Evolution backcasting of edge flows from partial observations using simplicial vector autoregressive models

Money, Rohan; Krishnan, Joshin; Beferull-Lozano, Baltasar; Isufi, Elvin

DOI
10.1109/ICASSP48485.2024.10448180

Publication date
2024

Document Version
Final published version

Published in
2024 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2024 - Proceedings

Citation (APA)
Money, R., Krishnan, J., Beferull-Lozano, B., & Isufi, E. (2024). Evolution backcasting of edge flows from partial observations using simplicial vector autoregressive models. In *2024 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2024 - Proceedings* (pp. 9516-9520). (ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings). IEEE. https://doi.org/10.1109/ICASSP48485.2024.10448180

ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 979-8-3503-4485-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICASSP48485.2024.10448180

EVOLUTION BACKCASTING OF EDGE FLOWS FROM PARTIAL OBSERVATIONS USING SIMPLICIAL VECTOR AUTOREGRESSIVE MODELS

Rohan Money[⋆], *Joshin Krishnan*[⋆], *Baltasar Beferull-Lozano*[⋆†] *and Elvin Isufi*[‡].

ABSTRACT

This paper proposes a novel algorithm to retroactively compute the evolution of edge signals from a given sequence of partial observations from topological structures, a concept referred to as *evolution backcasting*. Our backcasting algorithm exploits the spatio-temporal dependencies present in the real-world edge signals using the simplicial vector autoregressive (S-VAR) model. The proposed algorithm jointly estimates the S-VAR filter coefficients and recovers missing data from the partial observations. Subsequently, the algorithm capitalizes on the learned S-VAR model and the reconstructed signals to execute the backcasting of edge signal evolution. Using traffic and water distribution networks as case studies, we showcase the superior capabilities of our algorithm compared with baseline alternatives.

Index Terms— Simplicial convolution, simplicial vector autoregressive model, Hodge Laplacians.

1. INTRODUCTION

Signals defined on edges [1], commonly referred to as edge flows, play a crucial role in a multitude of real-world topological structures, including transportation, epidemic, and water distribution networks. Among these networks, a crucial endeavour involves retroactively computing the evolution of edge signals–an approach we refer to as *evolution backcasting*. This process involves estimating the evolution of edge signals based on a given sequence of observed edge signal data. Evolution backcasting can provide invaluable insights into the roots of epidemic spread, traffic congestion, and beyond.

Time series analysis encompasses a vast literature that exploits the spatio-temporal dependencies found in real-world signals to accomplish various signal processing tasks, such as time series forecasting [2, 3], change point identification [4], and denoising [5, 6]. Guided by these techniques, this paper explores the prospect of leveraging spatio-temporal dependencies to achieve evolution backcasting of edge signals. In this context, the vector autoregressive (VAR) model stands out as it is commonly utilized for modelling spatial and temporal dependencies among signals, largely owing to its mathematical traceability [7]. While the above works focus on the vertex domain signals, the VAR models can also be applied to model edge signals from real-world topological structures since the edges also exhibit spatio-temporal dependencies in accordance with the physical laws governing the system. However, the VAR model lacks the capacity to incorporate the structural information provided by a given topology, leading to an increased number of parameters. This makes VAR implementation challenging in certain conditions, such as when dealing with a large number of time series, limited data, or excessively noisy data. While [8] strives to leverage structural information for edge data imputation, it does so by incorporating structural aspects solely through regularization in the learning process, which does not reduce the high parameter count of the VAR model and can lead to overfitting issues with limited or noisy data. Thus, relying on a conventional VAR model-based approach to process the edge signals falls short.

Recent advancements in topological signal processing facilitate structure-aware data processing techniques using simplicial complexes [9–14], which is a potential tool to perform evolution backcasting. One noteworthy model among these is the simplicial vector autoregressive (S-VAR) model [14], which combines the features of both the VAR model and the simplicial complex to capture spatio-temporal dependencies in edge signals for forecasting. S-VAR model enables the infusion of structural insights into the process as well as reduces the model complexity, presenting substantial benefits in contrast to conventional VAR models. Nevertheless, the potential benefits provided by the simplicial complex formulation have not been harnessed when addressing critical tasks such as evolution backcasting, missing data imputation, change point detection, and more. Building upon this research gap, we propose an algorithm to backcast the evolution of edge signals from a given sequence of observations using S-VAR model.

Our algorithm comprises two stages: $i)$ it learns the S-VAR filter and estimates missing edge signals from partially observed data using a block coordinate descent technique; and $ii)$ leveraging the learned filter and the reconstructed signals, the algorithm backcasts the evolution of edge flows prior to the partial observations. Backcasting becomes more challenging when we only possess partial observations, a common occurrence in practical scenarios where data accessibility can be limited due to sensor and communication failures or simply the impracticality of placing sensors in every location. However, the rich structural information coupled with the strategic regularization, rooted in the simplicial complex formulation [15], enables our proposed algorithm to address this challenge proficiently.

2. PRELIMINARIES

2.1. Vector Autoregressive Models

For a multivariate time series $\{y_n[t]\}_{n=1}^{N}$ with $t = 1 \dots T$, a P-th order linear VAR model is expressed as

$$y_n[t] = \sum_{n'=1}^{N} \sum_{p=1}^{P} a_{n,n'}^{(p)} y_{n'}[t-p] + u_n[t], \tag{1}$$

where $y_n[t]$ is the value of the time-series at time t measured at a given node $1 \leq n \leq N$ and $a_{n,n'}^{(p)}$ captures the influence of the p-lagged data at node n' on the node n, and $u_n[t]$ is the process noise.

*SIGIPRO Department, Simula Metropolitan Center for Digital Engineering, Oslo, Norway; †WISENET Center, Department of ICT, University of Agder, Grimstad, Norway; ‡Intelligent Systems Department, Delft University of Technology, Delft, The Netherlands. rohan@simula.no; joshin@simula.no; baltasar.beferull@uia.no; baltasar@simula.no; e.isufi-1@tudelft.nl. The study was supported by the IKTPLUSS DISCO grant 338740/AAU and the TU Delft AI Labs programme.

2.2. Simplicial Complex and Simplicial Signals

Consider a set of vertices $\mathcal{V} = \{1, \ldots, N\}$. We can define a $k-$simplex, $\mathcal{S}^k$, as a subset of $\mathcal{V}$ encompassing $k+1$ unique elements. A simplicial complex (SC) $\mathcal{X}^K$of order K is a collection of simplices in which at least one K-simplex exists. Furthermore, a simplex $\mathcal{S}^k$ is part of $\mathcal{X}^K$ if, and only if, all its subsets are also present within $\mathcal{X}^K$ [16]. An example of $\mathcal{X}^2$ can be seen in Fig. 1.

In a SC, we can use incidence matrices and Hodge Laplacians to express adjacencies between the different simplices [16]. Letting N_k denote the number of $k-$simplices in a SC, the incidence matrix, $\mathbf{B}_k \in \mathbb{R}^{N_{k-1} \times N_k}$, is structured with $(k-1)-$simplices in its rows and $k-$simplices in its columns, thus reflecting the adjacencies between these elements. For instance, $\mathbf{B}_1$ stands for the node-to-edge incidence, whereas $\mathbf{B}_2$ stands for the edge-to-triangle incidence. These incidence matrices adhere to the boundary condition represented as $\mathbf{B}_1\mathbf{B}_2 = \mathbf{0}$. The Hodge Laplacians representing the structure of $\mathcal{X}^2$ are given by

$$\begin{aligned}
&\mathbf{L}_0 = \mathbf{B}_1\mathbf{B}_1^\top, \\
&\mathbf{L}_1 = \mathbf{L}_{1,\ell} + \mathbf{L}_{1,u} := \mathbf{B}_1^\top\mathbf{B}_1 + \mathbf{B}_2\mathbf{B}_2^\top, \\
&\mathbf{L}_2 = \mathbf{B}_2^\top\mathbf{B}_2.
\end{aligned} \tag{2}$$

Here, $\mathbf{L}_0$ represents the well-known graph Laplacian [17], which defines adjacencies between vertices by considering shared edges. The Hodge Laplacian $\mathbf{L}_1$ depicts adjacencies between edges based on shared vertices via the lower-Laplacian $\mathbf{L}_{1,\ell} = \mathbf{B}_1^\top\mathbf{B}_1$ and shared triangles via the upper-Laplacian $\mathbf{L}_{1,u} = \mathbf{B}_2\mathbf{B}_2^\top$. Similarly, $\mathbf{L}_2$ captures the proximity between triangles through common edges.

While the structure of a SC captures topological adjacencies between edges, we are interested in processing signals defined on them. Specifically, the simplicial signals are conceptualized as functions that map any $k-$simplex to the set of real numbers. An edge flow signal is denoted by $\mathbf{f} = [f_1, \ldots, f_{N_1}]^\top \in \mathbb{R}^{N_1}$, where f_e represents the flow on edge $e = (m, n)$ in $\mathcal{S}^1$. Similarly, signals defined on vertices and triangles are represented as $\mathbf{v} \in \mathbb{R}^{N_0}$ and $\boldsymbol{\tau} \in \mathbb{R}^{N_2}$, respectively, which are illustrated in Fig. 1. The inherent relationships within a SC are reflected as dependencies between the signals defined over it. The main aim is to harness these dependencies to process signals effectively [18].

2.3. Convolution in the Simplex

Simplicial signals can be refined using simplicial convolutional filters. For edge signals, this process is represented by [19]

$$\mathbf{f}_s = \underbrace{\sum_{k=0}^{K_\ell} \beta_k^\ell \mathbf{L}_{1,\ell}^k \, \mathbf{f}}_{\mathbf{A}(\mathbf{L}_{1,\ell})} + \underbrace{\sum_{k=1}^{K_u} \beta_k^u \mathbf{L}_{1,u}^k \, \mathbf{f}}_{\mathbf{B}(\mathbf{L}_{1,u})}. \tag{3}$$

Here, filter coefficients $\{\beta_k^\ell\}_{k=1}^{K_\ell}$ and $\{\beta_k^u\}_{k=1}^{K_u}$ weigh the edge signal $\mathbf{f}$ subjected to shifts by $\mathbf{L}_{1,\ell}$ for a range up to K_ℓ hops and by $\mathbf{L}_{1,u}$ up to K_u hops. The encompassed number of parameters in (3) $K := K_\ell + K_u + 1$. By defining the *simplicial convolutional filtering* matrix $\mathbf{H}(\mathbf{L}_1) := \mathbf{A}(\mathbf{L}_{1,\ell}) + \mathbf{B}(\mathbf{L}_{1,u})$, we can compactly write (3) as $\mathbf{f}_s = \mathbf{H}(\mathbf{L}_1)\mathbf{f}$. The convolutional filter is uniformly applied across edges, regardless of their specific labelling or flow direction.

2.4. Simplicial Vector Autoregressive Model

The S-VAR model of order P for time-varying edge flow $\mathbf{f}_t$ is given by [14]

$$\mathbf{f}[t] = \sum_{p=1}^{P} \mathbf{H}_p(\mathbf{L}_1)\mathbf{f}[t-p] + \boldsymbol{\varepsilon}[t]. \tag{4}$$

In (4), $\mathbf{H}_p(\mathbf{L}_1)$ encompasses simplicial convolutional filters characterized by parameters $\{\beta_{p,k}^\ell\}_{k=0}^{K_\ell}$ and $\{\beta_{p,k}^u\}_{k=1}^{K_u}$ (cf. (3)). Specifically, the filter $\mathbf{H}_p(\mathbf{L}_1)$ delineates the spatial dependencies of the process $\mathbf{f}_{t-p}$, while the VAR regression highlights the temporal dependencies up to a time lag P. The variable $\boldsymbol{\varepsilon}_t$ in (4) represents the model uncertainty and is assumed to be a zero-mean Gaussian.

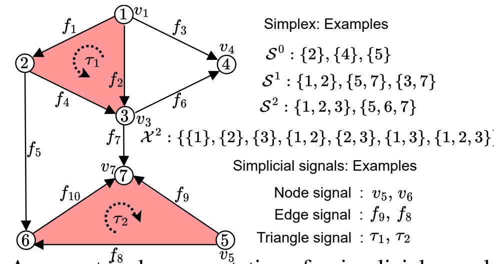

Fig. 1: A geometrical representation of a simplicial complex $\mathcal{X}^2$ and associated simplicial signals. The arrows marking the edges and triangular arcs symbolize the reference directions for edges and triangle signals, respectively, and their orientations are chosen arbitrarily.

In model (4), the simplicial convolution filters allow us to incorporate the structural information in modelling the process along with the data. The model proficiently capitalizes on the structure-sensitive proximities of the topological structure, thereby requiring fewer parameters compared to the traditional VAR model. Remarkably, the S-VAR model demands only KP parameters, making it independent of the process count N_1. In contrast, the traditional VAR approach requires many more parameters on the order of $N_1^2 P$. This structural efficiency allows S-VAR model to learn well in limited data scenarios. To reformulate (4) in a more amenable form, let us define $\boldsymbol{\beta}_p := [\beta_{p,0}^\ell, \ldots, \beta_{p,K_\ell}^\ell, \beta_{p,1}^u, \ldots, \beta_{p,K_u}^u] \in \mathbb{R}^K$ and $\tilde{\mathbf{F}}[t-p] := [\mathbf{L}_{1,\ell}^0\mathbf{f}[t-p], \mathbf{L}_{1,\ell}^1\mathbf{f}[t-p], \ldots, \mathbf{L}_{1,\ell}^{K_\ell}\mathbf{f}[t-p], \mathbf{L}_{1,u}^1\mathbf{f}[t-p], \ldots, \mathbf{L}_{1,u}^{K_u} \mathbf{f}[t-p]]$, where, the matrix $\tilde{\mathbf{F}}[t-p] \in \mathbb{R}^{N_1 \times K}$ collects the shifted versions of the edge signal $\mathbf{f}[t-p]$. Let us also define $\mathbf{F}[t] := [\tilde{\mathbf{F}}[t-1], \ldots, \tilde{\mathbf{F}}[t-P]] \in \mathbb{R}^{N_1 \times KP}$ and $\boldsymbol{\beta} := [\boldsymbol{\beta}_1^\top, \ldots, \boldsymbol{\beta}_P^\top]^\top \in \mathbb{R}^{KP}$, which allow us to write, the S-VAR model (4) in compact form as

$$\mathbf{f}[t] = \mathbf{F}[t]\boldsymbol{\beta} + \boldsymbol{\varepsilon}[t]. \tag{5}$$

3. PROBLEM FORMULATION

Let $\mathbf{f}[t] \in \mathbf{R}^{N_1}$ be an edge process evolving over a topological structure, with a known topology $\mathcal{X}^2$ over time indices $t = 1, 2, \cdots, T_f$. We have access to only partial observations of this process during the period $[T_s, T_f]$ where $T_s >> 1$, while no observations are accessible for the time interval $[1, T_s]$. The partial observation can be modelled as

$$\mathbf{f}_o[t] = \mathbf{M}[t](\mathbf{f}[t] + \mathbf{e}[t]), \text{for } t = T_s, T_s + 1, \cdots, T_f, \tag{6}$$

where $\mathbf{e}[t]$ the observation noise and $\mathbf{M}[t]$ is the diagonal masking matrix with $\mathbf{M}(n,n)[t] = 0/1$ if n-th flow is missing/present. Our goal is to recover the complete signals $\{\mathbf{f}[t]\}_{t=1}^{T_f}$ by learning an S-VAR model from the partial observations in the interval $[T_s, T_f]$ (see Fig. 2). As opposed to earlier works [20–22], our algorithm involves conducting evolution backcasting during the time interval $t = 1$ to $T_s - 1$, for which no observed data is accessible.

4. PROPOSED SOLUTION

The solution we propose operates in two steps, as outlined in Fig. 2. Initially, we employ the block coordinate descent method to jointly estimate missing signals in the interval $[T_s, T_f]$ and S-VAR coefficients. Subsequently, we leverage the learned S-VAR model and reconstructed signal to backcast the signal evolution recursively.

4.1. Joint Estimation of Signal and S-VAR Coefficients

An optimization problem to jointly estimate the filter coefficients $\boldsymbol{\beta}$ and edge signals $\{\mathbf{f}[t]\}_{t=T_s}^{T_f}$ can be formulated as[1]

$$\hat{\boldsymbol{\beta}}, \{\hat{\mathbf{f}}[t]\}_{t=T_s}^{T_f} = \arg\min_{\boldsymbol{\beta},\mathbf{f}_t} \frac{1}{2}\sum_{t=T_s}^{T_f} \|\mathbf{f}[t] - \mathbf{F}[t]\boldsymbol{\beta}\|_2^2 + \|\mathbf{f}_o[t] - \mathbf{M}\mathbf{f}[t]\|_2^2 + \Omega(\boldsymbol{\beta}, \mathbf{L}_1), \qquad (7)$$

where the first term fits the model with data and the second term makes the signal reconstruction consistent with observation. The third term $\Omega(\boldsymbol{\beta}, \mathbf{L}_1)$ is a regularizer that avoids overfitting to the observed values and biases the flow estimate based on the existing knowledge about the system. The regularizer consists of the addition of three terms: $i)$ $\hat{\mathbf{f}}[t]\mathbf{L}_{1,\ell}\hat{\mathbf{f}}[t]$, where $\hat{\mathbf{f}}[t] = \mathbf{F}[t]\boldsymbol{\beta}$, $ii)$ $\hat{\mathbf{f}}[t]\mathbf{L}_{1,u}\hat{\mathbf{f}}[t]$, and $iii)$ $\|\boldsymbol{\beta}\|_2^2$. Here, $i)$ and $ii)$ impose constraints based on the simplicial structure, e.g., $\hat{\mathbf{f}}[t]\mathbf{L}_{1l}\hat{\mathbf{f}}[t] = \|\mathbf{B}_1\hat{\mathbf{f}}[t]\|^2$ regularizes the divergence flows ($\mathbf{B}_1\hat{\mathbf{f}}[t]$), thereby enforcing the conservation of the flows at the nodes, and $\hat{\mathbf{f}}[t]\mathbf{L}_{1u}\hat{\mathbf{f}}[t] = \|\mathbf{B}_2^\top\hat{\mathbf{f}}[t]\|^2$ regularizes the cyclic flows ($\mathbf{B}_2^\top\hat{\mathbf{f}}[t]$) [11, 19].

The optimization problem can be split into $a)$ the parameter fitting part with respect to $\boldsymbol{\beta}$ and $b)$ the signal reconstruction part with respect to $\mathbf{f}[t]$. Then, a block coordinate descent method can solve the required optimization problem, alternating between the filter coefficient estimation and signal reconstruction. As both the subproblems are strongly convex, the block coordinate descent is ensured to converge to local minima.

Signal reconstruction from partial edge observations: The optimization problem (7) with respect to $\{\mathbf{f}[t]\}_{t=_s}^{T_f}$ can be decoupled across t:

$$\hat{\mathbf{f}}[t] = \arg\min_{f[t]} \|\mathbf{f}[t] - \mathbf{F}[t]\hat{\boldsymbol{\beta}}\|_2^2 + \frac{m}{2}\|\mathbf{f}_o - \mathbf{M}\mathbf{f}[t]\|_2^2 + \lambda_1\|\mathbf{B}_1\mathbf{f}[t]\|_2^2 + \lambda_2\|\mathbf{B}^\top\mathbf{f}[t]\|_2^2, \qquad (8)$$

where m, λ_1, and λ_2 are hyperparameters. The solution of (8) can be obtained in a closed-form as

$$\hat{\mathbf{f}}[t] = (\mathbf{I} + \mathbf{M}[t]^\top\mathbf{M}[t] + \lambda_1\mathbf{L}_{1,l} + \lambda_2\mathbf{L}_{1,u})^{-1}\mathbf{G}_1, \qquad (9)$$

where $\mathbf{G}_1 = \hat{\boldsymbol{\beta}}\mathbf{F}_t + \mathbf{M}(t)\mathbf{M}(t)^\top\mathbf{f}_o$ and $\hat{\boldsymbol{\beta}}$ is assumed to be initialized as $\mathbf{0} \in \mathbb{R}^{KP}$.

Optimization with respect to $\boldsymbol{\beta}$: The optimization problem (7) with respect to $\boldsymbol{\beta}$ is rewritten as

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \|\hat{\mathbf{f}} - \mathbf{F}\boldsymbol{\beta}\|_2^2 + \mu\|\boldsymbol{\beta}\|_2^2 \qquad (10)$$

where $\hat{\mathbf{f}} = [\hat{\mathbf{f}}[T_s]^\top, \hat{\mathbf{f}}[T_s+1]^\top, \ldots, \hat{\mathbf{f}}[T_f]^\top]^\top \in \mathbb{R}^{N_1(T_f - T_s + 1)}$, $\mathbf{F} = [\mathbf{F}[T_s]^\top, \mathbf{F}[T_s+1]^\top, \ldots, \mathbf{F}[T_f]^\top]^\top \in \mathbb{R}^{N_1(T_f - T_s+1)\times KP}$, and μ is a hyperparameter. The solution for (10) can be readily obtained in closed form as

$$\hat{\boldsymbol{\beta}} = (\mathbf{F}^\top\mathbf{F} + \mu\mathbf{I})^{-1}\mathbf{F}^\top\hat{\mathbf{f}}. \qquad (11)$$

4.2. Evolution Backcasting

Using the estimated S-VAR parameter $\hat{\boldsymbol{\beta}}$ and the reconstructed signals from partial observation $\{\hat{\mathbf{f}}[t]\}_{t=T_s}^{T_f}$, we perform a backcasting for the signals $\{\hat{\mathbf{f}}[t]\}_{t=1}^{T_s-1}$ in a recursive way. First, using equation (4) and the structure-aware regularizers, we formulate an optimization problem for estimating the signal at time $T-P$:

$$\hat{\mathbf{f}}[T-P] = \arg\min_{\mathbf{f}[T-P]} \left\|\hat{\mathbf{f}}[T] - \mathbf{H}_P(\mathbf{L}_1)\mathbf{f}[T-P] - \sum_{p=1}^{P-1}\mathbf{H}_p(\mathbf{L}_1)\hat{\mathbf{f}}[T-p]\right\|_2^2 + \Omega(\boldsymbol{\beta}, \mathbf{L}_1) \qquad (12)$$

[1]We convexified the problem by assuming $\{\hat{\mathbf{f}}[t]\}_{t=T_s}^{T_f}$ are independent realizations of random variables $\{\mathbf{f}[t]\}_{t=T_s}^{T_f}$ [22].

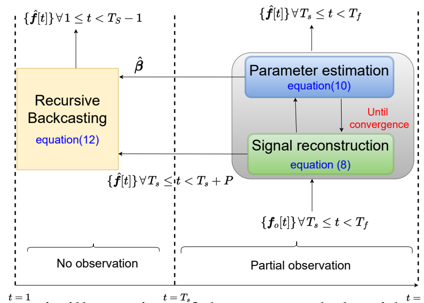

Fig. 2: Schematic illustration of the proposed algorithm. The partial observations are substituted in (9) with initial value $\hat{\boldsymbol{\beta}}$ set as $\mathbf{0}$ to reconstruct the signal. The estimates signals are fed to (11) to update the $\hat{\boldsymbol{\beta}}$ value. Equations (9) and (11) are repeated alternatively until the algorithm converges. Subsequently, the estimated coefficients and signals are substituted in (13) to perform evolution backcasting.

Solution for (12) in closed-form is given by

$$\hat{\mathbf{f}}[T-P] = \left(\mathbf{H}_p(\mathbf{L}_1) + \lambda_1\mathbf{L}_{1,\ell} + \lambda_2\mathbf{L}_{1,u}\right)^{-1}\mathbf{G}_2 \qquad (13)$$

where $\mathbf{G}_2 = \mathbf{H}_p(\mathbf{L}_1)^\top\left(\mathbf{f}[\hat{T}] - \sum_{p=1}^{P-1}\mathbf{H}_p(\mathbf{L}_1)\hat{\mathbf{f}}[T-p]\right)$. The optimization problem can be recursively run for $T = T_s + P - 1$ to $T = P - 1$ to obtain the estimates of $\{\mathbf{f}[t]\}_{t=1}^{T_s-1}$.

5. NUMERICAL RESULTS

We present experiments using edge signals simulated on the Sioux Falls transportation network [23] and Cherry Hills water network [24]. The accuracy of the estimates is quantified using the normalized mean squared error (NMSE), given by

$$\text{NMSE} = \frac{1}{N_1}\sum_{e=1}^{N_1}\frac{\sum_{t=T_s}^{T_f}(f_e[t] - \hat{f}_e[t])^2}{\sum_{t=T_s}^{T_f} f_e[t]^2}, \qquad (14)$$

where $\hat{f}_e[t]$ represents the estimated flow on edge e at time t. The same NMSE computation over the time interval $t=1$ to $t = T_s - 1$ is used to assess the quality of evolution backcasting. The hyperparameters m, λ_2, and μ for all the experiments are grid searched to yield the best NMSE, resulting in $(m, \lambda_2, \mu) = (1, 0, 0.01)$. Note that $\lambda_2 = 0$, given that the number of triangles is minimal in the examples and it does not provide any extra information.

We contrast our estimates with those generated by a comparable data imputation algorithm devised using the standard VAR model and show that our model requires fewer parameters and less training data. Furthermore, many real-world networks, such as water and transportation networks, inherently conserve the flows [8, 25]. By leveraging this contextual prior in the form of a regularizer, we further boost the performance of the proposed algorithm.

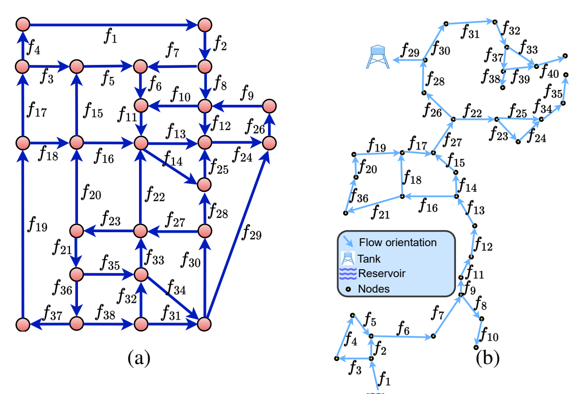

Fig. 3: (a) Sioux Falls transportation network, (b) Cherry Hills flows.

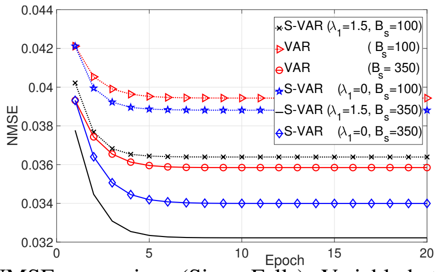

Fig. 4: NMSE comparison (Sioux Falls): Variable batch size (B_S). Omitted $B_S = 175$ for the plot.

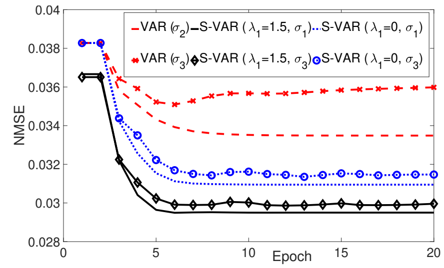

Fig. 5: NMSE comparison (Sioux Falls): Variable noise level ($\sigma_1 = 0.01$, $\sigma_3 = 0.3$). Omitted $\sigma_2 = 0.1$ for the clarity of the plot.

5.1. Sioux Falls Transportation Network

The Sioux Falls transportation network has 24 nodes (0-simplices), 38 edges (1-simplices), and 2 triangles (2-simplices), as shown in Fig. 3 [23]. The time-evolving edge signals are generated by assuming an inverse model as in (4), i.e.,

$$\mathbf{f}[t] = \sum_{p=1}^{3} \left(\sum_{k=0}^{3} \beta_{p,k}^{\ell} \mathbf{L}_{1,\ell}^{k} + \sum_{k=q}^{2} \beta_{p,k}^{u} \mathbf{L}_{1,u}^{k} \right)^{-1} \mathbf{f}[t-p] + \boldsymbol{\varepsilon}_t, \tag{15}$$

where $\boldsymbol{\varepsilon}_t \sim \mathcal{N}(0,1)$. The filter coefficients $\beta_{p,k}^{\ell}$ and $\beta_{p,k}^{u}$are randomly drawn from $\mathcal{N}(0,1)$. To show the influence of the regularizer, we made the flow partially flow-conserving by removing 0.9 times the gradient component of flow signal [19]. We conduct two experiments: i) by adjusting the training data batch size to assess performance with limited training data, and ii) by altering the noise level in the observation model to evaluate the algorithm robustness across varying noise conditions. The initial 20 data samples for both experiments are completely unobservable, and 40% of the remaining flow is observable.

Batch size. We train the model using datasets with three distinct batch sizes $B_s = T_f - T_s + 1 \in \{100, 175, 350\}$. As illustrated in Fig. 4, the conventional VAR model requires 350 data samples to achieve the same accuracy as our proposed algorithm does with just 100 data samples. Notably, even without the flow conservation regularizer ($\lambda_1 = 0$), the proposed model outperforms the standard VAR model. This superiority can be attributed to the simplicial convolutions, which capture intricate dynamics that the traditional VAR model cannot express. The introduction of the flow conservation regularizer further amplifies the algorithm performance, as reflected in Fig. 4. Table 1 further compares the VAR model and the proposed algorithm in evolution backcasting. As anticipated, the conventional VAR model underperforms significantly when juxtaposed with our proposed model. The superior performance of the proposed algorithm can be attributed to two main factors: *(i)* a reduced number of parameters, which minimizes the risk of overfitting, and *(ii)* the incorporation of additional information about the system, specifically regarding the system topology and its flow-conserving nature.

Noise level. We assess the robustness of our algorithm under varying noise levels of the observation model (6). We conduct the tests with three distinct noise levels, generating random noise $\mathbf{e}[t]$ with variances of $\sigma \in \{0.01, 0.1, 0.3\}$. Across all these settings, our algorithm consistently surpasses the traditional VAR model in performance. As depicted in Fig. 5, while the traditional VAR model's performance is sensitive to changes in noise levels, our proposed method remains largely unaffected. The performance of the evolution backcasting can be found in table Table 1; the proposed algorithm performs significantly better than the traditional VAR model.

5.2. Cherry Hills Water Networks

The Cherry Hills water network comprises 36 nodes (or 0-simplices), 40 pipes (or 1-simplices), and 2 triangles (or 2-simplices) [24]. As illustrated in Fig. 3, we establish a reference flow direction and generate flow signals using the EPANET software. Employing a demand-driven model ensures that the water flows satisfy the water requirements at each node. These flow signals represent the volume of water sampled hourly, measured in m^3/h. We evaluate the algorithm under varying conditions of missing flow data. Specifically, we tested it using datasets with 30%, 40%, 50%, and 60% missing flow at given times. As anticipated, the algorithm performance diminishes as the percentage of missing data increases. The traditional VAR model's performance is not shown in the Fig. 6 because the model becomes unstable. This instability of the VAR model can be attributed to its need to learn many parameters without an adequate inductive bias. For evolution backcasting, the performance of the proposed algorithm remains consistent with or without the flow conservation regularizer ($\lambda_1 = 1$ and 0, respectively), resulting in an NMSE of 0.26. On the other hand, the VAR model generates subpar estimates with an NMSE of 10.0.

Table 1: Evolution backcasting: Sioux Falls data

	Algorithms	Batch size 100	175	350
Batch size experiments	S-VAR ($\lambda_1 = 1.5$)	0.040	0.035	0.034
	S-VAR ($\lambda_1 = 0$)	0.052	0.052	0.052
	VAR	3.11	1.26	1.20

	Algorithms	Noise variance σ_1	σ_2	σ_3
Noise level experiments	S-VAR ($\lambda_1 = 1.5$)	0.039	0.039	0.039
	S-VAR ($\lambda_1 = 0$)	0.040	0.040	0.040
	VAR	0.620	0.993	1.816

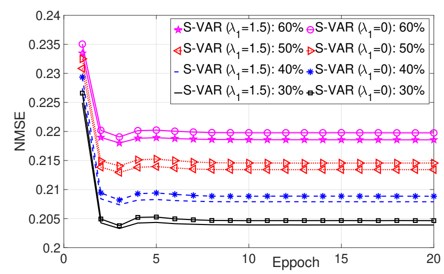

Fig. 6: NMSE comparison using Cherry Hills data. The performance of traditional VAR is not shown as the model proved unstable

6. CONCLUSION

We propose an algorithm to learn a S-VAR model, with a small number of parameters compared to VAR, from incomplete edge signal observations while also completing those observations and backcasting their evolution. The topological structural information encompassed by the S-VAR model and the regularization scheme provides sufficient inductive bias to learn the model without overfitting. As a result, our proposed method adeptly handles evolution backcasting of the edge signal. For future research, we aim to provide theoretical guarantees related to model estimation via identifiability analysis and evolution backcasting via observability analysis.

7. REFERENCES

[1] S. Barbarossa and S. Sardellitti, "Topological signal processing over simplicial complexes," *IEEE Transactions on Signal Processing*, vol. 68, pp. 2992–3007, 2020.

[2] B. Zaman, L. M. Lopez-Ramos, D. Romero, and B. Beferull-Lozano, "Online topology estimation for vector autoregressive processes in data networks," in *2017 IEEE CAMSAP*, 2017, pp. 1–5.

[3] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 2018.

[4] L. Lopez-Ramos, D. Romero, B. Zaman, and B. Beferull-Lozano, "Dynamic network identification from non-stationary vector autoregressive time series," in *2018 IEEE GlobalSIP*, Nov 2018, pp. 773–777.

[5] S. Mollaebrahim and B. Beferull-Lozano, "Design of asymmetric shift operators for efficient decentralized subspace projection," *IEEE Transactions on Signal Processing*, vol. 69, pp. 2056–2069, 2021.

[6] M. Schaub and S. Segarra, "Flow smoothing and denoising: Graph signal processing in the edge-space," in *2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE, 2018, pp. 735–739.

[7] H. Lütkepohl, *New introduction to multiple time series analysis*, Springer Science & Business Media, 2005.

[8] R. Money, J. Krishnan, B. Beferull-Lozano, and E. Isufi, "Online edge flow imputation on networks," *IEEE Signal Processing Letters*, vol. 30, pp. 115–119, 2023.

[9] G. Essl, "Topological IIR filters over simplicial topologies via sheaves," *IEEE Signal Processing Letters*, vol. 27, pp. 1215–1219, 2020.

[10] S. Ebli, M. Defferrard, and G. Spreemann, "Simplicial neural networks," in *TDA & Beyond*, 2020.

[11] M. Schaub, Y. Zhu, J. Seby, T. Roddenberry, and S. Segarra, "Signal processing on higher-order networks: Livin'on the edge... and beyond," *Signal Processing*, vol. 187, pp. 108149, 2021.

[12] M. Yang, E. Isufi, and G. Leus, "Simplicial convolutional neural networks," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 8847–8851.

[13] E. Isufi and M. Yang, "Convolutional filtering in simplicial complexes," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 5578–5582.

[14] J. Krishnan, R. Money, B. Beferull-Lozano, and E. Isufi, "Simplicial vector autoregressive model for streaming edge flows," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[15] G. Bianconi, "Higher-order networks: An introduction to simplicial complexes," *Elements in the Structure and Dynamics of Complex Networks*, 2021.

[16] L. Lim, "Hodge laplacians on graphs," *Siam Review*, vol. 62, no. 3, pp. 685–715, 2020.

[17] D. Shuman, S. 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.

[18] B. Sergio and S. Stefania, "Topological signal processing over simplicial complexes," *IEEE Transactions on signal processing*, 2020.

[19] M. Yang, E. Isufi, M. Schaub, and G. Leus, "Simplicial convolutional filters," *IEEE Transactions on Signal Processing*, vol. 70, pp. 4633–4648, 2022.

[20] M. Amarlingam, S. Santiago, C. Sundeep, and M. Antonio, "Generative adversarial networks for graph data imputation from signed observations," *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2020.

[21] V. Ioannidis, Y. Shen, and G. Giannakis, "Semi-blind inference of topologies and signals over graphs," in *2018 IEEE Data Science Workshop (DSW)*, 2018, pp. 165–169.

[22] R. Money, J. Krishnan, and B. Beferull-Lozano, "Online joint nonlinear topology identification and missing data imputation over dynamic graphs," *European Signal Processing Conference (EUSIPCO)*, 2022.

[23] L. Leblanc, "An algorithm for the discrete network design problem," *Transportation Science*, vol. 9, no. 3, pp. 183–199, 1975.

[24] L. Rossman, R. Clark, and W. Grayman, "Modeling chlorine residuals in drinking water distribution systems," *Journal of Environmental Engineering*, vol. 120, no. 4, pp. 803–820, 1994.

[25] J. Jia, M. Schaub, S. Segarra, and A. Benson, "Graph-based semi-supervised & active learning for edge flows," *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining - KDD '19*, 2019.