Time-Varying Graph Signal Reconstruction

Kai Qiu, *Student Member, IEEE*, Xianghui Mao, *Student Member, IEEE*, Xinyue Shen, *Student Member, IEEE*, Xiaohan Wang, Tiejian Li, *Member, IEEE*, and Yuantao Gu, *Senior Member, IEEE*

***Abstract*—Signal processing on graphs is an emerging research field dealing with signals living on an irregular domain that is captured by a graph, and has been applied to sensor networks, machine learning, climate analysis, etc. Existing works on sampling and reconstruction of graph signals mainly studied static bandlimited signals. However, many real-world graph signals are time-varying, and they evolve smoothly, so instead of the signals themselves being bandlimited or smooth on graph, it is more reasonable that their temporal differences are smooth on graph. In this paper, a new batch reconstruction method of time-varying graph signals is proposed by exploiting the smoothness of the temporal difference signals, and the uniqueness of the solution to the corresponding optimization problem is theoretically analyzed. Furthermore, driven by practical applications faced with real-time requirements, huge size of data, lack of computing center, or communication difficulties between two nonneighboring vertices, an online distributed method is proposed by applying local properties of the temporal difference operator and the graph Laplacian matrix. Experiments on a variety of synthetic and real-world datasets demonstrate the excellent performance of the proposed methods.**

***Index Terms*—Distributed computing, online reconstruction, time-varying graph signal, signal reconstruction.**

I. Introduction

A. Motivation

GRAPH signal processing is a new research direction that has received extensive attention in recent years [1]–[3]. It mainly analyzes signals living on an irregular domain that is captured by a graph, and has found wide applications in sensor networks [4], [5], machine learning [6]–[8], image processing [9], [10], biomedical fields [11]–[13], climate analysis [14]–[16], et cetera. Existing research topics of graph signal processing include graph filtering [17], [18], graph signal compression [19], [20], graph signal coarsening [21], [22], stationary graph signal processing [23]–[25], graph signal sampling and reconstruction [26]–[28], et cetera.

Time-varying graph signal reconstruction is an important topic with practical applications where missing data problem is prevalent. For example, sea surface temperature can provide important information in the study of the earth's climate dynamics. However, the spacial distribution of the sea surface temperature data collected from ships has varied in history due to economic and political changes, such as the opening of new canals and world wars, which significantly raises demands for reconstructing the global sea surface temperature [29]. For another example, in low-cost commodity sensor network, such as air temperature sensor network, lost data are common [30] due to sensor malfunctions or communication failures, which also calls for the reconstruction of time-varying signals.

In this paper, we study the reconstruction problem of time-varying graph signals. Suppose that a time-varying graph signal is sampled at M consecutive time instances with equal interval, and at each time instance only some of the vertices are observed, i.e., partial values of the graph signal are sampled. The problem is to recover the missing values residing on the unobserved vertices according to the sampled data.

B. Related Works

There has been plenty of theoretical analysis on sampling and reconstruction of bandlimited graph signals [31]–[35]. The concept of Paley-Wiener space is proposed in [31], and it is proved that a graph signal in Paley-Wiener space can be uniquely determined by its uniqueness set. That is to say, a bandlimited graph signal can be exactly reconstructed when sampling in accordance with its uniqueness set. In order to reconstruct bandlimited graph signals from decimation, several iterative reconstruction algorithms of bandlimited graph signals are proposed [32], [33], and the generalizations to local aggregations are also studied [34], [35].

In general, for real-world datasets the signals tend to be smooth rather than strictly bandlimited with respect to the underlying graph. The reconstruction of smooth graph signals can be formulated as a convex optimization problem [32], [36], [37], the objective function of which promotes smoothness of the graph signal and penalizes reconstruction error on the known samples. A kerneled distance penalty term is introduced into the optimization problem in [38], which unifies and subsumes the Tikhonov-regularized graph signal reconstruction schemes in the previous works.

As for time-varying graph signals, a direct assumption is that the signal at each time instant is bandlimited or smooth on graph.

Manuscript received November 15, 2016; revised March 31, 2017 and May 31, 2017; accepted June 6, 2017. Date of publication July 13, 2017; date of current version August 14, 2017. This work was supported in part by the National Natural Science Foundation of China under Grants NSFC 61571263, 61371137, 61531166005, and 51459003 and in part by Tsinghua University Initiative Scientific Research Program under Grant 2014Z01005. The guest editor coordinating the review of this paper and approving it for publication was Prof. Pascal Frossard. *(Corresponding author: Yuantao Gu.)*

K. Qiu is with the Department of Electronic Engineering, Tsinghua University, Beijing 100084, China, and also with the Aviation University Air Force, Changchun 130022, China (e-mail: q1987k@163.com).

X. Mao, X. Shen, X. Wang, and Y. Gu are with the Department of Electronic Engineering, Tsinghua University, Beijing 100084, China (e-mail: maoxh92@sina.com; shenxy13@mails.tsinghua.edu.cn; xiaohan9876543210@163.com; gyt@tsinghua.edu.cn).

T. Li is with the State Key Laboratory of Hydroscience and Engineering, Tsinghua University, Beijing 100084, China (e-mail: litiejian@tsinghua.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSTSP.2017.2726969

Based on this assumption, a distributed reconstruction algorithm of time-varying graph signal is proposed in [39], which performs well in tracking slowly time-varying graph signals. In [40], the authors reconstruct one or multiple smooth graph signals by solving a variation minimization problem. However, these two methods do not take full advantage of the temporal correlation of the graph signals.

Nowadays time-varying graph signal processing has attracted increasing attention [41]–[44]. In order to process massive datasets with irregular structure, filtering and Fourier transform on product graphs are studied in [41], which is also suitable for time-varying signals. In [42], product graphs are also used to describe time-varying data, and visualization of time-varying graph signals is studied based on graph wavelet theory and stacked graph metaphor. In [43], [44], a more in-depth study of frequency analysis of temporal graph signals is conducted, and joint graph and temporal filters are designed. Recently, a new concept of joint stationarity is studied in [45], [46], which generalizes the classical definition of time stationarity to time-varying graph signals, assuming that signals are stationary with respect to both the time direction and the graph topology. Based on joint stationarity, the reconstruction of time-varying graph signals can be formulated as a Tikhonov-regularization problem that constrains the solution to be smooth with respect to both time and graph [45].

C. Contributions

In this paper, the reconstruction of time-varying graph signals is studied. The existing literature of graph signal reconstruction usually assumes that the signal is bandlimited or at least approximately smooth on graph. However, we find that for most real-world time-varying graph signals, their temporal differences usually exhibit better smoothness on graph than the original signals do. Accordingly, a new batch reconstruction method for time-varying graph signals is proposed by exploiting the smoothness of the temporal difference signals on graph. Further, we propose an online distributed method based on the local properties of the temporal difference operator and the graph Laplacian matrix, which allows the signal reconstruction on each vertex to only rely on its neighbors and its own value at the previous time instance.

The rest of this paper is organized as follows. In Section II, some preliminaries are introduced, including the basics of graph signal processing and the reconstruction of smooth graph signals. In Section III, based on an analysis of the smoothness of real-world datasets, the structure of smoothly evolving time-varying graph signal is formulated. In Section IV, a batch reconstruction method of time-varying graph signals is proposed and discussed in detail. In Section V, an online distributed method is designed for scenarios faced with real-time requirements, no computing center and communication difficulties between two non-neighboring vertices, et cetera. In Section VI, the proposed methods are evaluated and compared with some existing methods using synthetic and real-world datasets. Section VII concludes the article.

II. Preliminaries

In this section, prior knowledge on graph Laplacian and the reconstruction of smooth graph signals are briefly introduced.

A. Notations

For the convenience of the readers, we introduce some notations first. The lowercase boldface letters, e.g., $\mathbf{x}$, indicate vectors, while the uppercase boldface letters, e.g., $\mathbf{X}$, denote matrices. Given a vector $\mathbf{x}$, x_i denotes the ith entry of $\mathbf{x}$. For a matrix $\mathbf{X}$, $\mathbf{X}_{i,:}$ and $\mathbf{X}_{:,j}$ denote the ith row and the jth column of $\mathbf{X}$ respectively. Both X_{ij} and $X_{i,j}$ denote the element on the ith row and jth column of matrix $\mathbf{X}$. An undirected, connected, and weighted graph is denoted as $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathbf{W})$, where $\mathcal{V}$ is the set of vertices with $|\mathcal{V}| = N$, $\mathcal{E}$ is the set of edges, and $\mathbf{W}$ is the weighted adjacency matrix. If there is an edge connecting vertices i and j, then $W_{ij} > 0$. Otherwise, $W_{ij} = 0$. For a graph signal $\mathbf{x}$ on graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathbf{W})$, $x(i)$ denotes the scalar residing on vertex i. The operators $\circ$ and $\otimes$ represent the Hadamard product and the Kronecker product respectively. For a vector $\mathbf{x} \in \mathbb{R}^N$, $\mathrm{diag}(\mathbf{x})$ denotes the diagonal matrix with $\{x_1, \ldots, x_N\}$ as its diagonal elements sequentially. The vectorization of matrix $\mathbf{X} \in \mathbb{R}^{N \times M}$ is denoted as $\mathrm{vec}(\mathbf{X}) = \left[\mathbf{x}_{:,1}^{\mathrm{T}}, \ldots, \mathbf{x}_{:,M}^{\mathrm{T}}\right]^{\mathrm{T}}$.

B. Graph Laplacian

In spectral graph theory [47] and graph signal processing [1], the graph Laplacian of an N-vertex, undirected, connected, and weighted graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathbf{W})$ is defined as

$$\mathbf{L} = \mathbf{D} - \mathbf{W}, \tag{1}$$

where $\mathbf{D}$ is a diagonal matrix whose ith diagonal element is equal to the degree of vertex i. The degree of a vertex is the sum of the weights of all the edges connected to it. Graph Laplacian is a real symmetric positive semidefinite matrix, so its eigenvalues are all non-negative.

We can define graph Fourier transform based on graph Laplacian [48]. Provided that the singular value decomposition of $\mathbf{L}$ is $\mathbf{U}\mathbf{\Lambda}\mathbf{U}^{\mathrm{T}}$, where $\mathbf{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_N)$ and $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_N]$, then the graph frequencies are defined as the ascending array of eigenvalues $0 = \lambda_1 < \lambda_2 \leq \cdots \leq \lambda_N$. The frequency component of a graph signal $\mathbf{x} \in \mathbb{R}^N$ corresponding to λ_k equals the inner product of signal $\mathbf{x}$ and basis vector $\mathbf{u}_k$, denoted as

$$\hat{x}(\lambda_k) = \langle \mathbf{x}, \mathbf{u}_k \rangle = \sum_{i=1}^{N} x(i) u_k(i). \tag{2}$$

The frequency components corresponding to small eigenvalues are low frequency components of a graph signal, while the ones corresponding to large eigenvalues are high frequency components.

C. Reconstruction of Smooth Graph Signals

The reconstruction of bandlimited graph signals requires that the signals must be strictly bandlimited on graph. However, most graph signals from real-world datasets tend to be smooth

on graph, rather than exactly bandlimited. The smoothness of graph signals is a qualitative characteristic that expresses how frequently a graph signal varies with respect to the underlying graph [40]. In most literature [1], [3], [36], [37], the smoothness of graph signal $\mathbf{x}$ can be characterized by the graph Laplacian quadratic form

$$S_2(\mathbf{x}) = \mathbf{x}^{\mathrm{T}}\mathbf{L}\mathbf{x}, \tag{3}$$

which can also be written as

$$S_2(\mathbf{x}) = \sum_{(i,j)\in\mathcal{E}} W_{ij}\left[x(j)-x(i)\right]^2 = \sum_{k=1}^{N}\lambda_k \hat{x}(\lambda_k)^2. \tag{4}$$

The smaller the function value, the smoother the signal on graph. From the point of view of vertex domain, $S_2(\mathbf{x})$ is small when signal $\mathbf{x}$ has similar values on neighboring vertices connected by an edge with a large weight. From the point of view of graph frequency domain, $S_2(\mathbf{x})$ is small when the frequency components corresponding to large eigenvalues are small.

The set of ϵ-smooth graph signals on graph $\mathcal{G}$ is defined as

$$\Omega_\epsilon(\mathcal{G}) = \left\{\mathbf{x} : \mathbf{x}^{\mathrm{T}}\mathbf{L}\mathbf{x} \le \epsilon\right\}, \tag{5}$$

where $\epsilon \ge 0$ indicates the smoothness level, i.e., smaller ϵ induces smoother graph signals. As an extreme case, when $\epsilon = 0$, $\Omega_0(\mathcal{G})$ only contains signals constant on the graph.

For a smooth graph signal $\mathbf{x}^*$, a noisy measurement is taken as $\mathbf{y} = \mathbf{j}\circ(\mathbf{x}^* + \mathbf{v})$, where $\mathbf{j} \in \{0,1\}^N$ is the sampling operator defined as

$$j_u = \begin{cases} 1 & u \in \mathcal{S}; \\ 0 & u \notin \mathcal{S}, \end{cases}$$

where $\mathcal{S}$ denotes the set of all sampled vertices. The problem of reconstructing $\mathbf{x}^*$ from $\mathbf{y}$ can be expressed as the following optimization problem [32]

$$\min_{\mathbf{x}} \quad \frac{1}{2}\|\mathbf{y} - \mathbf{j}\circ\mathbf{x}\|_2^2 + \frac{\lambda}{2}\|\mathbf{H}\mathbf{x}\|_2^2, \tag{6}$$

where $\mathbf{H} = \mathbf{U}\boldsymbol{\Gamma}\mathbf{U}^{\mathrm{T}} \in \mathbb{R}^{N\times N}$ is a high-pass graph filter, with the diagonal elements of diagonal matrix $\boldsymbol{\Gamma}$ ascending from the upper left to the lower right, and λ is the regularization parameter. Here we set $\mathbf{H} = \mathbf{L}^{1/2}$, so that $\|\mathbf{H}\mathbf{x}\|_2^2 = \mathbf{x}^{\mathrm{T}}\mathbf{L}\mathbf{x}$. The first term penalizes the error of the samples on the known vertices, and the second term encourages the smoothness of the reconstructed graph signal.

III. Structure of Time-Varying Graph Signals

A time-varying graph signal can be expressed by a matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M]$, where $\mathbf{x}_t$ denotes the signal at time instant t, and each row of $\mathbf{X}$ can be viewed as a time series on the corresponding vertex. The smoothness of time-varying graph signal $\mathbf{X}$ is defined as

$$S_2(\mathbf{X}) = \sum_{t=1}^{M} S_2(\mathbf{x}_t) = \mathrm{tr}\left(\mathbf{X}^{\mathrm{T}}\mathbf{L}\mathbf{X}\right). \tag{7}$$

In this section, the structure of time-varying graph signals is abstracted from real-world datasets.

TABLE I
Smoothness Comparison

Datasets	SST	SLP	PM2.5
$\|\mathbf{X}\mathbf{D}_h\|_F^2$	3.8×10^4	1.4×10^5	3.8×10^5
$S_2(\mathbf{X})$	1.6×10^3	230	210
$S_2(\mathbf{X}\mathbf{D}_h)$	71	91	170

If all the signals $\mathbf{x}_t$, $t \in \{1,\ldots,M\}$, are smooth on graph, then $\mathbf{X}$ could be recovered column by column using reconstruction methods of smooth graph signals. However, in many practical applications, $\mathbf{x}_t$ is not smooth enough on graph, which may lead to poor reconstruction quality by the aforementioned methods. Since a time-varying graph signal exhibits correlations both on graph and along the time direction, the reconstruction quality will be significantly improved by adequately exploiting the correlations of the signal from these two aspects.

We find that for most time-varying graph signals obtained from real-world datasets, the difference signal $\mathbf{x}_t - \mathbf{x}_{t-1}$ exhibits smoothness on graph, even if signals $\mathbf{x}_t$, $t \in \{1,\ldots,M\}$ are not smooth on graph. Take the three real-world datasets that we will use in the experiments as examples. The smoothness of these signals is compared in Table I, where SST, SLP, and PM2.5 denote, respectively, the sea surface temperature dataset [49], the global sea-level pressure dataset [50], and the daily mean PM2.5 concentration of California [51]. More information on these datasets could be found in Section VI. $\mathbf{D}_h$ is the temporal difference operator, denoted as

$$\mathbf{D}_h = \begin{bmatrix} -1 & & & \\ 1 & -1 & & \\ & 1 & \ddots & \\ & & \ddots & -1 \\ & & & 1 \end{bmatrix} \in \mathbb{R}^{M\times(M-1)}, \tag{8}$$

and the temporal difference signal is

$$\mathbf{X}\mathbf{D}_h = \left[\mathbf{x}_2 - \mathbf{x}_1, \mathbf{x}_3 - \mathbf{x}_2, \ldots, \mathbf{x}_M - \mathbf{x}_{M-1}\right].$$

It could be readily read that the temporal difference signals exhibit better smoothness property compared with the original signals. To be noted, since the maximum singular value of $\mathbf{D}_h$ is larger than one, the observed decrease in the quantity $S_2(\mathbf{X}\mathbf{D}_h)$ is due to the inherent data structure and not due to that the operator $\mathbf{D}_h$ reduces the magnitude of $\mathbf{X}$.

Based on the above observation, we provide a mathematical description of the structure of time-varying graph signals in Definition 1.

Definition 1: The ϵ-structured set $\Psi_\epsilon(\mathcal{G})$ composed of smoothly evolving graph signals are defined as

$$\Psi_\epsilon(\mathcal{G}) = \left\{\mathbf{X} : \mathrm{tr}\left(\left(\mathbf{X}\mathbf{D}_h\right)^{\mathrm{T}}\mathbf{L}\mathbf{X}\mathbf{D}_h\right) \le (M-1)\epsilon\right\}. \tag{9}$$

Being consistent with the ϵ-smooth set for time-invariant graph signals, parameter ϵ in Definition 1 indicates the smoothness level of the temporal difference signals.

IV. Batch Reconstruction of Time-Varying Graph Signals

The sampling of a time-varying graph signal $\mathbf{X}^*$ can be represented as

$$\mathbf{Y} = \mathbf{J} \circ \mathbf{X}^* + \mathbf{V}, \tag{10}$$

where $\mathbf{V}$ is the additive Gaussian white noise, and $\mathbf{J} \in \{0,1\}^{N \times M}$ is the sampling operator defined as

$$J_{u,t} = \begin{cases} 1 & u \in \mathcal{S}_t; \\ 0 & u \notin \mathcal{S}_t, \end{cases} \tag{11}$$

where $\mathcal{S}_t$ denotes the set of vertices sampled at time instant t.

In this section, we focus on the case where there is a computing center to collect the sampled data $\mathbf{Y}$ of all the vertices, and the graph Laplacian matrix $\mathbf{L}$ is also known. A batch reconstruction method for time-varying graph signals based on the smoothness of the temporal difference signal is proposed and analyzed in the following subsections, and the noiseless case and the noisy case are discussed respectively.

A. Noiseless Case

In the noiseless case, the observation equation is shown as (10) with $\mathbf{V} = 0$, where $\mathbf{X}^* \in \Psi_\epsilon(\mathcal{G})$ is an ϵ-structured time-varying graph signal according to Definition 1 with structure parameter $\epsilon \geq 0$. The signal reconstruction can be formulated as a constrained optimization problem

$$\min_{\mathbf{X}} \quad \frac{1}{2}\mathrm{tr}\left(\left(\mathbf{X}\mathbf{D}_h\right)^{\mathrm{T}} \mathbf{L}\mathbf{X}\mathbf{D}_h\right) = f_c(\mathbf{X}) \tag{12}$$
$$\text{s.t.} \quad \mathbf{Y} = \mathbf{J} \circ \mathbf{X}.$$

The purpose of minimizing function $f_c(\mathbf{X})$ is to induce smoothness of the temporal difference signal $\mathbf{X}\mathbf{D}_h$ on the graph. Problem (12) can be solved by a gradient projection algorithm with backtracking line search. The iterative procedure is denoted as

$$\mathbf{X}^{k+1} = \left(\mathbf{X}^k - \tau \nabla f_c\left(\mathbf{X}^k\right)\right)^+, \tag{13}$$

where the gradient of function $f_c(\mathbf{X})$ is

$$\nabla f_c\left(\mathbf{X}\right) = \mathbf{L}\mathbf{X}\mathbf{D}_h\mathbf{D}_h^{\mathrm{T}}, \tag{14}$$

and $(\mathbf{X}_p)^+$ denotes the projection of signal $\mathbf{X}_p$ to the signal space $\mathbf{Y} = \mathbf{J} \circ \mathbf{X}$, which can be calculated as

$$(\mathbf{X}_p)^+ = \mathbf{Y} + \mathbf{X}_p - \mathbf{J} \circ \mathbf{X}_p. \tag{15}$$

The iterative stepsize τ is decided by backtracking line search. Given the initial stepsize $\tau = \tau_0$, judge whether the iterative signal satisfies the Armijo rule

$$f_c\left(\mathbf{X}^{k+1}\right) \leq f_c\left(\mathbf{X}^k\right) + \alpha\langle \nabla f_c\left(\mathbf{X}^k\right), \mathbf{X}^{k+1} - \mathbf{X}^k\rangle, \tag{16}$$

where $0 < \alpha < 1$, and the inner product of matrices $\mathbf{P}$ and $\mathbf{Q}$ is defined as $\langle \mathbf{P}, \mathbf{Q}\rangle = \sum_{i,j} P_{ij}Q_{ij}$. If the Armijo rule is not satisfied, the stepsize will be multiplied by a decay factor.

TABLE II
Batch Reconstruction Method Without Noise

Input	$\mathbf{Y}$:	sampled data
	$\mathbf{J}$:	sampling operator
	$\mathbf{L}$:	Laplacian matrix
	α, β, τ_0:	line search parameters
	K:	maximum number of iterations
Output	$\mathbf{X}^k$:	reconstructed signal

Initialization: $\mathbf{X}^0 = 0$;
Iteration:
1) **Gradient update:**
$\mathbf{G}^k = \nabla f_c\left(\mathbf{X}^k\right)$;
2) **Stepsize selection:**
$\tau = \tau_0$;
$\mathbf{X}^{k+1} = \left(\mathbf{X}^k - \tau\mathbf{G}^k\right)^+$;
while $f_c\left(\mathbf{X}^{k+1}\right) > f_c\left(\mathbf{X}^k\right) + \alpha\langle \mathbf{G}^k, \mathbf{X}^{k+1} - \mathbf{X}^k\rangle$
$\tau = \beta\tau$;
$\mathbf{X}^{k+1} = \left(\mathbf{X}^k - \tau\mathbf{G}^k\right)^+$;
$k = k + 1$;
3) **Repeat** steps 1 and 2 until $k = K$.

The proposed method for solving problem (12) is listed in Table II. According to Proposition 2.3.3 in [52], the sequence $\{\mathbf{X}^k\}$ generated by this method converges to a stationary point of the function $f_c(\mathbf{X})$, which is also an optimal solution of problem (12).

Please note that although the solution of model (12) is the signal with the smoothest temporal difference, it is not necessarily the original signal. In other words, there may be a signal that is the same as the original signal at sampling points, but with smoother temporal difference than the original signal. It may be possible to avoid this case by adding more conditions to the graphs or the graph signals, but nowadays it is still an open problem that deserves more studies in the future.

B. Noisy Case

In this part, the measurement noise will be taken into account. By relaxing the constraint to a penalty term in the objective function, problem (12) can be reformulated as an unconstrained optimization problem

$$\min_{\mathbf{X}} \ \frac{1}{2}\|\mathbf{J}\circ\mathbf{X}-\mathbf{Y}\|_F^2 + \frac{\lambda}{2}\mathrm{tr}\left(\left(\mathbf{X}\mathbf{D}_h\right)^{\mathrm{T}}\mathbf{L}\mathbf{X}\mathbf{D}_h\right) = f_u(\mathbf{X}), \tag{17}$$

where λ is the regularization parameter. In the following, we will first provide a theorem to ensure the uniqueness of the solution to problem (17).

Theorem 1: When the following two conditions are simultaneously satisfied by the sampling operator $\mathbf{J}$, the optimal solution to problem (17) is unique.

1) For any $n \in \{1, \ldots, N\}$, there exists $m \in \{1, \ldots, M\}$, such that $J_{n,m} = 1$.
2) There is a fiducial time $m_0 \in \{1, \ldots, M\}$, such that for any $m \in \{1, \ldots, M\}, m \neq m_0$, there exists a vertex $n_m \in \{1, \ldots, N\}$ satisfying that $J_{n_m,m_0} = J_{n_m,m} = 1$.

Proof: Being prepared for the following analyses, we introduce the function $\tilde{f}_u(\mathrm{vec}(\mathbf{X})) = f_u(\mathbf{X})$. As we have

$$\begin{aligned}
&\mathrm{tr}\left(\left(\mathbf{X}\mathbf{D}_h\right)^{\mathrm{T}}\mathbf{L}\mathbf{X}\mathbf{D}_h\right)\\
&= \mathrm{vec}\left(\mathbf{X}\mathbf{D}_h\right)^{\mathrm{T}}\mathrm{vec}\left(\mathbf{L}\mathbf{X}\mathbf{D}_h\right)\\
&= \left[\left(\mathbf{D}_h^{\mathrm{T}}\otimes\mathbf{I}_N\right)\mathrm{vec}(\mathbf{X})\right]^{\mathrm{T}}\left(\mathbf{D}_h^{\mathrm{T}}\otimes\mathbf{L}\right)\mathrm{vec}(\mathbf{X})\\
&= \mathrm{vec}(\mathbf{X})^{\mathrm{T}}\left(\mathbf{D}_h\otimes\mathbf{I}_N\right)\left(\mathbf{D}_h^{\mathrm{T}}\otimes\mathbf{L}\right)\mathrm{vec}(\mathbf{X})\\
&= \mathrm{vec}(\mathbf{X})^{\mathrm{T}}\left(\mathbf{D}_h\mathbf{D}_h^{\mathrm{T}}\otimes\mathbf{L}\right)\mathrm{vec}(\mathbf{X}),
\end{aligned}$$

problem (17) can be equivalently written as

$$\min_{\mathbf{z}}\ \frac{1}{2}\|\mathbf{Q}\left[\mathbf{z}-\mathrm{vec}(\mathbf{Y})\right]\|_2^2+\frac{\lambda}{2}\mathbf{z}^{\mathrm{T}}\left[\left(\mathbf{D}_h\mathbf{D}_h^{\mathrm{T}}\right)\otimes\mathbf{L}\right]\mathbf{z}=\tilde{f}_u(\mathbf{z}),$$

where $\mathbf{Q}=\mathrm{diag}(\mathrm{vec}(\mathbf{J}))\in\mathbb{R}^{MN\times MN}$, and $\mathbf{z}=\mathrm{vec}(\mathbf{X})$. The gradient of $\tilde{f}_u$ can be easily deduced as

$$\nabla\tilde{f}_u(\mathbf{z})=\mathbf{Q}\left[\mathbf{z}-\mathrm{vec}(\mathbf{Y})\right]+\lambda\left[\left(\mathbf{D}_h\mathbf{D}_h^{\mathrm{T}}\right)\otimes\mathbf{L}\right]\mathbf{z}.$$

Further, the Hessian matrix of function $\tilde{f}_u$ is

$$\nabla^2\tilde{f}_u=\mathbf{Q}+\lambda\left(\mathbf{D}_h\mathbf{D}_h^{\mathrm{T}}\right)\otimes\mathbf{L}.$$

According to the definition of sampling operator $\mathbf{J}$ in (11), $\mathbf{Q}$ is a diagonal matrix with diagonal elements either 0 or 1. Hence, $\mathbf{Q}$ is positive semi-definite. Observing that $\mathbf{D}_h\mathbf{D}_h^{\mathrm{T}}$ and $\mathbf{L}$ are both positive semi-definite, we have that $\tilde{f}_u(\cdot)$ is convex, so is true for $f_u(\cdot)$.

Furthermore, we will show that they are strongly convex by verifying that the intersect of the null spaces of $\mathbf{Q}$ and $\left(\mathbf{D}_h\mathbf{D}_h^{\mathrm{T}}\right)\otimes\mathbf{L}$ is merely $\mathbf{0}_N$. We notice that the null space of $\mathbf{D}_h\mathbf{D}_h^{\mathrm{T}}$ is spanned by $\mathbf{1}_M$, while that of $\mathbf{L}$ is spanned by $\mathbf{1}_N$. According the property of Kronecker product, we have that the null space of $\left(\mathbf{D}_h\mathbf{D}_h^{\mathrm{T}}\right)\otimes\mathbf{L}$ is spanned by the columns of $[\mathbf{S},\mathbf{V}]$, $\mathbf{S}=[\mathbf{s}_1,\ldots,\mathbf{s}_N]\in\mathbb{R}^{MN\times N}$, $\mathbf{V}=[\mathbf{v}_1,\ldots,\mathbf{v}_M]\in\mathbb{R}^{MN\times M}$, where

$$\begin{aligned}
\mathbf{s}_t&=\mathbf{1}_M\otimes\mathbf{e}_t^{(N)},\quad 1\le t\le N,\\
\mathbf{v}_t&=\mathbf{e}_t^{(M)}\otimes\mathbf{1}_N,\quad 1\le t\le M,
\end{aligned}$$

where $\mathbf{e}_t^{(N)}\in\mathbb{R}^N$ denotes the tth indicator vector, i.e.,

$$e_{t,k}^{(N)}=\begin{cases}1, & k=t;\\ 0, & k\ne t.\end{cases}$$

It should be noticed that the column rank of $[\mathbf{S},\mathbf{V}]$ is $M+N-1$, and we can always fix a fiducial index $m_0\in\{1,\ldots,M\}$ such that $\left[\mathbf{S},\mathbf{V}_{(m_0)}\right]$ is of full column rank, where $\mathbf{V}_{(m_0)}\in\mathbb{R}^{MN\times(M-1)}$ is formed by deleting the m_0th column of $\mathbf{V}$.

We only need to show that

$$\mathbf{R}=\left[\mathbf{S},\mathbf{V}_{(m_0)}\right]^{\mathrm{T}}\mathbf{Q}\left[\mathbf{S},\mathbf{V}_{(m_0)}\right]\in\mathbb{R}^{(M+N-1)\times(M+N-1)}$$

is positive definite. In fact, matrix $\mathbf{R}$ could be written as $\left[\begin{smallmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{B}^{\mathrm{T}} & \mathbf{C}\end{smallmatrix}\right]$, where

$$\begin{aligned}
\mathbf{A}&=\mathrm{diag}\left(\sum_{m=1}^{M}J_{1,m},\ldots,\sum_{m=1}^{M}J_{N,m}\right),\\
\mathbf{C}&=\mathrm{diag}\left(\sum_{n=1}^{N}J_{n,1},\ldots,\sum_{n=1}^{N}J_{n,m_0-1},\right.\\
&\qquad\left.\sum_{n=1}^{N}J_{n,m_0+1},\ldots,\sum_{n=1}^{N}J_{n,M}\right),\\
\mathbf{B}&=\left[\mathbf{J}_{:,1},\ldots,\mathbf{J}_{:,m_0-1},\mathbf{J}_{:,m_0+1},\ldots,\mathbf{J}_{:,M}\right].
\end{aligned}$$

According to Schur complement, $\mathbf{R}\succ 0$ is equivalent to $\mathbf{A}\succ 0$ and $\mathbf{Z}=\mathbf{C}-\mathbf{B}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}\succ 0$. The first condition in Theorem 1 is an explicit expression of $\mathbf{A}\succ 0$. Hence we only need to show $\mathbf{Z}\succ 0$, whose elements are as the following

$$Z_{ij}=\begin{cases}\sum_{n=1}^{N}\dfrac{J_{n,t_i}\sum_{m'=1,m'\ne t_i}^{M}J_{n,m'}}{\sum_{m=1}^{M}J_{n,m}}, & i=j;\\ -\sum_{n=1}^{N}\dfrac{J_{n,t_i}J_{n,t_j}}{\sum_{m=1}^{M}J_{n,m}}, & i\ne j,\end{cases}$$

where the vector $\mathbf{t}=[1,\ldots,m_0-1,m_0+1,\ldots,M]$ acts as a map between indices. Referring to Gershgorin circle theorem, we have that any eigenvalue λ of $\mathbf{Z}$ lies in at least one Gershgorin disc. The $M-1$ discs of $\mathbf{Z}$ are $[Z_{ii}-D_i,Z_{ii}+D_i]$, $i\in\{1,\ldots,M-1\}$, where $D_i=\sum_{j=1,j\ne i}^{M-1}|Z_{ij}|$ and

$$Z_{ii}-D_i=\sum_{n=1}^{N}\frac{J_{n,t_i}J_{n,m_0}}{\sum_{m=1}^{M}J_{n,m}}. \tag{18}$$

The second condition in Theorem 1 guarantees that for any $i\in\{1,\ldots,M-1\}$ there exists $n_{t_i}\in\{1,\ldots,N\}$ such that $J_{n_{t_i},t_i}J_{n_{t_i},m_0}=1$, which assures that any eigenvalue λ of $\mathbf{Z}$ is positive, so we have $\mathbf{Z}\succ 0$. ■

Remark 1: Theorem 1 gives properties that the sampling operator should satisfy to ensure the uniqueness of the solution to problem (17), so it suggests how to design a sampling operator. The first condition in the theorem means that every vertex should be sampled at least once. It makes sense, in that if there is a vertex never sampled, then adding a constant to the signal on that vertex will not change the value of the cost function.

According to the second condition, there should be a time m_0, such that for any other time $m\ne m_0$, there is a vertex sampled both at m_0 and m. The time m_0 serves as a fiducial time, so that the temporal difference becomes valid.

It should be noted that the two conditions in Theorem 1 are sufficient but not necessary.

According to the form of function $\tilde{f}_u$, problem (17) is to minimize a quadratic function, hence there is a closed form solution to it. When the conditions in Theorem 1 is satisfied, the unique optimal solution $\mathbf{X}^*$ could be obtained via

$$\mathrm{vec}(\mathbf{X}^*)=\left[\mathbf{Q}+\lambda\left(\mathbf{D}_h\mathbf{D}_h^{\mathrm{T}}\right)\otimes\mathbf{L}\right]^{-1}\mathbf{Q}\mathrm{vec}(\mathbf{Y}). \tag{19}$$

However, (19) involves calculating the inverse of a matrix of size $MN\times MN$, which is of high computational complexity,

TABLE III
BATCH RECONSTRUCTION METHOD WITH NOISE

Input	$\mathbf{Y}$:	sampled data
	$\mathbf{J}$:	sampling operator
	$\mathbf{L}$:	Laplacian matrix
	λ:	regularization parameter
	K:	maximum number of iterations
	δ :	tolerance
Output	$\mathbf{X}^k$:	reconstructed signal

Initialization: $\mathbf{X}^0 = 0;\ \Delta\mathbf{X}^0 = -\nabla f_u\left(\mathbf{X}^0\right);$
Iteration:
1) **Stepsize decision:**
$\tau = -\dfrac{\langle \Delta\mathbf{X}^k, \nabla f_u\left(\mathbf{X}^k\right)\rangle}{\langle \Delta\mathbf{X}^k, \nabla f_u\left(\Delta\mathbf{X}^k\right) + \mathbf{Y}\rangle};$
2) **Search direction updating:**
$\mathbf{X}^{k+1} = \mathbf{X}^k + \tau\Delta\mathbf{X}^k;$
$\gamma = \dfrac{||\nabla f_u\left(\mathbf{X}^{k+1}\right)||_F^2}{||\nabla f_u\left(\mathbf{X}^k\right)||_F^2};$
$\Delta\mathbf{X}^{k+1} = -\nabla f_u\left(\mathbf{X}^{k+1}\right) + \gamma\Delta\mathbf{X}^k;$
$k = k + 1;$
3) **Repeat** steps 1 and 2 until $k = K$ or $\|\Delta\mathbf{X}^k\|_F \leq \delta$.

especially when we deal with a long-term tracking problem over a large-scale vertex set. Hence, we propose to solve problem (17) by conjugate gradient method.

In each iteration, the algorithm mainly consists of two steps: the determination of stepsize and the update of the next search direction. Denoting the search direction of the kth step as $\Delta\mathbf{X}^k$, the optimal stepsize τ of the kth step is decided by the line minimization rule

$$\min_{\tau} \quad f_u(\mathbf{X}^k + \tau\Delta\mathbf{X}^k).$$

By taking derivative, we have

$$0 = \frac{\partial f_u(\mathbf{X}^k + \tau\Delta\mathbf{X}^k)}{\partial\tau} = \langle \Delta\mathbf{X}^k, \nabla f_u(\mathbf{X}^k + \tau\Delta\mathbf{X}^k)\rangle, \quad (20)$$

where the gradient of function $f_u(\mathbf{X})$ is

$$\nabla f_u(\mathbf{X}) = \mathbf{J}\circ\mathbf{X} - \mathbf{Y} + \lambda\mathbf{L}\mathbf{X}\mathbf{D}_h\mathbf{D}_h^{\mathrm{T}}. \quad (21)$$

The optimal stepsize is the solution of (20)

$$\tau = -\frac{\langle \Delta\mathbf{X}^k, \nabla f_u\left(\mathbf{X}^k\right)\rangle}{\langle \Delta\mathbf{X}^k, \nabla f_u\left(\Delta\mathbf{X}^k\right) + \mathbf{Y}\rangle}.$$

According to the optimal stepsize τ, the iterative procedure is denoted as

$$\mathbf{X}^{k+1} = \mathbf{X}^k + \tau\Delta\mathbf{X}^k.$$

The search direction of the $(k+1)$th step $\Delta\mathbf{X}^{k+1}$ is the linear combination of the gradient at the $(k+1)$th step and the search direction at the kth step

$$\Delta\mathbf{X}^{k+1} = -\nabla f_u\left(\mathbf{X}^{k+1}\right) + \gamma\Delta\mathbf{X}^k,$$

where $\gamma = \dfrac{||\nabla f_u\left(\mathbf{X}^{k+1}\right)||_F^2}{||\nabla f_u\left(\mathbf{X}^k\right)||_F^2}$.

The proposed method for solving problem (17) is listed in Table III. According to [52], this method can find the global optimal solution of problem (17) after at most MN iterations. When MN is large, after fewer iterations a sufficiently accurate solution can be obtained.

Because sampling in practical applications is usually with noise, we mainly consider problem (17) and its solving method in the following parts of the paper including experiments.

V. ONLINE DISTRIBUTED RECONSTRUCTION METHOD

The batch reconstruction of time-varying graph signals requires to obtain the sampled values within a long time before reconstructing them all together, which leads to high reconstruction delay and high computational complexity. Considering that among the prevalent practical applications, real-time analysis is of vital importance, so we focus on online reconstruction method in this section. Furthermore, we deal with the fully distributed scenario, where no computing center exists, and all communications are limited in the one-hop neighborhood. In such scenario, we propose a distributed reconstruction method to solve the online reconstruction problem.

A. Online Reconstruction of Time-Varying Graph Signals

The aim of online reconstruction of time-varying graph signals is to recover the current unsampled values according to the sampled data of current and previous time. In order to alleviate the storage burden, we only preserve the current sampled data and the previous reconstructed signal.

The online reconstruction of time-varying graph signals can be formulated as an unconstrained optimization problem

$$\min_{\mathbf{x}_t} \ \frac{1}{2}\|\mathbf{j}_t\circ\mathbf{x}_t - \mathbf{y}_t\|_2^2 + \frac{\lambda}{2}\left(\mathbf{x}_t - \hat{\mathbf{x}}_{t-1}\right)^{\mathrm{T}}\mathbf{L}\left(\mathbf{x}_t - \hat{\mathbf{x}}_{t-1}\right). \quad (22)$$

Denote the above objective function as $f_o(\mathbf{x}_t)$, where $\mathbf{x}_t$ is the current signal to be reconstructed, $\hat{\mathbf{x}}_{t-1}$ is the reconstructed signal of the previous time, $\mathbf{j}_t$ is the current sampling operator, $\mathbf{y}_t$ is the current sampled data, and λ is the regularization parameter.

Remark 2: The proposed online reconstruction problem differs from the batch problem (17) in the following two aspects.

1) In the online reconstruction problem, only information before time instant t is available. Hence, the difference of $\mathbf{x}_t$ and $\mathbf{x}_{t+1}$ appearing in (17) is omitted in (22).
2) As problem (22) is solved sequentially, when $\hat{\mathbf{x}}_t$ is reconstructed, there is no feedback to reestimate $\mathbf{x}_{t-1}$, which is different from the joint reconstruction of $\{\mathbf{x}_1, \ldots, \mathbf{x}_M\}$ in solving (17).

Consequently, it is reasonable that the batch method and online method would lead to different reconstruction result.

The gradient of function $f_o(\mathbf{x}_t)$ is

$$\nabla f_o(\mathbf{x}_t) = \mathbf{j}_t\circ\mathbf{x}_t - \mathbf{y}_t + \lambda\mathbf{L}(\mathbf{x}_t - \hat{\mathbf{x}}_{t-1}). \quad (23)$$

Letting the above gradient equal zero, we get the closed form solution of problem (22)

$$\hat{\mathbf{x}}_t = \left(\lambda\mathbf{L} + \mathbf{J}_t\right)^{-1}\left(\lambda\mathbf{L}\hat{\mathbf{x}}_{t-1} + \mathbf{y}_t\right), \quad (24)$$

where $\mathbf{J}_t = \mathrm{diag}\left(\mathbf{j}_t\right)$. With the reconstructed signal at the $(t-1)$th time instant close to the ground truth $\mathbf{x}_{t-1}^*$, we show in

Proposition 1 that the estimation error of the tth time instant is well bounded.

Proposition 1: Assume that the difference signal $\mathbf{g}_t^* := \mathbf{x}_t^* - \mathbf{x}_{t-1}^*$ is ϵ-smooth with respect to graph $\mathcal{G}$, and there is at least one vertex sampled at the tth time instant. With the estimation error of the $(t-1)$th time instant satisfying $\|\hat{\mathbf{x}}_{t-1} - \mathbf{x}_{t-1}^*\|_2 \leq \delta_{t-1}$, and the additive noise at the tth time instant $\|\mathbf{v}_t\|_2 \leq \sigma_t$, we have

$$\|\hat{\mathbf{x}}_t - \mathbf{x}_t^*\|_2 \leq \frac{2\sqrt{2}\theta_t}{\lambda_{\min}} := \delta_t, \tag{25}$$

$$\theta_t = \sqrt{\sigma_t^2 + \lambda(\sqrt{\epsilon} + s_{\max}\delta_{t-1})^2}, \tag{26}$$

where λ_{min} denotes the smallest singular value of matrix $\mathbf{J}_t + \sqrt{\lambda}\mathbf{L}^{1/2}$, and $s_{\max}$ denotes the largest singular value of $\mathbf{L}^{1/2}$.

Proof: For the ease of expression, we introduce the bias vector $\mathbf{e}_t = \hat{\mathbf{x}}_t - \mathbf{x}_t^*, \forall t$. Based on the optimality of the reconstructed signal $\hat{\mathbf{x}}_t$, we have

$$f_o(\hat{\mathbf{x}}_t) \leq f_o(\mathbf{x}_t^*). \tag{27}$$

Or equivalently, we have

$$\begin{aligned}&\|\mathbf{J}_t\mathbf{e}_t - \mathbf{v}_t\|_2^2 + \lambda\|\mathbf{L}^{1/2}(\mathbf{e}_t + \mathbf{g}_t^* - \mathbf{e}_{t-1})\|_2^2 \\ &\leq \|\mathbf{v}_t\|_2^2 + \lambda\|\mathbf{L}^{1/2}(\mathbf{g}_t^* - \mathbf{e}_{t-1})\|_2^2.\end{aligned} \tag{28}$$

Based on the assumption, the second term on the right hand side of (28) is upper bounded.

$$\begin{aligned}\|\mathbf{L}^{1/2}(\mathbf{g}_t^* - \mathbf{e}_{t-1})\|_2 &\leq \|\mathbf{L}^{1/2}\mathbf{g}_t^*\|_2 + \|\mathbf{L}^{1/2}\mathbf{e}_{t-1}\|_2 \\ &\leq \sqrt{\epsilon} + s_{\max}\delta_{t-1}.\end{aligned} \tag{29}$$

Thus, we have

$$\begin{aligned}&\|\mathbf{J}_t\mathbf{e}_t - \mathbf{v}_t\|_2^2 + \|\sqrt{\lambda}\mathbf{L}^{1/2}(\mathbf{e}_t + \mathbf{g}_t^* - \mathbf{e}_{t-1})\|_2^2 \\ &\leq \sigma_t^2 + \lambda(\sqrt{\epsilon} + s_{\max}\delta_{t-1})^2 = \theta_t^2.\end{aligned} \tag{30}$$

Applying triangle inequality to the left hand side of (30), we have

$$\|(\mathbf{J}_t + \sqrt{\lambda}\mathbf{L}^{1/2})\mathbf{e}_t + \sqrt{\lambda}\mathbf{L}^{1/2}(\mathbf{g}_t^* - \mathbf{e}_{t-1}) - \mathbf{v}_t\|_2 \leq \sqrt{2}\theta_t. \tag{31}$$

Thus

$$\begin{aligned}\lambda_{\min}\|\mathbf{e}_t\|_2 &\leq \|(\mathbf{J}_t + \sqrt{\lambda}\mathbf{L}^{1/2})\mathbf{e}_t\|_2 \\ &\leq \sqrt{2}\theta_t + \sqrt{\lambda}\|\mathbf{L}^{1/2}(\mathbf{g}_t^* - \mathbf{e}_{t-1})\|_2 + \|\mathbf{v}_t\|_2 \\ &\leq \sqrt{2}\theta_t + \sqrt{\lambda}(\sqrt{\epsilon} + s_{\max}\delta_{t-1}) + \sigma_t \\ &\leq 2\sqrt{2}\theta_t.\end{aligned} \tag{32}$$

Finally, we have

$$\|\mathbf{e}_t\|_2 \leq \frac{2\sqrt{2}\theta_t}{\lambda_{\min}}, \tag{33}$$

and complete the proof. ■

As shown in Proposition 1, the estimation error achieved by solving the online reconstruction problem (22) is upper bounded by a function of the noise level σ_t and the estimation error at the previous time instant δ_{t-1}. Furthermore, by affecting parameter $\lambda_{\min}$, the sampling set at the tth time instant plays an important role in the bound δ. With the graph $\mathcal{G}$ being connected, the null space of the graph Laplacian $\mathbf{L}$ is composed of the constant vectors, so is that of $\mathbf{L}^{1/2}$. As long as there is at least one vertex sampled at the tth time instant, we have $\mathbf{1}_N^{\mathrm{T}}\mathbf{J}_t\mathbf{1}_N \neq 0$, which indicates that matrix $\mathbf{J}_t + \sqrt{\lambda}\mathbf{L}^{1/2}$ is positive definite with $\lambda_{\min} > 0$. Furthermore, when the vertex set is fully sampled at the tth time instant, i.e., $\mathbf{J}_t = \mathbf{I}_N$, the parameter $\lambda_{\min}$ is maximized, and equals 1; correspondingly, the error bound δ_t is minimized. To be noted, as shown in Proposition 1, the estimation error may be accumulated over time. To avoid large deviation from the ground true signal, an estimation reset may be required after several time instants.

If the graph has too many vertices, the above inverse matrix will be hard to calculate. In such case, problem (22) can be solved by a gradient descent method

$$\mathbf{x}_t^{k+1} = \mathbf{x}_t^k - \mu\nabla f_o(\mathbf{x}_t^k), \tag{34}$$

where $\mathbf{x}_t^k$ is the iterative value of vector $\mathbf{x}_t$ at the kth step, and μ is the stepsize. Because problem (22) is convex, $\mathbf{x}_t^k$ will converge to the global optimal solution $\hat{\mathbf{x}}_t$ after sufficient iterations.

B. Distributed Reconstruction of Time-Varying Graph Signals

In this part, we propose a distributed reconstruction method based on a distributed calculation of the gradient in (34).

Denoting $\mathbf{d}_t^k = \mathbf{x}_t^k - \hat{\mathbf{x}}_{t-1}$ and plugging (23) into (34), we have

$$\mathbf{x}_t^{k+1} = \mathbf{x}_t^k - \mu\left(\mathbf{j}_t \circ \mathbf{x}_t^k - \mathbf{y}_t\right) - \mu\lambda\mathbf{L}\mathbf{d}_t^k. \tag{35}$$

Due to the local property of the graph Laplacian $\mathbf{L}$, the update in (35) can be implemented locally. The iterative formula of sampled vertex s at time instant t is

$$\begin{aligned}x_t^{k+1}(s) = (1-\mu)x_t^k(s) + \mu y_t(s) - \mu\lambda \sum_{j\in N(s)} W_{sj} \\ \times \left[d_t^k(s) - d_t^k(j)\right],\end{aligned} \tag{36}$$

where W_{sj} is the element of the weighted matrix $\mathbf{W}$, and $N(s)$ is the set of the neighbors of vertex s. The update of $x_t^{k+1}(s)$ is composed of two parts. One is the new information from the sampled value of vertex s, and the other is the value change of its neighboring vertices. The iterative formula of unsampled vertex u at time instant t is

$$x_t^{k+1}(u) = x_t^k(u) - \mu\lambda \sum_{j\in N(u)} W_{uj}\left[d_t^k(u) - d_t^k(j)\right]. \tag{37}$$

The update of $x_t^{k+1}(u)$ is from the value change on its neighboring vertices. The overall online distributed reconstruction method is presented in Table IV.

VI. Experiments

In the experiments we have two synthetic datasets and three real-world datasets which are the sea surface temperature [49], the global sea-level pressure [50], and the daily mean PM2.5 concentration of California [51], respectively. We compare the proposed batch and online distributed methods with natural

TABLE IV
ONLINE DISTRIBUTED METHOD AT TIME INSTANT t

Input	$\mathbf{y}_t$:	sampled data at time instant t
	$\hat{\mathbf{x}}_{t-1}$:	the recovered signal at time instant $t-1$
	$\mathbf{j}_t$:	sampling operator at time instant t
	$\mathbf{W}$:	Adjacency matrix
	λ :	regularization parameter
	μ :	stepsize
	K :	maximum number of iterations
Output	$\hat{\mathbf{x}}_t$:	reconstructed signal

Initialization: $\mathbf{x}_t^0 = \hat{\mathbf{x}}_{t-1}$;
Iteration:
1) **Distributed gradient descent:**
For each vertex $s \in \{1, \ldots, N\}$ do:
$d_t^k(s) = x_t^k(s) - \hat{x}_{t-1}(s)$;
if $J_{s,t} = 1$
$x_t^{k+1}(s) = (1-\mu)x_t^k(s) + \mu y_t(s) - \mu\lambda \sum_{j \in N(s)} W_{sj}\left(d_t^k(s) - d_t^k(j)\right)$;
else
$x_t^{k+1}(s) = x_t^k(s) - \mu\lambda \sum_{j \in N(s)} W_{sj}\left(d_t^k(s) - d_t^k(j)\right)$;
$k = k+1$;
2) **Repeat** step 1 until $k = K$.

neighbor interpolation [53], low-rank matrix completion [54], the smooth graph signal reconstruction method [32] which is called graph regularization, and the Tikhonov-regularization method [45] which is called graph-time Tikhonov in this paper.

In each experiment, simple random sampling scheme is adopted to select the sampled data. For example, when the sampling rate is 40%, it means that 40% of the vertices are randomly sampled at each time instant, and the goal is to reconstruct the whole signal according to the sampled data. In our proposed methods, the maximum number of iterations K is 10^4, the tolerance δ is 10^{-6}, and the stepsize μ of the online distributed method is taken as 10^{-2}. The parameters of all the methods are scanned to generate best performance.[1]

The reconstruction result is evaluated by the root-mean-square error (RMSE)

$$\mathrm{RMSE} = \frac{\|\hat{\mathbf{x}} - \mathbf{x}^*\|_2}{\sqrt{N_x}}, \tag{38}$$

where $\mathbf{x}^*$ is the ground truth, $\hat{\mathbf{x}}$ is the recovered signal, and N_x is the length of the signals. When the signal is in matrix form, it is vectorized to fit (38).

A. Synthetic Graphs and Signals

In a 100×100 square area, 100 points are generated randomly from the uniform distribution. A graph is constructed by k-nearest neighbors algorithm to characterize the geographical adjacency of these points, with the weight of each edge inversely proportional to the square of the distance between its two vertices.

[1]The MATLAB codes for the proposed methods and all experiments are available at http://gu.ee.tsinghua.edu.cn/codes/Timevarying_GS_Reconstruction.zip

In order to generate a 100×600 time-varying graph signal $\mathbf{X}$, we define $\mathbf{L}^{-1/2}$ first. Note that the graph Laplacian $\mathbf{L} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{\mathrm{T}}$, where $\mathbf{\Lambda} = \mathrm{diag}(0, \lambda_2, \ldots, \lambda_N)$. Define $\mathbf{\Lambda}^{-1/2} = \mathrm{diag}(0, \lambda_2^{-1/2}, \ldots, \lambda_N^{-1/2})$, and $\mathbf{L}^{-1/2} = \mathbf{U}\mathbf{\Lambda}^{-1/2}\mathbf{U}^{\mathrm{T}}$.

The initial signal $\mathbf{x}_1$ is a low frequency graph signal with energy 10^4. At the follow-up time t ($t \geq 2$), an i.i.d. white Gaussian signal $\mathbf{f}_t$ is generated and rescaled so that $\|\mathbf{f}_t\|_2 = \epsilon$, and then $\mathbf{x}_t = \mathbf{x}_{t-1} + \mathbf{L}^{-1/2}\mathbf{f}_t$. Therefore, the generated $\mathbf{X}$ satisfies Definition 1.

In the following subsections, if not specifically stated, the experimental settings are as follows: 1) The graph is constructed by k-nearest neighbors algorithm with $k = 5$. 2) The time-varying graph signal is generated under Definition 1 with $\epsilon = 1$. 3) The measurement noise at each point is i.i.d. Gaussian noise $\mathcal{N}(0, \sigma^2)$, where the standard deviation of noise $\sigma = 0.1$. 4) Simple random sampling scheme is adopted with a sampling rate of 40%.

1) Selection of k in Graph Construction: In this experiment, k-nearest neighbors algorithm is applied to construct the graph, so we study the impact of k to the performance of the proposed methods first. The reconstruction error of the proposed methods under different k is shown in Fig. 1(a). As can be seen, in order to achieve good performance, k should be neither too small nor too large. In Fig. 1(b), we plot the constructed graphs when $k = 2, 3, 5, 10$, respectively. Intuitively, small values of k would lead to a sparse graph. However, if k is too small, the constructed graph may be unconnected and the adjacency between vertices can not be well described, such as the case $k = 2$ in Fig. 1(b). As a result, the performance of the proposed methods will be seriously degraded. If k is too large, boundaries between communities will be less distinct, such as the case $k = 10$ in Fig. 1(b), so the performance of the proposed methods will decrease slightly. Therefore, we heuristically take $k = 5$ in our experiments.

2) Selection of Parameter λ: To begin with, the proposed batch method is tested under different noise level σ, and for each σ the value of parameter λ is swept. The result is shown in Fig. 2(a). For each σ, as the value of λ increases, the reconstruction error decreases first, and then increases after reaching the lowest point, i.e., the optimal λ. When σ increases, the optimal λ also increases. It should also be noticed that the value of λ can be within a large range for the reconstruction error to be small, so the performance is not extremely sensitive to the choice of λ.

Secondly, the proposed online distributed method is tested. From the result in Fig. 2(b), we can draw a similar conclusion to the one in the batch method. In addition, for each σ the lowest recovery error of the online distributed method is larger than that of the batch method. It is reasonable because the batch method utilizes information of all the time for reconstruction, while the online distributed method only utilizes information of current time and previous time.

3) The Time Complexity With Respect to the Number of Vertices and Time Steps: In order to evaluate the time complexity of all methods, we test the running time under different time steps and number of vertices. All experiments are carried out in MATLAB R2014b on a 3.2 GHz macOS with 32 GB memory,

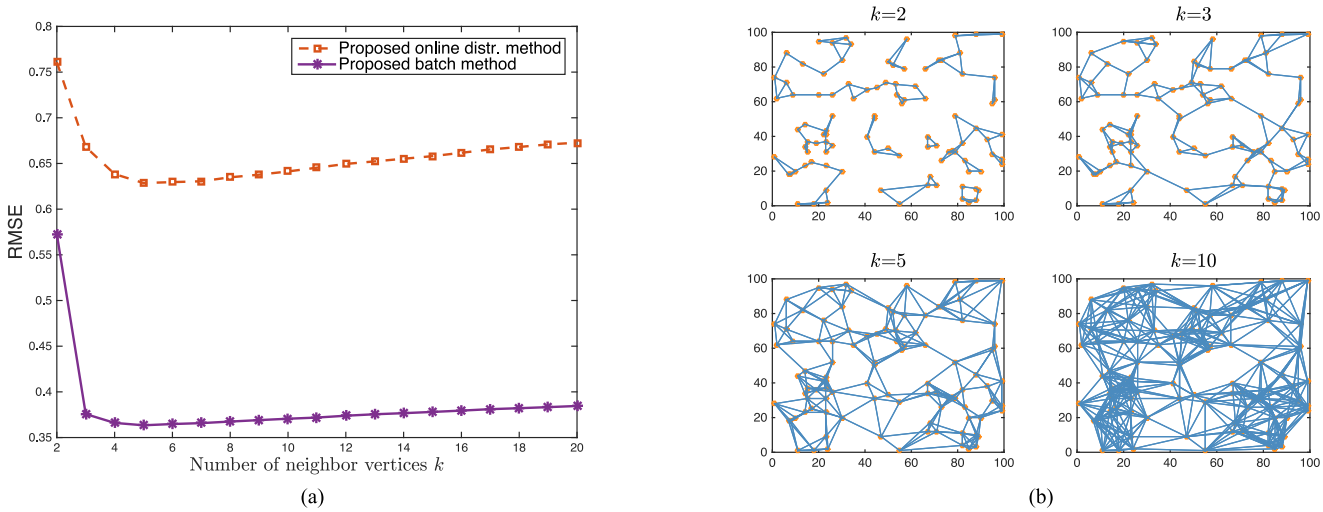


Fig. 1. (a) The reconstruction error of the proposed methods under different k, where k is the neighbor number at the graph construction stage using k-nearest neighbors algorithm. (b) Graphs constructed by k-nearest neighbors algorithm. The weight of each edge is inversely proportional to the square of the distance between its two vertices.

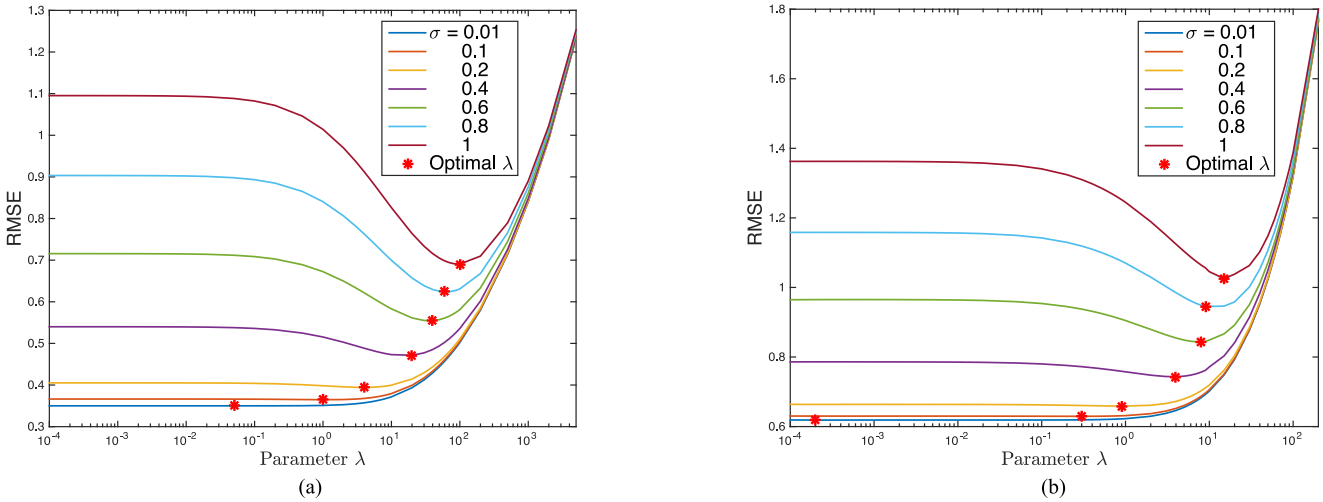


Fig. 2. (a) The reconstruction error of the proposed batch method on the synthetic dataset. Under different noise level σ, the value of parameter λ is swept. The red star on each curve denotes the lowest point, i.e., the optimal λ. (b) The reconstruction error of the online distributed method under different noise level.

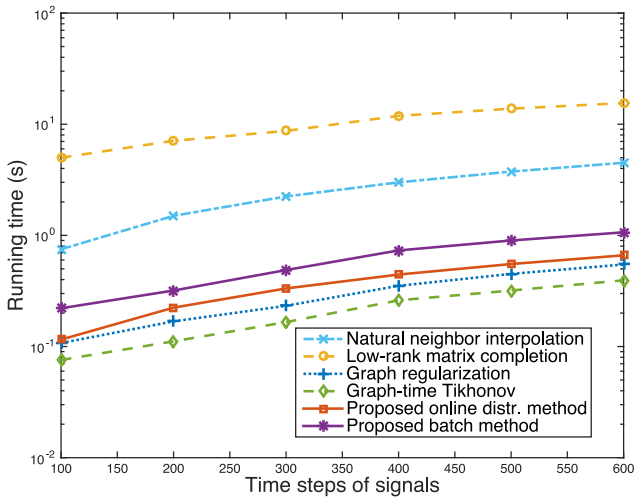
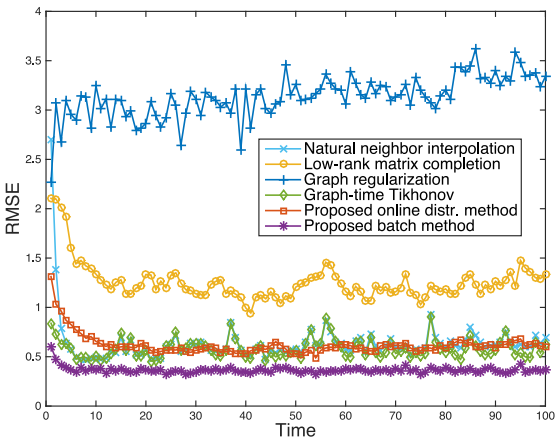


Fig. 3. The average running time of all the methods under different time steps.

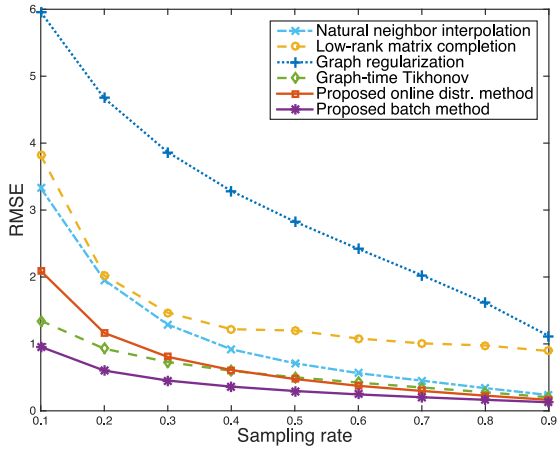
TABLE V
RUNNING TIME UNDER DIFFERENT NUMBER OF VERTICES

Algorithms \ Number of vertices	25	50	100	200
Natural neighbor interpolation	0.36	0.72	1.53	3.17
Low-rank matrix completion	1.94	3.56	7.42	18.17
Graph regularization	0.044	0.075	0.17	0.40
Graph-time Tikhonov	0.042	0.071	0.11	0.29
Proposed online distr. method	0.11	0.12	0.24	0.31
Proposed batch method	0.15	0.22	0.32	0.79

and the unit of running time is second. Fig. 3 shows the average running time of all the methods under different time steps of signals when sampling rate is 40%, and the average running time of all the methods under different number of vertices is listed in Table V. We can see that graph-time Tikhonov is the fastest one in both cases. The speed of the two proposed methods are also satisfactory, and the running time of the proposed online



(a)



(b)

Fig. 4. (a) The reconstruction error when sampling rate is 40% on the synthetic dataset. Only the first 100 time instants are drawn to facilitate the display. Every method has been run 50 times, and the RMSE of each time instant is an average of 50 tests. (b) The reconstruction error under different sampling rate. The RMSE of each method is an average of 50 tests.

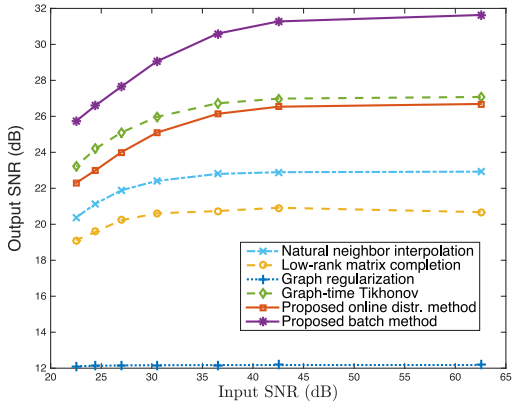
distributed method is about $45\% \sim 75\%$ of the proposed batch method.

4) The Reconstruction Error With Respect to Sampling Rate: Fig. 4(a) shows the reconstruction error versus time of all the methods when sampling rate is 40%, and only the first 100 time instants are drawn to facilitate the display. The total RMSEs of all the methods are 0.92, 1.21, 6.57, 0.60, 0.61 and 0.36 respectively, which are listed in the same order as in the figure legend. As can be seen, the performance of the proposed batch method is better than all the comparison methods, and the performance of the proposed online distributed method is almost the same as the graph-time Tikhonov method.

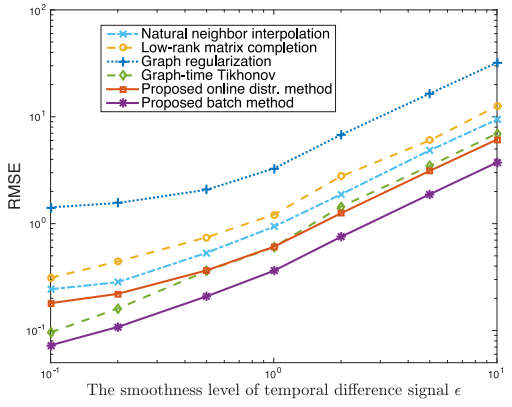
We then test the performance of all the methods under different sampling rate, and the results are displayed in Fig. 4(b). One can read from the results that the recovery error of all the methods decreases with the increase of the sampling rate, while the proposed batch method is always superior to the others. The performance of the proposed online distributed method is not as good as the graph-time Tikhonov method when sampling rate is low, but it outperforms the latter when the sampling rate is higher than 40%. Besides, the performance of the graph regularization is much worse than the others, for the reason that the graph signal is not smooth enough.

It should be mentioned that the natural neighbor interpolation, low-rank matrix completion, and the proposed batch method are all batch methods, which utilize information of all the time for reconstruction, while both graph regularization and the proposed online distributed method are online methods which use much less information for reconstruction. Thus it is reasonable for the performance degradation of online methods compared with batch methods. Note that the graph-time Tikhonov method used here also utilizes information of all the time for reconstruction. However, it can be implemented by a joint filter defined in the joint spectral domain [45].

5) The Reconstruction Error With Respect to Measurement Noise and Smoothness Level: The reconstruction error of all the methods under different measurement noise level and under



(a)



(b)

Fig. 5. (a) The reconstruction error under different input SNR on the synthetic dataset. (b) The reconstruction error under different smoothness level ϵ. The RMSE of each method is an average of 50 tests.

different smoothness level of the temporal difference signal is shown in Fig. 5. From Fig. 5(a) one can read that the performance of all the methods improves with the increase of input SNR, and the proposed batch method is always the best one, while the per-

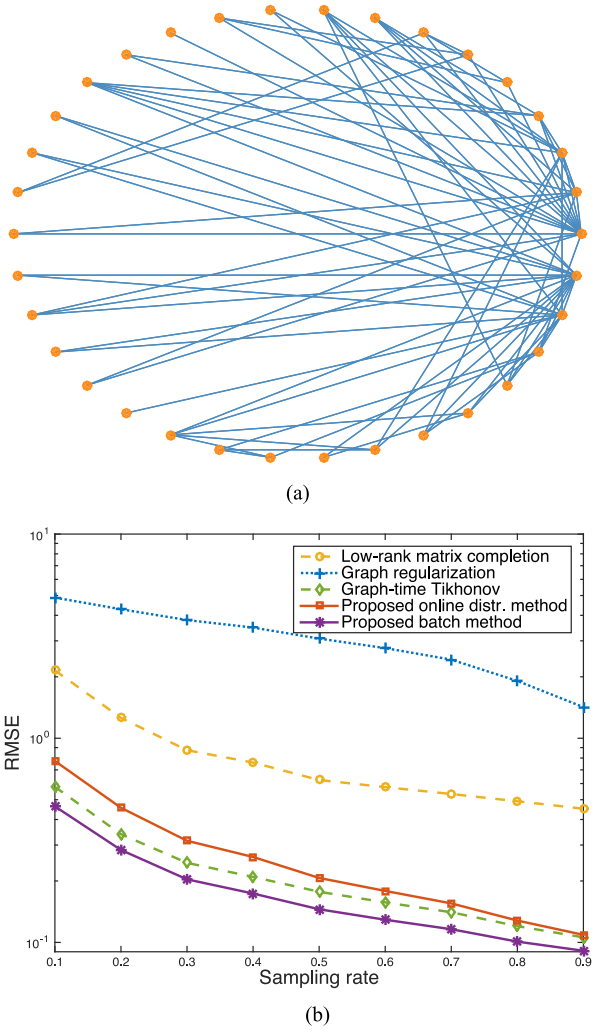


Fig. 6. (a) The graph structure of the social network of Zachary's karate club. (b) The reconstruction error under different sampling rate. The RMSE of each method is an average of 50 tests.

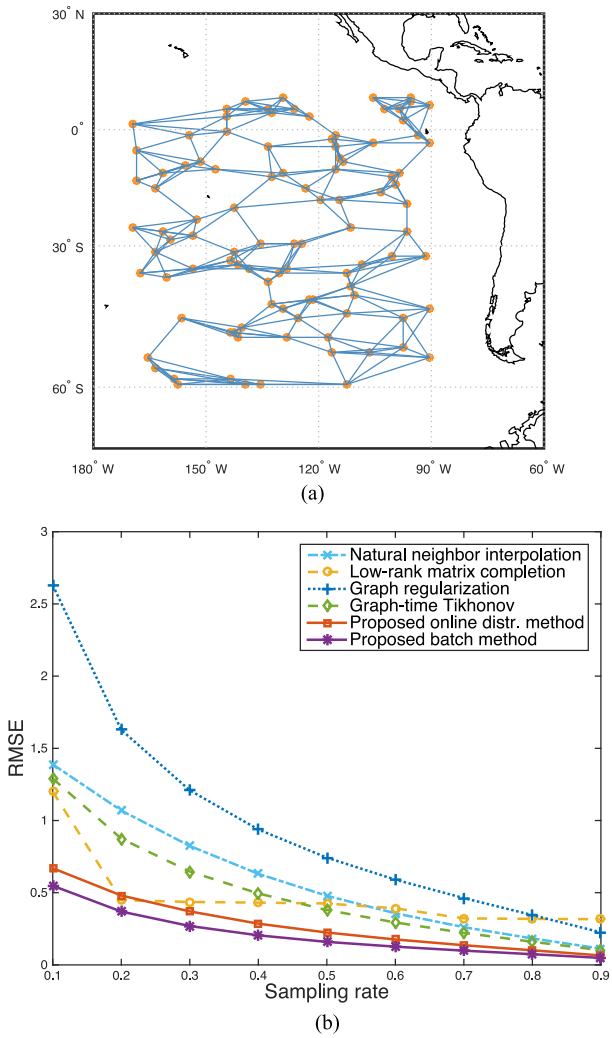


Fig. 7. The sea surface temperature dataset. (a) The graph constructed by 5-nearest neighbors algorithm. (b) The reconstruction error under different sampling rate. The RMSE of each method is an average of 50 tests.

formance of the proposed online distributed method is slightly worse than the graph-time Tikhonov method. We can draw a similar conclusion for the RMSE versus the smoothness level of temporal difference signal from Fig. 5(b). The only difference is that the performance of the proposed online distributed method is better than the graph-time Tikhonov method when the smoothness level of the temporal difference signal is larger than 1.

B. The Social Network of Zachary's Karate Club

The graph used in this experiment is a real-world social network of Zachary's karate club [55], which consists of 34 vertices representing members of the club and 78 undirected edges denoting friendships among members, shown in Fig. 6(a). We generate a 34×200 time-varying signal residing on the graph in a similar way to the description in Section VI-A.

When the sampling rate is 38%, the RMSEs of low-rank matrix completion, graph regularization, graph-time Tikhonov, the proposed online distributed method and the proposed batch method are 0.61, 11.81, 0.21, 0.26 and 0.17, respectively. The performance of all the methods under different sampling rate is displayed in Fig. 6(b). One can read from the results that the performance of the proposed batch method is better than all the comparison methods, and the performance of the proposed online distributed method is slightly worse than the graph-time Tikhonov method.

C. The Sea Surface Temperature

The sea surface temperature dataset [49] is published by the Earth System Research Laboratory. It is collected monthly from 1870 to 2014, and the spatial resolution is 1° latitude $\times$ 1° longitude. We randomly select 100 points on the Pacific Ocean from 170° west to 90° west, and 60° south to 10° north, within a time period of 600 months. The selected data range from -0.01 °C to 30.72 °C, and the mean is 19.15 °C.

A graph is constructed by 5-nearest neighbors algorithm, shown in Fig. 7(a). The horizontal and vertical coordinates are longitude and latitude respectively, and the black curves are the coastlines. When the sampling rate is 40%, the RMSEs of natural neighbor interpolation, low-rank matrix completion, graph regularization, graph-time Tikhonov, the proposed online

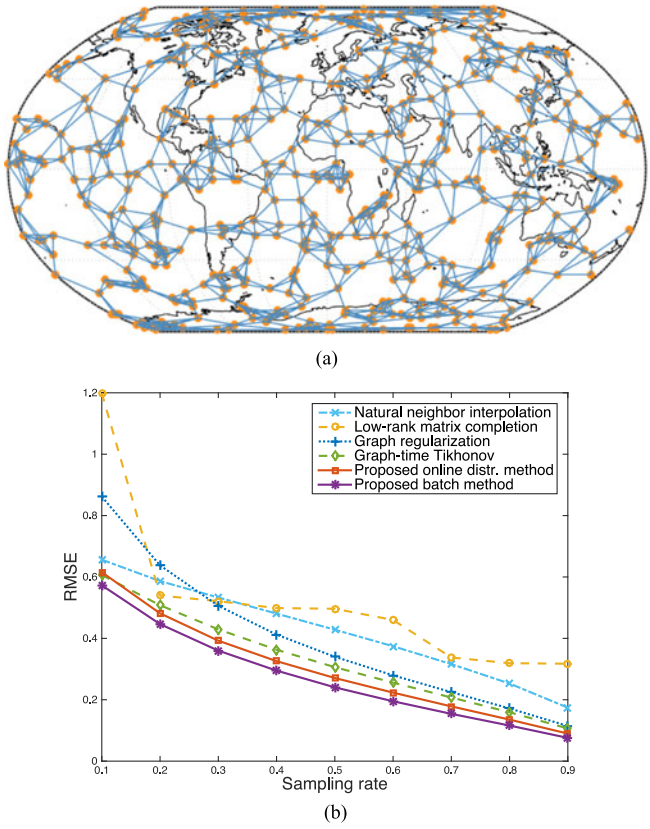


Fig. 8. The global sea-level pressure dataset. (a) The graph constructed by 5-nearest neighbors algorithm. (b) The reconstruction error under different sampling rate. The RMSE of each method is an average of 50 tests.

distributed method and the proposed batch method are 0.63, 0.43, 0.94, 0.49, 0.29 and 0.20, respectively. The performance of all the methods under different sampling rate is shown in Fig. 7(b). As can be seen, the performance of the proposed methods is better than all the comparison methods. There are two main reasons for the improvement of the proposed methods. Firstly, we construct a graph to describe the relation among the temperatures on the sea surface, which exploits the spatial correlation adequately. Secondly, we utilize the smoothness of the temporal difference signal as a prior rather than the smoothness of the signal itself, and the former is more reasonable for this real-world time-varying graph signal.

D. The Global Sea-Level Pressure

The global sea-level pressure dataset [50] is published by the Joint Institute for the Study of the Atmosphere and Ocean. It is collected from 1948 to 2010, with the spatial resolution being 2.5° latitude $\times$ 2.5° longitude, and the temporal resolution being five days. We randomly select 500 points worldwide over a time period of 600. The selected data range from 96.22 kPa to 110.06 kPa, and the average is 101.22 kPa.

A graph is constructed by 5-nearest neighbors algorithm, shown in Fig. 8(a). Note that 180° west and 180° east are coincident geographically, so the leftmost points are very close to the rightmost points in the figure, and there are edges between them that are not drawn in the figure.

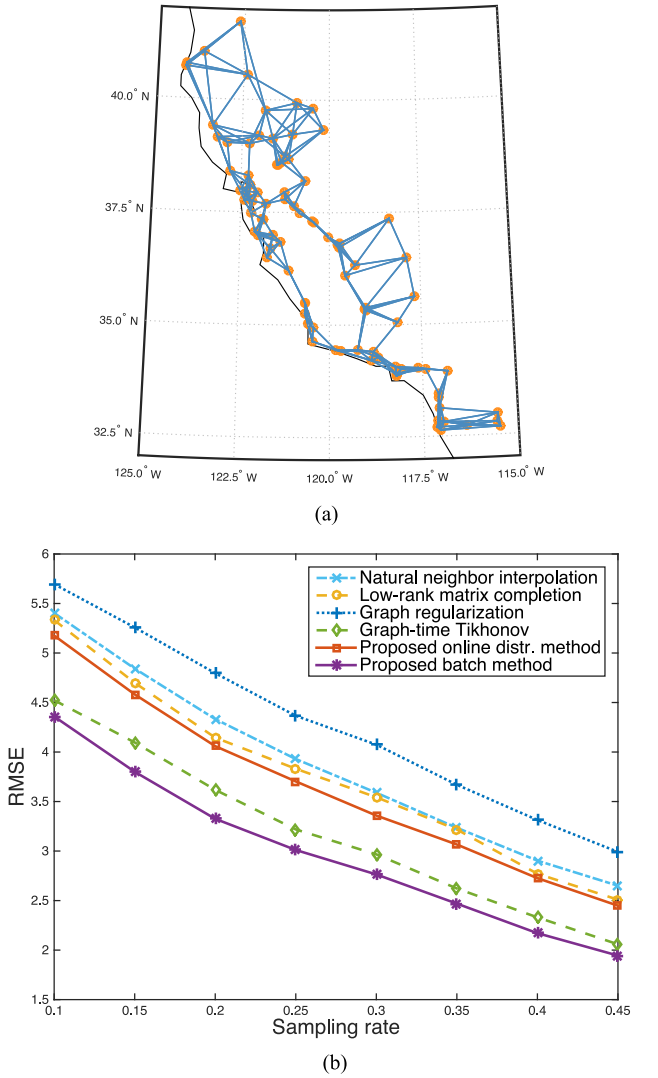


Fig. 9. The daily mean PM2.5 concentration dataset. (a) The graph constructed by 5-nearest neighbors algorithm. (b) The reconstruction error under different sampling rate. The RMSE of each method is an average of 50 tests.

When the sampling rate is 40%, the RMSEs of natural neighbor interpolation, low-rank matrix completion, graph regularization, graph-time Tikhonov, the proposed online distributed method and the proposed batch method are 0.48, 0.80, 0.41, 0.36, 0.33 and 0.29, respectively. Fig. 8(b) gives the reconstruction error of all the methods under different sampling rate. We can see that the performance of the proposed methods is better than the comparison methods.

E. The Daily Mean PM2.5 Concentration of California

The California daily mean PM2.5 concentration dataset [51] is published by the US Environmental Protection Agency. It is collected by 93 observation sites over 200 days starting from January 1, 2015, with the size of 93 $\times$ 200. Not all of these sites collected valid data everyday, and the percentages of valid data collected by these sites roughly range from 90 to 45. The valid data range from 0.1 μg/m^3 to 102.7 μg/m^3. A graph is constructed by 5-nearest neighbors algorithm, shown in Fig. 9(a).

In this experiment, the sampled vertices are among those with valid data, and the reconstruction error is evaluated among the valid data. When the sampling rate is 40%, the RMSEs of natural neighbor interpolation, low-rank matrix completion, graph regularization, graph-time Tikhonov, the proposed online distributed method and the proposed batch method are 2.87, 2.80, 3.30, 2.31, 2.71 and 2.18, respectively. Fig. 9(b) gives the reconstruction error of all the methods under different sampling rate. These results show that the proposed batch method achieves better performance than all the comparison methods.

VII. Conclusions

This paper studies the reconstruction of time-varying graph signals. Unlike the existing works which assume that the signal is smooth on graph, we find that the temporal difference of a time-varying signal exhibits better smoothness than the signal itself in many cases. Accordingly, a new batch reconstruction method is proposed by exploiting the smoothness of the temporal difference of graph signals, and an online distributed method is further developed that allows the reconstruction on each vertex to be only related to its neighbors and its own previous recovered value. The excellent performance of the proposed methods has been demonstrated by various experiments. In the future, we will further study the applications of smoothness of temporal difference signals, and may combine it with other properties of signals, such as low rank. Besides, it is also interesting to consider the situation where both the signal and the graph are time-varying.

References

[1] 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.

[2] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs," *IEEE Trans. Signal Process*, vol. 61, no. 7, pp. 1644–1656, Apr. 2013.

[3] N. Thanou, "Graph signal processing: Sparse representation and applications," Ph.D. dissertation, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, 2016.

[4] X. Zhu and M. Rabbat, "Graph spectral compressed sensing for sensor networks," in *Proc. 37th IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2012, pp. 2865–2868.

[5] X. Shi, H. Feng, M. Zhai, and T. Yang, "Infinite impulse response graph filters in wireless sensor networks," *IEEE Signal Process. Lett.*, vol. 22, no. 8, pp. 1113–1117, Aug. 2015.

[6] S. K. Narang, A. Gadde, and A. Ortega, "Signal processing techniques for interpolation in graph structured data," in *Proc. 38th IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 5445–5449.

[7] A. Gadde, A. Anis, and A. Ortega, "Active semi-supervised learning using sampling theory for graph signals," in *Proc. 20th ACM Int. Conf. Knowl. Discovery Data Mining*, 2014, pp. 492–501.

[8] A. Gadde and A. Ortega, "A probabilistic interpretation of sampling theory of graph signals," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2015, pp. 3257–3261.

[9] O. Lezoray, A. Elmoataz, and S. Bougleux, "Graph regularization for color image processing," *Comput. Vis. Image Understanding*, vol. 107, no. 12, pp. 38–55, 2007.

[10] S. K. Narang, Y. H. Chao, and A. Ortega, "Graph-wavelet filterbanks for edge-aware image processing," in *Proc. IEEE Stat. Signal Process. Workshop*, 2012, pp. 141–144.

[11] A. Y. Mutlu, E. Bernat, and S. Aviyente, "A signal-processing-based approach to time-varying graph analysis for dynamic brain network identification," *Comput. Math. Methods Med.*, vol. 2012, 2012, Art. no. 451516.

[12] N. Leonardi and D. V. D. Van, "Tight wavelet frames on multislice graphs," *IEEE Trans. Signal Process.*, vol. 61, no. 13, pp. 3357–3367, Jul. 2013.

[13] W. H. Kim *et al.* "Multi-resolutional brain network filtering and analysis via wavelets on non-Euclidean space," *Med. Image Comput. Comput. Assist. Interv.*, vol. 16, no. Pt3, pp. 643–651, 2013.

[14] S. Chen, A. Sandryhaila, J. M. F. Moura, and J. Kovacevic, "Signal denoising on graphs via graph filtering," in *Proc. 2014 IEEE Global Conf. Signal Inf. Process.* IEEE, 2014, pp. 872–876.

[15] X. Dong, D. Thanou, P. Frossard, and P. Vandergheynst, "Laplacian matrix learning for smooth graph signal representation," in *Proc. 2015 IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2015, pp. 3736–3740.

[16] J. Mei and J. M. F. Moura, "Signal processing on graphs: Estimating the structure of a graph," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2015, pp. 5495–5499.

[17] S. Chen, A. Sandryhaila, J. M. F. Moura, and J. Kovacevic, "Adaptive graph filtering: Multiresolution classification on graphs," in *Proc. 2013 IEEE Global Conf. Signal Inf. Process.*, 2013, pp. 427–430.

[18] E. Isufi, A. Loukas, A. Simonetto, and G. Leus, "Distributed time-varying graph filtering," arXiv:1602.04436, 2016.

[19] X. Zhu and M. Rabbat, "Approximating signals supported on graphs," in *Proc. 37th IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2012, pp. 3921–3924.

[20] D. Thanou, P. A. Chou, and P. Frossard, "Graph-based compression of dynamic 3D point cloud sequences," *IEEE Trans. Image Process.*, vol. 25, no. 4, pp. 1765–1778, Apr. 2016.

[21] P. Liu, X. Wang, and Y. Gu, "Coarsening graph signal with spectral invariance," in *Proc. 39th IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2014, pp. 1075–1079.

[22] P. Liu, X. Wang, and Y. Gu, "Graph signal coarsening: Dimensionality reduction in irregular domain," in *Proc. 2014 IEEE Global Conf. Signal Inf. Process.*, 2014, pp. 798–802.

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

[24] A. G. Marques, S. Segarra, G. Leus, and A. Ribeiro, "Stationary graph processes and spectral estimation," arXiv:1603.04667, 2016.

[25] N. Perraudin and P. Vandergheynst, "Stationary signal processing on graphs," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3462–3477, Jul. 2017.

[26] I. Pesenson, "Variational splines and Paley–Wiener spaces on combinatorial graphs," *Constructive Approx.*, vol. 29, no. 1, pp. 1–21, 2009.

[27] I. Z. Pesenson and M. Z. Pesenson, "Sampling, filtering and sparse approximations on combinatorial graphs," *J. Fourier Anal. Appl.*, vol. 16, no. 6, pp. 921–942, 2010.

[28] S. Chen, R. Varma, A. Sandryhaila, and J. Kovačević, "Discrete signal processing on graphs: Sampling theory," *IEEE Trans. Signal Process.*, vol. 63, no. 24, pp. 6510–6523, Dec. 2015.

[29] T. M. Smith and R. W. Reynolds, "Improved extended reconstruction of SST (1854-1997)," *J. Climate*, vol. 17, no. 12, pp. 2466–2477, 2004.

[30] L. Kong, M. Xia, X. Y. Liu, M. Wu, and X. Liu, "Data loss and reconstruction in sensor networks," in *Proc. IEEE INFOCOM*, 2013, pp. 1654–1662.

[31] I. Pesenson, "Sampling in Paley-Wiener spaces on combinatorial graphs," *Trans. Amer. Math. Soc.*, vol. 360, no. 10, pp. 5603–5627, 2008.

[32] S. K. Narang, A. Gadde, E. Sanou, and A. Ortega, "Localized iterative methods for interpolation in graph structured data," in *Proc. 1st IEEE Global Conf. Signal Inf. Process.*, 2013, pp. 491–494.

[33] X. Wang, P. Liu, and Y. Gu, "Local-set-based graph signal reconstruction," *IEEE Trans. Signal Process.*, vol. 63, no. 9, pp. 2432–2444, May 2015.

[34] A. G. Marques, S. Segarra, G. Leus, and A. Ribeiro, "Sampling of graph signals with successive local aggregations," *IEEE Trans. Signal Process.*, vol. 64, no. 7, pp. 1832–1843, Apr. 2015.

[35] X. Wang, J. Chen, and Y. Gu, "Local measurement and reconstruction for noisy bandlimited graph signals," *Signal Process.*, vol. 129, no. 5, pp. 119–129, 2016.

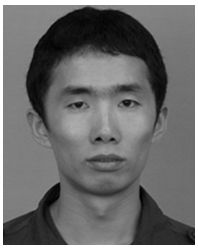
[36] M. Belkin, I. Matveeva, and P. Niyogi, "Regularization and semi-supervised learning on large graphs," in *Proc. Int. Conf. Comput. Learn. Theory*. Berlin, Germany: Springer-Verlag, 2004, pp. 624–638.

[37] S. Bougleux, A. Elmoataz, and M. Melkemi, "Discrete regularization on weighted graphs for image and mesh filtering," in *Proc. Int. Conf. Scale Space Variational Methods Comput. Vis.* Berlin, Germany: Springer-Verlag, 2007, pp. 128–139.

[38] D. Romero, M. Ma, and G. B. Giannakis, "Kernel-based reconstruction of graph signals," *IEEE Trans. Signal Process.*, vol. 65, no. 3, pp. 764–778, Feb. 2017.

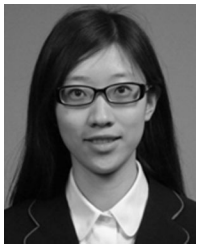
[39] X. Wang, M. Wang, and Y. Gu, "A distributed tracking algorithm for reconstruction of graph signals," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 4, pp. 728–740, Jun. 2015.

[40] S. Chen, A. Sandryhaila, J. M. F. Moura, and J. Kovacevic, "Signal recovery on graphs: Variation minimization," *IEEE Trans. Signal Process.*, vol. 63, no. 17, pp. 4609–4624, Sep. 2015.
[41] A. Sandryhaila and J. M. F. Moura, "Big data analysis with signal processing on graphs: Representation and processing of massive data sets with irregular structure," *IEEE Signal Process. Mag.*, vol. 31, no. 5, pp. 80–90, Sep. 2014.
[42] P. Valdivia, F. Dias, F. Petronetto, C. T. Silva, and L. G. Nonato, "Wavelet-based visualization of time-varying data on graphs," in *Proc. 2015 IEEE Conf. Visual Analytics Sci. Technol.*, Oct. 2015, pp. 1–8.
[43] A. Loukas and D. Foucard, "Frequency analysis of temporal graph signals," arXiv:1602.04434, 2016.
[44] E. Isufi, A. Loukas, A. Simonetto, and G. Leus, "Autoregressive moving average graph filtering," *IEEE Trans. Signal Process.*, vol. 65, no. 2, pp. 274–288, Jan. 2017.
[45] N. Perraudin, A. Loukas, F. Grassi, and P. Vandergheynst, "Towards stationary time-vertex signal processing," in *Proc. 2017 IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2017, pp. 3914–3918.
[46] A. Loukas and N. Perraudin, "Stationary time-vertex signal processing," arXiv:1611.00255, 2016.
[47] F. R. K. Chung, *Spectral Graph Theory*. Providence, RI, USA: AMS, 1997.
[48] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *Appl. Comput. Harmon. Anal.*, vol. 30, no. 2, pp. 129–150, 2011.
[49] "Sea surface temperature (SST) v2," Dec. 18, 2015. [Online]. Available: http://www.esrl.noaa.gov/psd/data/gridded/data.noaa.oisst.v2.html
[50] "Sea-level pressure, 1948–2010," May 25, 2016. http://research.jisao.washington.edu/data_sets/reanalysis/
[51] "Air quality data," May 19, 2016. https://www.epa.gov/outdoor-air-quality-data
[52] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA, USA: Athena Scientific, 1999, pp. 130–145.
[53] S. Robert, "A brief description of natural neighbour interpolation," *Interpreting Multivariate Data*, vol. 21, pp. 21–36, 1981.
[54] S. Ma, D. Goldfarb, and L. Chen, "Fixed point and Bregman iterative methods for matrix rank minimization," *Math. Program.*, vol. 128, no. 1/2, pp. 321–353, 2011.
[55] W. W. Zachary, "An information flow model for conflict and fission in small groups," *J. Anthropological Res.*, vol. 33, no. 4, pp. 452–473, 1977.



Kai Qiu received the B.E. degree in electronic engineering, in 2009, from Tsinghua University, Beijing, China, where he is currently working toward the Ph.D. degree in electronic engineering. His research interests include magnetic resonance imaging and graph signal processing.

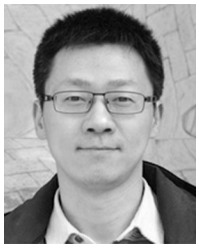
Xianghui Mao received the B.E. degree in electronic engineering from Tsinghua University, Beijing, China, in 2014. She is currently working toward the Ph.D. degree in electronic engineering. Her recent research interests include optimization related topics and graph signal processing.



Xinyue Shen received the B.E. (Hons.) degree in electronic engineering from Tsinghua University, Beijing, China, in 2013. She is currently working toward the Ph.D. degree in electronic engineering. Her research interests include structured data recovery and nonconvex optimization.

Xiaohan Wang received the Ph.D. degree from Tsinghua University, Beijing, China, in 2016. He is currently a Researcher in Huawei Technologies Co., Ltd., Shanghai, China. His research interests include signal processing and complex networks.

Tiejian Li (M'13) received the Ph.D. degree from Tsinghua University, Beijing, China, in 2008. He is currently an Associate Professor at Tsinghua University and an Adjunct Professor at Qinghai University, Xining, China. His research interests include the fields of river basin modeling, parallel computing in hydrological models, and data mining algorithms in hydroscience.



Yuantao Gu (S'02– M'03– SM'17) received the B.E. degree from Xi'an Jiaotong University, Xi'an, China, in 1998, and the Ph.D. (Hons.) degree from Tsinghua University, Beijing, China, in 2003, both in electronic engineering.

He joined the faculty of Tsinghua in 2003 and is currently an Associate Professor in the Department of Electronic Engineering. He was a Visiting Scientist at Microsoft Research Asia, Beijing (during December 2005 to February 2006), at the Research Laboratory of Electronics at Massachusetts Institute of Technology (MIT), Cambridge, MA (during August 2012 to August 2013), and in the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI (during September to October 2015). His research interests include adaptive signal processing, sparse signal recovery, graph signal processing, and related topics in wireless communications and information networks. He received the Best Paper Award of IEEE Global Conference on Signal and Information Processing in 2015, the Award for Best Presentation of Journal Paper of the IEEE International Conference on Signal and Information Processing in 2015, and Zhang Si-Ying (the 29th Chinese Control and Decision Conference) Outstanding Youth Paper Award (with his Ph.D. student) in 2017. He has been an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING since February 2015, a Handling Editor of the *EURASIP Digital Signal Processing* since February 2015, and an Elected Member of the IEEE Signal Processing Theory and Methods Technical Committee since January 2017.