$$

Multiply $\boldsymbol{U}^T$ on the left and $\boldsymbol{U}$ on the right of the covariance $\boldsymbol{H}_t$ of the G-GARCH model in (19):

$$\begin{aligned} \widehat{\boldsymbol{H}}_t = \boldsymbol{U}^T \boldsymbol{U} \boldsymbol{\Lambda}_W \boldsymbol{U}^T \boldsymbol{U} + \sum_{i=1}^{q} \boldsymbol{U}^T \boldsymbol{U} \boldsymbol{\Lambda}_{\boldsymbol{C},i} \boldsymbol{U}^T \boldsymbol{\xi}_{t-i} \boldsymbol{\xi}_{t-i}^T \boldsymbol{U} \\ + \sum_{j=1}^{p} \boldsymbol{U}^T \boldsymbol{U} \boldsymbol{\Lambda}_{\boldsymbol{D},j} \boldsymbol{U}^T \boldsymbol{H}_{t-j} \boldsymbol{U}. \end{aligned} \tag{23}$$

Canceling the terms with $\boldsymbol{U}\boldsymbol{U}^T = I$, we will have the following expression:

$$\widehat{\boldsymbol{H}}_t = \boldsymbol{\Lambda}_{\boldsymbol{W}} + \sum_{i=1}^{q} \boldsymbol{\Lambda}_{\boldsymbol{C},i} \widehat{\boldsymbol{\xi}}_{t-i} \widehat{\boldsymbol{\xi}}_{t-i}^T + \sum_{j=1}^{p} \boldsymbol{\Lambda}_{\boldsymbol{D},j} \widehat{\boldsymbol{H}}_{t-j}. \tag{24}$$

Use (22) and take the expectation of (24), then for the left-hand side of the equation, we have

$$\mathbb{E}\left[\widehat{\boldsymbol{H}}_t\right] = \mathbb{E}\left[\mathbb{E}\left[\widehat{\boldsymbol{\xi}}_t \widehat{\boldsymbol{\xi}}_t^T | \widehat{\boldsymbol{\xi}}_u, u < t\right]\right] = \mathbb{E}\left[\widehat{\boldsymbol{\xi}}_t \widehat{\boldsymbol{\xi}}_t^T\right]. \tag{25}$$

For the right-hand side of the equation, we get

$$\begin{aligned} &\boldsymbol{\Lambda}_{\boldsymbol{W}} + \sum_{i=1}^{q} \boldsymbol{\Lambda}_{\boldsymbol{C},i} \mathbb{E}\left[\widehat{\boldsymbol{\xi}}_{t-i} \widehat{\boldsymbol{\xi}}_{t-i}^T\right] + \sum_{j=1}^{p} \boldsymbol{\Lambda}_{\boldsymbol{D},j} \mathbb{E}\left[\widehat{\boldsymbol{H}}_{t-j}\right] = \boldsymbol{\Lambda}_{\boldsymbol{W}} \\ &+ \sum_{i=1}^{q} \boldsymbol{\Lambda}_{\boldsymbol{C},i} \mathbb{E}\left[\widehat{\boldsymbol{\xi}}_{t-i} \widehat{\boldsymbol{\xi}}_{t-i}^T\right] + \sum_{j=1}^{p} \boldsymbol{\Lambda}_{\boldsymbol{D},j} \\ &\times \mathbb{E}\left[\mathbb{E}[\widehat{\boldsymbol{\xi}}_{t-j} \widehat{\boldsymbol{\xi}}_{t-j}^T | \widehat{\boldsymbol{\xi}}_u, u < (t-j)]\right] \\ &= \boldsymbol{\Lambda}_{\boldsymbol{W}} + \sum_{i=1}^{q} \boldsymbol{\Lambda}_{\boldsymbol{C},i} \mathbb{E}\left[\widehat{\boldsymbol{\xi}}_{t-i} \widehat{\boldsymbol{\xi}}_{t-i}^T\right] + \sum_{j=1}^{p} \boldsymbol{\Lambda}_{\boldsymbol{D},j} \mathbb{E}\left[\widehat{\boldsymbol{\xi}}_{t-j} \widehat{\boldsymbol{\xi}}_{t-j}^T\right]. \end{aligned} \tag{26}$$

By the graph stationarity condition, we have

$$\mathbb{E}[\widehat{\boldsymbol{H}}_t] = \mathbb{E}\left[\widehat{\boldsymbol{\xi}}_t \widehat{\boldsymbol{\xi}}_t^T\right] = \mathbb{E}\left[\boldsymbol{U}^T \boldsymbol{\xi}_t \boldsymbol{\xi}_t^T \boldsymbol{U}\right] = \Lambda_{\boldsymbol{\xi}_t}. \tag{27}$$

Now, combine (25) and (26), we get

$$\mathbb{E}[\widehat{\boldsymbol{H}}_t] = \boldsymbol{\Lambda}_{\boldsymbol{W}} + \sum_{i=1}^{q} \boldsymbol{\Lambda}_{\boldsymbol{C},i} \mathbb{E}\left[\widehat{\boldsymbol{\xi}}_{t-i} \widehat{\boldsymbol{\xi}}_{t-i}^T\right] + \sum_{j=1}^{p} \boldsymbol{\Lambda}_{\boldsymbol{D},j} \mathbb{E}\left[\widehat{\boldsymbol{\xi}}_{t-j} \widehat{\boldsymbol{\xi}}_{t-j}^T\right]. \tag{28}$$

Since both sides of (28) are diagonal, we can decompose the G-GARCH model into N uncorrelated GARCH processes in the graph frequency domain in the expectation sense. □

B. Asymmetric Graph GARCH Models

The Graph Exponential GARCH (G-EGARCH) model can be obtained by combining the G-GARCH model in (19) and the EGARCH model in (29). The covariance of the G-EGARCH model is shown in the following equation:

$$\ln \boldsymbol{H}_t = \boldsymbol{W}(\boldsymbol{L}) + \sum_{i=1}^{q} \boldsymbol{C}_i(\boldsymbol{L}) \boldsymbol{U} diag(\boldsymbol{U}^T \boldsymbol{\xi}_{t-i}) \boldsymbol{U}^T + \sum_{i=1}^{q} \boldsymbol{E}_i(\boldsymbol{L}) \boldsymbol{U} diag(f(\boldsymbol{U}^T \boldsymbol{\xi}_{t-i})) \boldsymbol{U}^T + \sum_{j=1}^{p} \boldsymbol{D}_j(\boldsymbol{L}) \ln \boldsymbol{H}_{t-j}, \tag{29}$$

where

$$f(\boldsymbol{U}^T \boldsymbol{\xi}_{t-j}) = (|\boldsymbol{U}^T \boldsymbol{\xi}_{t-j}| - E(|\boldsymbol{U}^T \boldsymbol{\xi}_{t-j}|)). \tag{30}$$

Comparing the G-EGARCH model in (29) with the symmetric G-GARCH model, the asymmetry comes from the $\ln(\cdot)$ function and $f(\boldsymbol{U}^T \boldsymbol{\xi}_{t-j})$, both are the same with the EGARCH model in (11).

Theorem 2: The G-EGARCH model can be divided into N uncorrelated EGARCH processes in the graph frequency domain in the expectation sense if $\boldsymbol{\xi}_t$ are graph stationary.

Proof: From linear algebra [12], the exponent of a symmetric matrix $\boldsymbol{A}$ will also be symmetric:

$$e^{\boldsymbol{A}} = \sum_{i=0}^{\infty} \frac{\boldsymbol{A}^i}{i!}. \tag{31}$$

Combining with the GFT,

$$\boldsymbol{U}^T e^{\boldsymbol{A}} \boldsymbol{U} = \sum_{i=0}^{\infty} \frac{\boldsymbol{U}^T \boldsymbol{A}^i \boldsymbol{U}}{i!} = \sum_{i=0}^{\infty} \frac{(\boldsymbol{U}^T \boldsymbol{A} \boldsymbol{U})^i}{i!} = e^{\boldsymbol{U}^T \boldsymbol{A} \boldsymbol{U}}. \tag{32}$$

The matrix operation $\ln(\cdot)$ is the inverse of the matrix exponential operation; this relation holds for the matrix $\ln(\cdot)$ operation if the matrix is symmetric. By (29) and (32),

$$\boldsymbol{U}^T \ln \boldsymbol{H}_t \boldsymbol{U} = \ln \widehat{\boldsymbol{H}}_t. \tag{33}$$

Now, combine (33) with (32),

$$\ln \widehat{\boldsymbol{H}}_t = \boldsymbol{\Lambda}_W + \sum_{i=1}^{q} \boldsymbol{\Lambda}_{C,i} diag(\widehat{\boldsymbol{\xi}}_{t-i}) + \sum_{i=1}^{q} \boldsymbol{\Lambda}_{E,i} diag\left(f\left(\widehat{\boldsymbol{\xi}}_{t-i}\right)\right) + \sum_{j=1}^{p} \boldsymbol{\Lambda}_{D,j} \ln \widehat{\boldsymbol{H}}_{t-j}. \tag{34}$$

Taking the expectation on both sides, then both sides of the following equation are diagonal by (31) and (27), the resulting expression will be the same as in Theorem 1,

$$\mathbb{E}\left[\ln \widehat{\boldsymbol{H}}_t\right] = \boldsymbol{\Lambda}_W + \sum_{i=1}^{q} \boldsymbol{\Lambda}_{C,i} \mathbb{E}\left[diag\left(\widehat{\boldsymbol{\xi}}_{t-i}\right)\right] + \sum_{i=1}^{q} \boldsymbol{\Lambda}_{E,i} \mathbb{E}\left[diag\left(f\left(\widehat{\boldsymbol{\xi}}_{t-i}\right)\right)\right] + \sum_{j=1}^{p} \boldsymbol{\Lambda}_{D,j} \ln \mathbb{E}\left[\widehat{\boldsymbol{H}}_{t-j}\right]. \tag{35}$$

□

Similar to the GJR-GARCH model, we can derive the G-GJR-GARCH by introducing a parametric expression to represent the asymmetry. The covariance of the G-GJR-GARCH model has the following formulation:

$$\boldsymbol{H}_t = \boldsymbol{W}(\boldsymbol{L}) + \sum_{i=1}^{q} \boldsymbol{C}_i(\boldsymbol{L}) \boldsymbol{\xi}_{t-i} \boldsymbol{\xi}_{t-i}^T + \sum_{i=1}^{q} F_i(\boldsymbol{L}) \boldsymbol{U} diag(g(\boldsymbol{U}^T \boldsymbol{\xi}_{t-i} \boldsymbol{U} \boldsymbol{\xi}_{t-i}^T)) \boldsymbol{U}^T + \sum_{j=1}^{p} \boldsymbol{D}_j(\boldsymbol{L}) \boldsymbol{H}_{t-j}. \tag{36}$$

In (36), the term $\boldsymbol{F}_i(\boldsymbol{L})$ is a graph filter, and the term $g(\boldsymbol{U}^T \boldsymbol{\xi}_{t-i} \boldsymbol{U} \boldsymbol{\xi}_{t-i}^T) = g(\widehat{\boldsymbol{\xi}}_{t-i} \widehat{\boldsymbol{\xi}}_{t-i}^T)$ has the expression

$$g(\widehat{\boldsymbol{\xi}}_{t-i} \widehat{\boldsymbol{\xi}}_{t-i}^T) = \begin{cases} \widehat{\xi}_{t-i,m} \widehat{\xi}_{t-i,n}^T & \text{if } \widehat{\xi}_{t-i,m} < 0 \\ & \text{and } \widehat{\xi}_{t-i,n} < 0, \\ 0 & \text{otherwise} \end{cases}$$

where $\widehat{\xi}_{t-i,m}$ and $\widehat{\xi}_{t-i,n}^T$ denote the mth and the nth residuals among $\widehat{\boldsymbol{\xi}}_{t-i}$. Also, the asymmetric effect is present in the G-GJR-GARCH model because of $g(\widehat{\boldsymbol{\xi}}_{t-i} \widehat{\boldsymbol{\xi}}_{t-i}^T)$ is more sensitive to the negative components than the positive components.

Theorem 3: The G-GJR-GARCH model can be divided into N uncorrelated univariate GJR-GARCH processes in the graph frequency domain in the expectation sense if $\boldsymbol{\xi}_t$ is graph stationary.

Proof: Same as Theorem 1, if we multiply $\boldsymbol{U}$ and its transpose on both side of (36), then use (18) and (36) will result in

$$\widehat{\boldsymbol{H}}_t = \boldsymbol{\Lambda}_{\boldsymbol{W}} + \sum_{i=1}^{q} \boldsymbol{\Lambda}_{\boldsymbol{C},i} \widehat{\boldsymbol{\xi}}_{t-i} \widehat{\boldsymbol{\xi}}_{t-i}^T$$

$$+\sum_{i=1}^{q}\boldsymbol{\Lambda}_{\boldsymbol{F},i}diag\left(g\left(\widehat{\boldsymbol{\xi}}_{t-i}\widehat{\boldsymbol{\xi}}_{t-i}^{T}\right)\right)$$
$$+\sum_{j=1}^{p}\boldsymbol{\Lambda}_{\boldsymbol{D},j}\widehat{\boldsymbol{H}}_{t-j}. \tag{37}$$

By taking the expectation, then both sides of the following equation are diagonal by (36) and (27), the same result holds as in Theorem 1,

$$\mathbb{E}\left[\widehat{\boldsymbol{H}}_t\right]=\boldsymbol{\Lambda}_{\boldsymbol{W}}+\sum_{i=1}^{q}\boldsymbol{\Lambda}_{\boldsymbol{C},i}\mathbb{E}\left[\widehat{\boldsymbol{\xi}}_{t-i}\widehat{\boldsymbol{\xi}}_{t-i}^{T}\right]$$
$$+\sum_{i=1}^{q}\boldsymbol{\Lambda}_{\boldsymbol{F},i}\mathbb{E}\left[diag(g(\widehat{\boldsymbol{\xi}}_{t-i}\widehat{\boldsymbol{\xi}}_{t-i}^{T}))\right]$$
$$+\sum_{j=1}^{p}\boldsymbol{\Lambda}_{\boldsymbol{D},j}\mathbb{E}\left[\widehat{\boldsymbol{H}}_{t-j}\right]. \tag{38}$$

□

C. Principal Component Graph GARCH

We defined the principal component of graph GARCH as

$$\boldsymbol{H}_t'=\boldsymbol{U}(\mathbb{I}\cdot\widehat{\boldsymbol{H}}_t)\boldsymbol{U}^T, \tag{39}$$

where $\boldsymbol{H}_t'$ denotes the principal component approximation of $\boldsymbol{H}_t$. The matrix $\mathbb{I}$ in (39) is the N by N indicator matrix with diagonal entries 0 or 1, which decides which of the graph frequency is adopted as the principal component. This matrix is known as the bandlimited filter and is commonly used in GSP [26].

In practice, for a graph signal with N frequencies, the low-frequency parts such as λ_0 indicate the slow-varying component among the graph, and normally the low-frequency components have a higher impact than the high-frequency components. In other words, the graph signals are assumed to be smooth [4]. With these in mind, we can directly use the first M lower graph frequency with the indicator matrix $\mathbb{I}_M$ to define an ideal graph low-pass filter. Then, the principal components become

$$\boldsymbol{H}_t'=\boldsymbol{U}(\mathbb{I}_M\cdot\widehat{\boldsymbol{H}}_t)\boldsymbol{U}^T, \text{where } \mathbb{I}_{M,ii}=1 \quad \text{if} \quad i<M. \tag{40}$$

It is worth mentioning that the low-pass filter in the graph spectral domain corresponds to diffusion in the graph spatial domain as a low-pass filter could represent the signal propagation among the node neighborhood [37], [38].

Remarks: To estimate a GARCH model, we need to first estimate the ARMA process to obtain the parameters of the ARMA part and the approximate ARMA residuals, then again estimate the ARMA process of the residuals to get the parameters of the GARCH part [44]. Likewise, the estimation process of our graphical GARCH models is the same. Therefore, the discussion of the estimation methods, such as the Maximum Likelihood Estimation (MLE), is not included. Please refer to [44] for the details. The main result of our models is to decompose the multivariate GARCH into several uncorrelated univariate GARCH with parameter saving. For instance, the original parameter size of the GARCH part of the G-GARCH model is $N\times N\times(p+q)$, the G-EGARCH model, and the G-GJR-GARCH model are of size $N\times N\times(p+2\times q)$. By conducting eigendecomposition to the graph Laplacian matrix, we could transfer the graph signals into the graph spectral domain with GFT, and the new parameter size of the GARCH part of G-GARCH is $N\times(p+q)$, and the new parameter size of these asymmetric graphical GARCH models becomes $N\times(p+2\times q)$. The parameter size of the G-VARMA model is $N\times(P+Q)$ after GFT, our models inherit the parameters saving advantage and add GARCH parts with additional $N\times(p+q)$ or $N\times(p+2\times q)$ to model the volatility. In contrast to N univariate GARCH processes, we could utilize the graph topology information and make better predictions with the same model complexity. In addition, we can further reduce the estimation cost using the principal component graph GARCH.

IV. Experimental Studies

In this section, we will test the performance of the graph formulation of multivariate GARCH in synthetic data and real data, including the G-GARCH model, the G-EGARCH model, and the G-GJR-GARCH model. The baseline algorithms will be the ARMA model, the GARCH model, and the G-VARMA model. The code of the experiment is written in MATLAB (All the codes are open at https://github.com/Jacob-Hong17/Graph-GARCH).

A. Evaluation Metrics

In all the experiments, the metrics to evaluate the performance are the MAE (Mean Absolute Error) and the rNMSE (root Normalize Mean Square Error):

$$MAE=\frac{1}{N}\frac{1}{T}\sum_{n=1}^{N}\sum_{t=1}^{T}|\overline{y}_{nt}-y_{nt}|, \tag{41}$$

and

$$rNMSE=\sqrt{\frac{\sum_{n=1}^{N}\sum_{t=1}^{T}||\overline{y}_{nt}-y_{nt}||_2^2}{\sum_{n=1}^{N}\sum_{t=1}^{T}||y_{nt}||_2^2}}. \tag{42}$$

In (41) and (42), N is still the number of nodes, T is the total test sample length, y_{nt} implicates the observation for node n at time t, and $\overline{y}_{nt}$ denotes the prediction for node n at time t.

B. Synthetic Data Experiments

In this part, the purpose is to verify that our framework is effective at multi-step prediction, ranging from 1 step to 6 steps, and test whether the asymmetric graph GARCH models are capable of capturing asymmetric volatility. We would like to start with a toy example with 5 nodes, and the underlying graph topology of the synthetic graph signal is shown in Fig. 1. The graph spectral-domain signals are generated using a G-GARCH(2,1,1,1) for the symmetric case, and a G-EGARCH(2,1,1,1) for the asymmetric case. These two cases share the same ARMA coefficients shown in Table I. The time-varying covariance model coefficients for the G-GARCH and G-EGARCH are given in Table II, respectively. The duration of the time-varying graph signal is 1000 time steps. Then we use the inverse GFT to acquire the data in

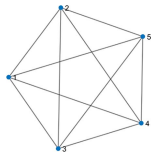


Fig. 1. Graph topology of the synthetic graph signal.

TABLE I
MODEL COEFFICIENTS OF THE ARMA

Graph frequency	Constant	AR(2)	MA(1)
λ_0	-1.5	(0.8, 0.1)	0.4
λ_1	-0.3	(0.6, -0.4)	-0.2
λ_2	-0.2	(-0.4, 0.2)	0.3
λ_3	0.1	(-0.3, 0.4)	-0.1
λ_4	0.05	(0.2, -0.1)	0.2

TABLE II
MODEL COEFFICIENTS OF G-GARCH

Frequency	G-GARCH			G-EGARCH			
	α_0	α_1	β_1	α_0	α_1	β_1	γ_1
λ_0	0.02	0.05	0.9	-3	0.1	0.85	-1.5
λ_1	0.007	0.1	0.85	-4.05	0.1	0.85	-1.5
λ_2	0.007	0.1	0.85	-4.05	0.1	0.85	-1.5
λ_3	0.002	0.1	0.85	-5.3	0.1	0.85	-1.5
λ_4	0.002	0.15	0.80	-5.3	0.1	0.85	-1.5

the graph spatial domain. When generating the graph signal, the signals in the graph frequency domain are purposefully kept to be i.i.d to match the graph stationary assumption. The first 800 samples and the last 200 samples are used as the training and testing set respectively. The experiment is repeated 50 times and each time we re-generate the graph signal randomly.

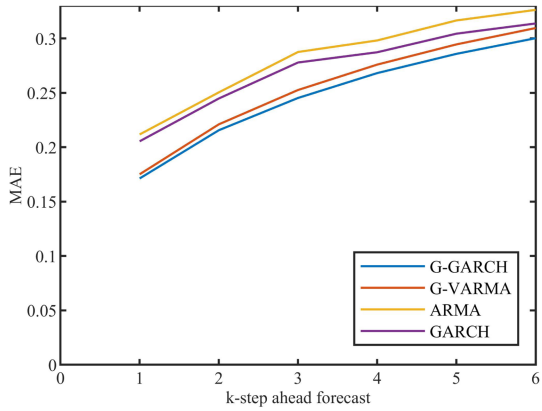
The average MAE and rNMSE of the 50 experiment runs are shown in Fig. 2 for the experiments with symmetric data and Fig. 3 for the experiments with asymmetric data respectively. From Fig. 2, we can clearly observe that the G-GARCH model has lower forecasting MAE and rNMSE than the ARMA, GARCH, and G-VARMA models from 1 step to 6 steps. It can also be observed from Fig. 2 that the performance improvement of the G-GARCH model compared to the G-VARMA model increases as the desired number of forecasting steps increases. Furthermore, in Fig. 3, we can see for asymmetric GARCH data, asymmetric models such as G-EGARCH and G-GJR-GARCH are superior to the G-GARCH both in MAE and rNMSE.

C. Real Data Experiments

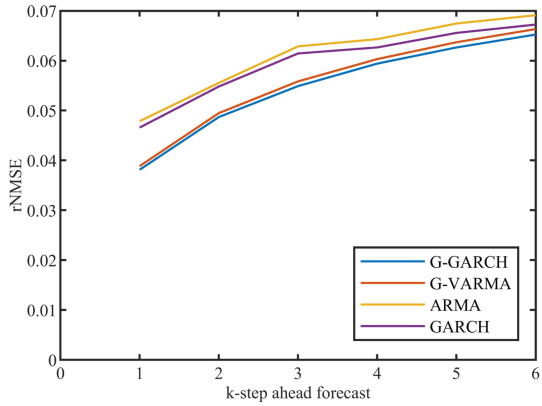
In this section, we evaluate these proposed models in two real-world data scenarios: sensor networks and power grids. Each data set is split into training set, validation set, and test set. The experiments on the real data sets are conducted using the following five-step procedure.

1. *Data collection and preprocessing:* For the real-world data sets, Engle's ARCH test [6] is applied to every univariate time series in all of these data sets to examine if there is an ARCH effect [6] present in the data sets or not. Using this strategy, we have confirmed that all two data sets show the existence of the ARCH effect. Also, we can test whether the graph residuals have a correlation or not. Therefore, using these data sets, we expect that there is a performance gap between models with GARCH and without GARCH.
2. *Graph construction:* The original geographic information contained in the data is the latitude and the longitude of the location where the time series are gathered. To generate the graph topologies, we projected these nodes into a 2D plane with the tool pyproj [45], and generated a distance matrix for all nodes, then we can calculate the graph topologies and weights mentioned in Section II-A.
3. *Model parameter selection:* The model parameters of (P, Q, p, q) are chosen by the lowest rNMSE from the validation set. We first determine the P and Q, then the p and q. Note that the p and q are either ARCH(1) or GARCH(1,1) due to the fact that GARCH(1,1) is sufficient for most applications [46]. All the graphical GARCH models share the same parameters as G-GARCH(P, Q, p, q) for comparison.
4. *Estimation:* We estimate the parameters via the MLE method through MATLAB under the Gaussian distribution setting.
5. *Forecasting:* Make the forecasting with the test set using our proposed graphical models and compare them with the baseline models.

The first real-world data set is the Molene weather data, which has been widely used in the GSP community [43]. Here we select 12 nodes with 744 observations for hourly temperature

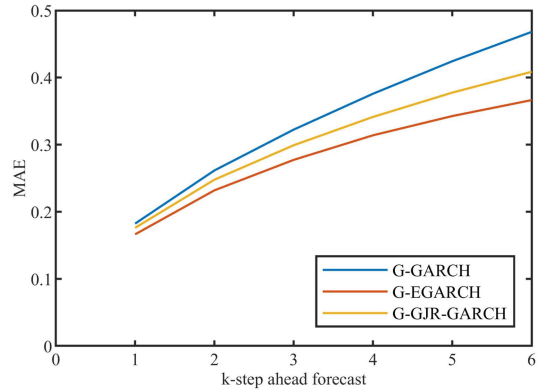


(a) MAE Results

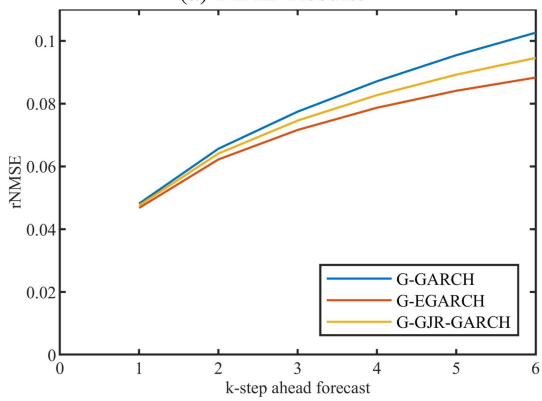


(b) rNMSE Results

Fig. 2. Performance of the G-GARCH model on simulated data.

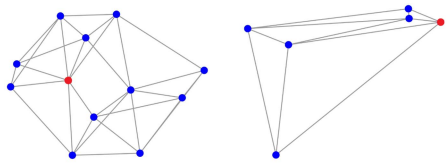


(a) MAE Results



(b) rNMSE Results

Fig. 3. Performance of the asymmetric graphical GARCH models on simulated data.



(a) Molene (b) AEMO

Fig. 4. Graph topology of Molene and AEMO data sets, red points are used in the later demonstration.

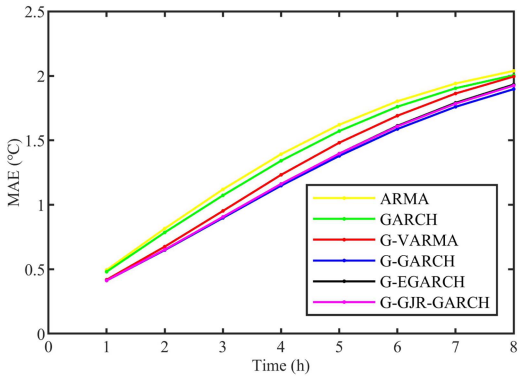
monitoring in France, which could be seen as a sensor network. The data set is split using the following ratio: 35% of the data as the training set, 15% of the data as the validation set, and the rest of the data as the test set. For graph construction, we use k-NN with $k = 4$ to generate the graph is also shown in Fig. 4. In this data set, we reduce the models to behave similarly to an AR model by setting Q to 0, as a consequence in our validation, we only select a P value ranging from 1 to 5. The resulting G-GARCH(2,0,1,1) from the validation is the optimal G-GARCH parameter, while the GARCH baseline is set to GARCH(3,0,1,1).

The second real-world data set of the experiments is the Australian Energy Market Operator (AEMO) wind farms data set from R software[1] [47], which can be viewed as a case of a power grid. Among the wind farms, we select the data from the following 6 close wind farms to be our graph signals: STARHLWF, SNOWTWN1, NBHWF1, NBHWF1.1, WATERLWF, and LKBONNY1. We use the first 20000 data samples, which make the time interval 5 minutes, and the maximum wind farm power is 100 MWH. We split the data set using the following ratio: 70% of the data as the training set, 10% of the data as the validation set, and 20% of the data as the test set. To find the optimal GARCH and G-GARCH parameters (P, Q, p, q), a grid search is performed on the validation set performance. For graph construction, we select $k = 3$ for the AEMO data set, the graph topology can be seen in Fig. 4. The selection of $P \in \{1, 2, 3, 4, 5\}$, and $Q \in \{0, 1, 2, 3, 4, 5\}$. The validation results obtained from the grid search for the experiment on the AEMO data set are G-GARCH(4,4,1,1) and GARCH(2,5,1,1).

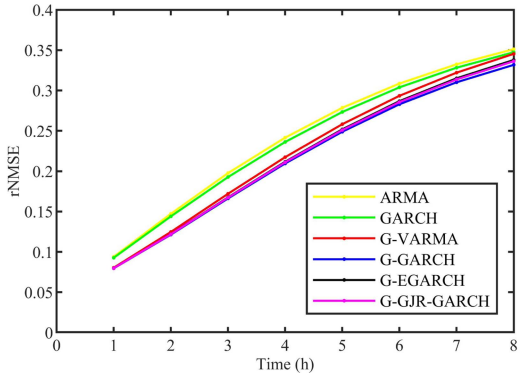
Molene data set: From Fig. 5, the first thing we can observe is that the G-GARCH model is superior to the ARMA model, the GARCH model, and the G-VARMA model for all times in MAE and rNMSE. The G-GARCH model outperforms the G-VARMA model more and more significantly as the time horizon increases, such as the G-GARCH model is superior to the G-VARMA model by 5.5 percent in MAE and 3.7 percent in rNMSE for 7-step forecasting. Furthermore, the graph models are better than the non-graph models, to give an example, the G-VARMA model achieves better results than the ARMA model. The models with volatility are better than the models without volatility, for example, the GARCH model is better than the ARMA model in terms of MAE and rNMSE performance. The asymmetric Graph GARCH models, such as the G-EGARCH model and the G-GJR-GARCH model, are inferior to the G-GARCH model but still superior to the G-VARMA model. One possible reason is that the volatility is symmetric in temperature data [48]. In addition, we only need 4 graph frequency estimations to achieve lower MAE, and 9 graph frequencies to achieve a lower rNMSE than ARMA, GARCH, and G-VARMA models in 3 steps ahead prediction. Finally, the G-GARCH model can match the true signals well in three steps ahead prediction in the selected node mentioned in Fig. 4. The forecasting result of the node in red is shown in Fig. 6(a), and the G-GARCH model has a smaller error 62% of the time, and 6.4% less error on average in contrast to the G-VARMA model from Fig. 6(b). Among all the 12 nodes, the G-GARCH model has better forecasting results than the G-VARMA model in 10 nodes for 3-step ahead. In Fig. 6(c), we show the 95% confidence interval of the G-GARCH model for 3-step prediction.

AEMO data set: In Fig. 7, the G-GARCH model outperforms the ARMA, GARCH, and G-VARMA models for all times in terms of MAE and rNMSE. The G-GARCH model can surpass the G-VARMA model by more than 6 percent in MAE, and about 2.5 percent in rNMSE for 3 hours prediction. The G-EGARCH model and the G-GJR-GARCH model perform slightly better than the G-GARCH model due to the asymmetry factor consideration. For instance, the G-EGARCH model outperforms the G-GARCH model by 1.2 percent in MAE and 0.7 percent in rNMSE. And the G-GJR-GARCH model is superior to the G-GARCH model by 0.7 percent in MAE and 0.6 percent in rNMSE for 3 hours prediction. Furthermore, we can use 4 graph

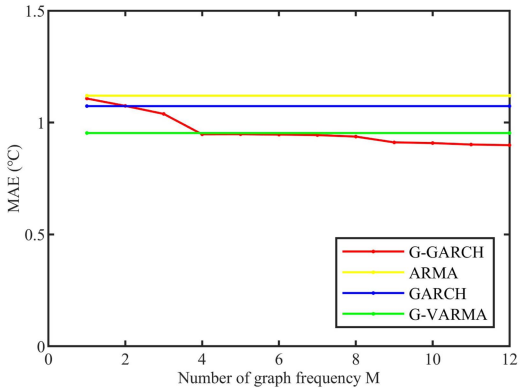
[1]We obtained from AEMO function, onlineVAR(version 0.1-1)



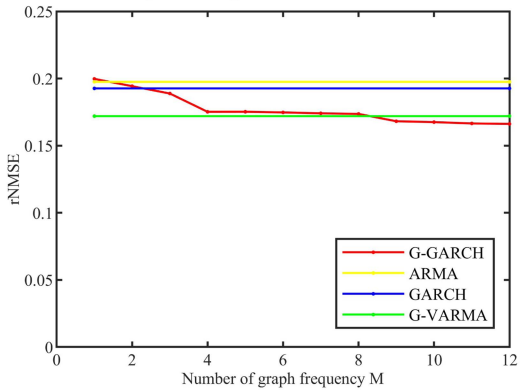
(a) MAE Results



(b) rNMSE Results

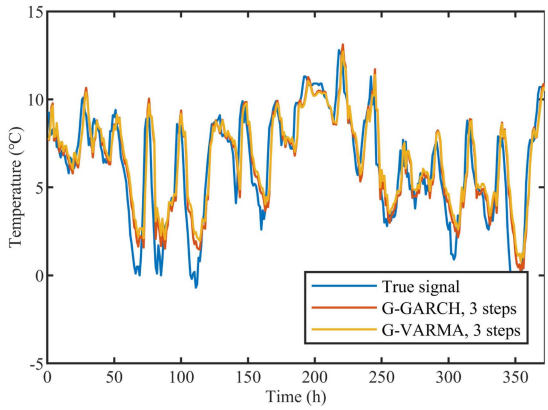


(c) Principal component results of MAE

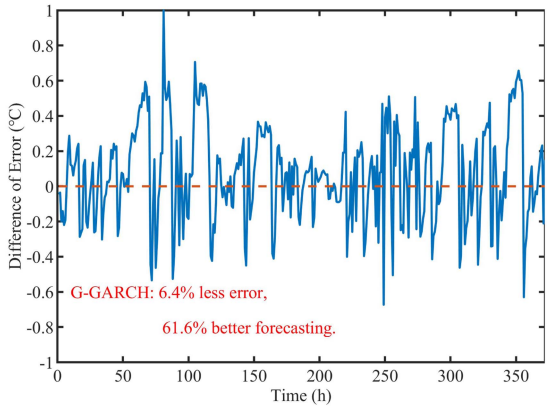


(d) Principal component results of rNMSE

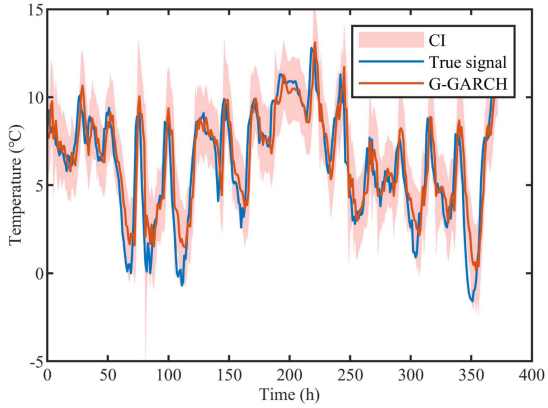
Fig. 5. Experiment results for the Molene data set.



(a) Forecasting Results



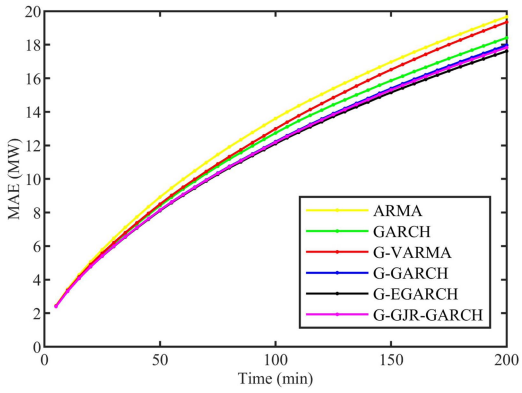
(b) Error(G-VARMA) - Error(G-GARCH)



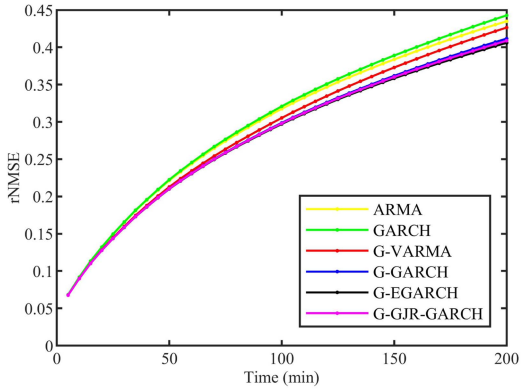
(c) 95% confidence interval

Fig. 6. Forecasting for the Molene data set.

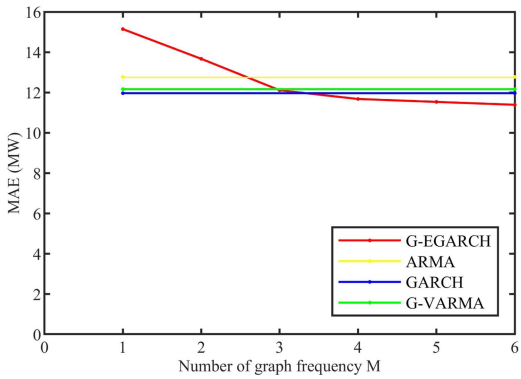
frequencies to get better performance in MAE and rNMSE in 18-step prediction. The asymmetric GARCH models on graphs, such as the G-EGARCH model, are able to predict the true signals well for 18 steps ahead prediction for the selected node mentioned in Fig. 4. The forecasting result of the node in red is shown in Fig. 8(a), the G-EGARCH model has a smaller error 63.5% of the time, and 7.0% less error on average compared with the G-VARMA model from Fig. 8(b). Among all the 6 nodes, the G-EGARCH model has better forecasting results than the G-VARMA model in all 6 nodes for 18 steps ahead. In Fig. 8(c),



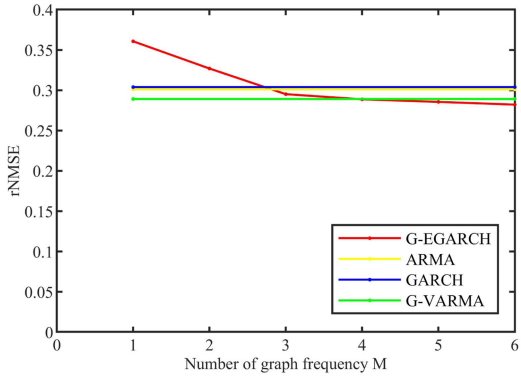
(a) MAE Results



(b) rNMSE Results

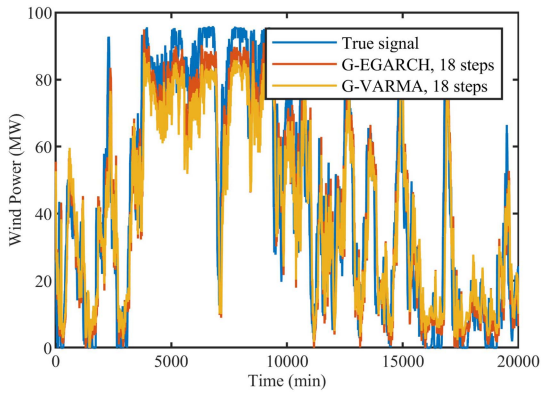


(c) Principal component results of MAE

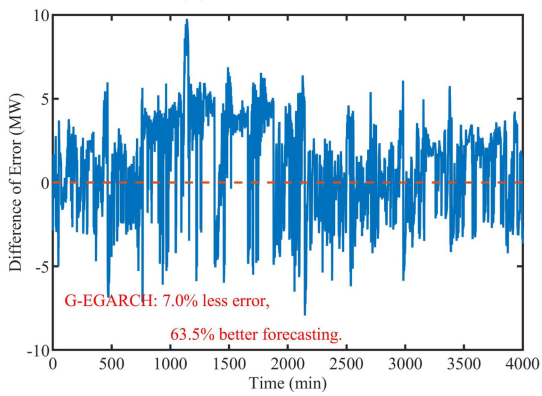


(d) Principal component results of rNMSE

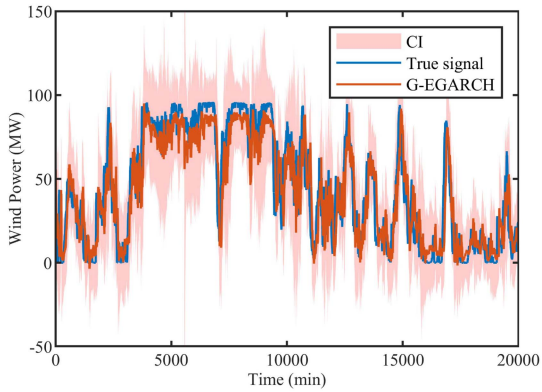
Fig. 7. Experiment results for the AEMO data set.



(a) MAE Results



(b) rNMSE Results



(c) 95% confidence interval

Fig. 8. Forecasting for the AEMO data set.

we demonstrate the 95% confidence interval of the G-EGARCH model for 18-step prediction.

V. Conclusion and Future Work

We have proposed a graph framework for the multivariate GARCH process with significant parameter and complexity reduction. The framework allows us to use the principal components of graph frequency to balance performance and cost. Based on this framework, we are able to propose the G-GARCH model, the G-EGARCH model, and the G-GJR-GARCH model. Among the three models we proposed, we can use G-GARCH

for the symmetric data, and the G-EGARCH model or the G-GJR-GARCH model for asymmetric data. Experiment results show that the graph spectral GARCH methods can surpass the multi-step forecasting performance of the G-VARMA model and of the ordinary GARCH model. Compared with the G-VARMA model, by extending the volatility to the graph frequency domain, our models can achieve better results in high-volatility cases.

Our future research will be directed to studying time-varying graphs, that is graphs with varying branch weights and eventually time-varying topology. We will also explore other applications where signals present high volatility.

REFERENCES

[1] M. Roozbehani, M. A. Dahleh, and S. K. Mitter, "Volatility of power grids under real-time pricing," *IEEE Trans. Power Syst.*, vol. 27, no. 4, pp. 1926–1940, Nov. 2012.

[2] Y. Liu, Y. Qiao, S. Han, Y. Xu, T. Geng, and T. Ma, "Quantitative evaluation methods of cluster wind power output volatility and source-load timing matching in regional power grid," *Energies*, vol. 14, no. 16, 2021, Art. no. 5214.

[3] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83–98, May 2013.

[4] A. Ortega, P. Frossard, J. Kovačević, J. M. Moura, and P. Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proc. IEEE*, vol. 106, no. 5, pp. 808–828, May 2018.

[5] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time Series Analysis: Forecasting and Control*. Hoboken, NJ, USA: Wiley, 2015.

[6] R. F. Engle, "Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation," *Econometrica: J. Econometric Soc.*, vol. 50, pp. 987–1007, 1982.

[7] T. Bollerslev, "Generalized autoregressive conditional heteroskedasticity," *J. Econometrics*, vol. 31, no. 3, pp. 307–327, 1986.

[8] D. B. Nelson, "Conditional heteroskedasticity in asset returns: A new approach," *Econometrica: J. Econometric Soc.*, vol. 59, pp. 347–370, 1991.

[9] L. R. Glosten, R. Jagannathan, and D. E. Runkle, "On the relation between the expected value and the volatility of the nominal excess return on stocks," *J. Finance*, vol. 48, no. 5, pp. 1779–1801, 1993.

[10] R. F. Engle and V. K. Ng, "Measuring and testing the impact of news on volatility," *J. Finance*, vol. 48, no. 5, pp. 1749–1778, 1993.

[11] T. Bollerslev, R. F. Engle, and J. M. Wooldridge, "A capital asset pricing model with time-varying covariances," *J. Political Economy*, vol. 96, no. 1, pp. 116–131, 1988.

[12] H. Kawakatsu, "Matrix exponential GARCH," *J. Econometrics*, vol. 134, no. 1, pp. 95–128, 2006.

[13] H. N. Byström, "Orthogonal GARCH and covariance matrix forecasting: The Nordic stock markets during the Asian financial crisis 1997–1998," *Eur. J. Finance*, vol. 10, no. 1, pp. 44–67, 2004.

[14] R. Van der Weide, "GO-GARCH: A multivariate generalized orthogonal GARCH model," *J. Appl. Econometrics*, vol. 17, no. 5, pp. 549–564, 2002.

[15] T. Bollerslev, "Modelling the coherence in short-run nominal exchange rates: A multivariate generalized ARCH model," *Rev. Econ. Statist.*, vol. 72, pp. 498–505, 1990.

[16] C. Alexander, "Principal component models for generating large Garch covariance matrices," *Econ. Notes*, vol. 31, no. 2, pp. 337–359, 2002.

[17] Y. K. Tse, "A test for constant correlations in a multivariate GARCH model," *J. Econometrics*, vol. 98, no. 1, pp. 107–127, 2000.

[18] L. Bauwens, S. Laurent, and J. V. Rombouts, "Multivariate GARCH models: A survey," *J. Appl. Econometrics*, vol. 21, no. 1, pp. 79–109, 2006.

[19] H. Liu, E. Erdem, and J. Shi, "Comprehensive evaluation of ARMA–GARCH (-M) approaches for modeling the mean and volatility of wind speed," *Appl. Energy*, vol. 88, no. 3, pp. 724–732, 2011.

[20] A. Gupta, K. C. Sharma, A. Vijayvargia, and R. Bhakar, "Very short term wind power prediction using hybrid univariate ARIMA-GARCH model," in *Proc. IEEE 8th Int. Conf. Power Syst.*, 2019, pp. 1–6.

[21] P. Romilly, "Time series modelling of global mean temperature for managerial decision-making," *J. Environ. Manage.*, vol. 76, no. 1, pp. 61–70, 2005.

[22] H. Sun, D. Yan, N. Zhao, and J. Zhou, "Empirical investigation on modeling solar radiation series with ARMA–GARCH models," *Energy Convers. Manage.*, vol. 92, pp. 385–395, 2015.

[23] X. Lin and Y. Huang, "Short-term high-speed traffic flow prediction based on ARIMA-GARCH-M model," *Wireless Pers. Commun.*, vol. 117, no. 4, pp. 3421–3430, 2021.

[24] M. Faal and F. Almasganj, "ECG signal modeling using volatility properties: Its application in sleep apnea syndrome," *J. Healthcare Eng.*, vol. 2021, Art. no. 4894501.

[25] J. Mei and J. M. F. Moura, "Signal processing on graphs: Causal modeling of unstructured data," *IEEE Trans. Signal Process.*, vol. 65, no. 8, pp. 2077–2092, Apr. 2017.

[26] P. Di Lorenzo, S. Barbarossa, P. Banelli, and S. Sardellitti, "Adaptive least mean squares estimation of graph signals," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 2, no. 4, pp. 555–568, Dec. 2016.

[27] P. Di Lorenzo, P. Banelli, E. Isufi, S. Barbarossa, and G. Leus, "Adaptive graph signal processing: Algorithms and optimal sampling strategies," *IEEE Trans. Signal Process.*, vol. 66, no. 13, pp. 3584–3598, Jul. 2018.

[28] M. J. M. Spelta and W. A. Martins, "Normalized LMS algorithm and data-selective strategies for adaptive graph signal estimation," *Signal Process.*, vol. 167, 2020, Art. no. 107326.

[29] N. H. Nguyen, K. Doğançay, and W. Wang, "Adaptive estimation and sparse sampling for graph signals in alpha-stable noise," *Digit. Signal Process.*, vol. 105, 2020, Art. no. 102782.

[30] Y. Yan, R. Adel, and E. E. Kuruoglu, "Graph Normalized-LMP algorithm for signal estimation under impulsive noise," *J. Signal Process. Syst.*, vol. 95, pp. 25–36, 2023.

[31] Y. Yan, E. E. Kuruoglu, and M. A. Altinkaya, "Adaptive sign algorithm for graph signal processing," *Signal Process.*, vol. 200, 2022, Art. no. 108662.

[32] J. Mei and J. M. Moura, "SILVar: Single index latent variable models," *IEEE Trans. Signal Process.*, vol. 66, no. 11, pp. 2790–2803, Jun. 2018.

[33] E. Isufi, A. Loukas, N. Perraudin, and G. Leus, "Forecasting time series with VARMA recursions on graphs," *IEEE Trans. Signal Process.*, vol. 67, no. 18, pp. 4870–4885, Sep. 2019.

[34] X. Zhu, R. Pan, G. Li, Y. Liu, and H. Wang, "Network vector autoregression," *Ann. Statist.*, vol. 45, no. 3, pp. 1096–1123, 2017.

[35] J. Zhou, D. Li, R. Pan, and H. Wang, "Network GARCH model," *Statistica Sinica*, vol. 30, no. 4, pp. 1723–1740, 2020.

[36] Z. Wu, S. Pan, G. Long, J. Jiang, X. Chang, and C. Zhang, "Connecting the dots: Multivariate time series forecasting with graph neural networks," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2020, pp. 753–763.

[37] L. Stanković and E. Sejdić, *Vertex-Frequency Analysis of Graph Signals*. Berlin, Germany: Springer, 2019.

[38] T. Xie, B. Wang, and C.-C. Kuo, "GraphHop: An enhanced label propagation method for node classification," *IEEE Trans. Neural Netw. Learn. Syst*, early access, Mar. 18, 2022, doi: 10.1109/TNNLS.2022.3157746.

[39] E. Isufi, A. Loukas, A. Simonetto, and G. Leus, "Filtering random graph processes over random time-varying graphs," *IEEE Trans. Signal Process.*, vol. 65, no. 16, pp. 4406–4421, Aug. 2017.

[40] L. Jiang, Z. Cai, D. Wang, and S. Jiang, "Survey of improving k-nearest-neighbor for classification," in *Proc. IEEE 4th Int. Conf. Fuzzy Syst. Knowl. Discov.*, 2007, pp. 679–683.

[41] N. Perraudin et al., "GSPBOX: A toolbox for signal processing on graphs," Aug. 2014, *arXiv:1408.5781*.

[42] J. Liu, E. Isufi, and G. Leus, "Filter design for autoregressive moving average graph filters," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 5, no. 1, pp. 47–60, Mar. 2019.

[43] B. Girault, "Stationary graph signals using an isometric graph translation," in *Proc. IEEE 23rd Eur. Signal Process. Conf.*, 2015, pp. 1516–1520.

[44] C. Francq and J.-M. Zakoian, *GARCH Models: Structure, Statistical Inference and Financial Applications*. Hoboken, NJ, USA: Wiley, 2019.

[45] PROJ contributors, "PROJ coordinate transformation software library," *Open Source Geospatial Found.*, 2023. doi: 10.5281/zenodo.5884394. [Online]. Available: https://proj.org/

[46] T. Bollerslev, R. Y. Chou, and K. F. Kroner, "ARCH modeling in finance: A review of the theory and empirical evidence," *J. Econometrics*, vol. 52, no. 1/2, pp. 5–59, 1992.

[47] J. Dowell and P. Pinson, "Very-short-term probabilistic wind power forecasts by sparse vector autoregression," *IEEE Trans. Smart Grid*, vol. 7, no. 2, pp. 763–770, Mar. 2016.

[48] F. E. Benth and J. š. Benth, "The volatility of temperature and pricing of weather derivatives," *Quantitative Finance*, vol. 7, no. 5, pp. 553–561, 2007.

Junping Hong received the B.E. degree in radiation protection and environmental engineering from Lanzhou University, Lanzhou, China, in 2012. He received the master's degree from Tsinghua-Berkeley Shenzhen Institute, Tsinghua University, Beijing, China, in 2022. He will be working toward the Ph.D. degree with Tsinghua-Berkeley Shenzhen Institute in Autumn 2023. His research interests include time series forecasting and graph signal processing.

Yi Yan (Graduate Student Member, IEEE) received the B.S. degree in electrical engineering from the University of California, Santa Barbara, CA, in 2017. He received the M.S. degree majoring in electrical engineering from the University of Southern California, Los Angeles, CA, USA, in 2019. He is currently a third-year Ph.D. student majoring in data science and information technology with Tsinghua-Berkeley Shenzhen Institute, Tsinghua University, Beijing, China. His research interests include dynamic graph models, graph signal processing, and graph neural networks.

Ercan Engin Kuruoglu (Senior Member, IEEE) received M.Phil. and Ph.D. degrees in information engineering from the University of Cambridge, Cambridge, U.K., in 1995 and 1998, respectively. In 1998, he joined Xerox Research Center Europe, Cambridge. In January 2002, he was with ISTICNR, Pisa, Italy, where he became the Chief Scientist in 2020. Since 2022, he has been a Professor with Tsinghua-Berkeley Shenzhen Institute, Shenzhen, China. He was an Associate Editor for IEEE TRANSACTIONS ON SIGNAL PROCESSING and IEEE TRANSACTIONS ON IMAGE PROCESSING. He was the Technical Co-Chair of EUSIPCO 2006 and tutorials Co-Chair of ICASSP 2014. His research interests include the areas of statistical signal and image processing and information and coding theory with applications in computational biology, remote sensing, telecommunications, earth sciences, and astrophysics. He was an Editor-in-Chief of *Digital Signal Processing: A Review Journal*. He is currently the Co-Editor-in-Chief of the *Journal of the Franklin Institute*. He is a Member of the IEEE Technical Committees on Machine Learning for Signal Processing and Image and Vision and Multidimensional Signal Processing. He was a plenary Speaker at DAC 2007, ISSPA 2010, IEEE SIU 2017, Entropy 2018, TBSI-WODS 2019, and tutorial speaker at IEEE ICSPCC 2012. He was an Alexander von Humboldt Experienced Research Fellow with the Max Planck Institute for Molecular Genetics from 2013 to 2015. He was alslo an ERCIM Fellow with INRIA-Sophia Antipolis, France, in 2000.

Wai Kin (Victor) Chan (Senior Member, IEEE) received the B.E. degree from Shanghai Jiao Tong University, Shanghai, China, M.E. degree from Tsinghua University, Beijing, China, and Ph.D. degree from the University of California, Berkeley, CA, USA, respectively. He is a Professor and the Vice Dean of the Tsinghua Shenzhen International Graduate School, Shenzhen, China and Co-Deputy Director of the Tsinghua-Berkeley Shenzhen Institute, Tsinghua University, China. Prior to joining Tsinghua, he was with the Faculty of the Department of Industrial and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY, USA. Dr. Chan was with the program committee for a number of international conferences and panelist for U.S. National Science Foundation, Swiss National Science Foundation, and Macau Science and Technology Development Fund. His research interests include discrete-event simulation, agent-based simulation, big-data analytics and their applications in social networks, business big-data, service systems, healthcare, transportation, energy markets, and manufacturing. He was the conference Chair of the 2022 INFORMS Conference on Service Science, Editor of the 2017 and 2015 Winter Simulation Conference, Program Chair of the 2013 Industrial and System Engineering Research Conference, Editor of the IEEE Conference on Automation Science and Engineering, an Associate Editor for the IIE Transactions, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, and *Asia-Pacific Journal of Operational Research*.