Predicting Higher-Order Dynamics With Unknown Hypergraph Topology

Zhou, Zili; Li, Cong; Mieghem, Piet Van; Li, Xiang

DOI
10.1109/TCSI.2024.3513406

Publication date
2024

Document Version
Final published version

Published in
IEEE Transactions on Circuits and Systems I: Regular Papers

Predicting Higher-Order Dynamics With Unknown Hypergraph Topology

Zili Zhou, Cong Li, *Member, IEEE*, Piet Van Mieghem, *Fellow, IEEE*, and Xiang Li, *Senior Member, IEEE*

***Abstract*— Predicting future dynamics on networks is challenging, especially when the complete and accurate network topology is difficult to obtain in real-world scenarios. Moreover, the higher-order interactions among nodes, which have been found in a wide range of systems in recent years, such as the nets connecting multiple modules in circuits, further complicate accurate prediction of dynamics on hypergraphs. In this work, we proposed a two-step method called the topology-agnostic higher-order dynamics prediction (TaHiP) algorithm. The observations of nodal states of the target hypergraph are used to train a surrogate matrix, which is then employed in the dynamical equation to predict future nodal states in the same hypergraph, given the initial nodal states. TaHiP outperforms three latest Transformer-based prediction models in different real-world hypergraphs. Furthermore, experiments in synthetic and real-world hypergraphs show that the prediction error of the TaHiP algorithm increases with mean hyperedge size of the hypergraph, and could be reduced if the hyperedge size distribution of the hypergraph is known.**

***Index Terms*— Nonlinear system, dynamics on networks, predicting higher-order dynamics, contagion, hypergraph.**

I. Introduction

THE study of dynamical processes in networked systems is fundamental in complexity science [1], [2]. Examples include the cascading failures in power systems [3], the propagation diseases in complex networks [4] and synchronization in coupled oscillator networks [5], [6]. The dynamical processes on networks are determined by two independent parts: the topological structure of the network and the fundamental physical rule that governs the system. A significant amount of research [7], [8], [9], [10] is dedicated to investigating the influence of the network structure on dynamical processes, assuming prior knowledge of the network topology.

One of the crucial challenges in the study of dynamical processes is the prediction of future dynamics on networks, such as predicting cascading failures in power grids [11] and predicting the spread of an infectious disease on a human contact network. In a network of N nodes, we denote the nodal state of node i at time t by $x_i(t)$. The adjacency matrix of the network is denoted as $\mathbf{A}$ with elements A_{ij}, where $A_{ij} = 1$ if nodes i and j are connected, otherwise $A_{ij} = 0$. The intricate interplay between network structure and physical rules that govern the system could be captured in the following equation, which characterizes a general class of dynamical process on networks [12], [13], [14]

$$\frac{dx_i(t)}{dt} = f\left(x_i(t)\right) + \sum_{j=1}^{N} A_{ij} g\left(x_i(t), x_j(t)\right) , \qquad (1)$$

where $f(x_i)$ represents the internal dynamics of node i, and the interaction between node i and any other node j depends on the adjacency matrix $\mathbf{A}$ and the interaction function $g(x_i(t), x_j(t))$. For certain complex systems, the underlying functions f and g in (1) are unknown. Additionally, the majority of real-world network topologies is complicated, and a sufficiently accurate network reconstruction is a difficult task [15]. These factors further complicate the accurate prediction of complex system dynamics.

In system where the dynamics has been modeled by (1), it appears intuitive to first infer the topology of the network and then predict the dynamics based on the inferred topology. However, recent research [16] has demonstrated that a general class of autonomous dynamics without any control, including the Lotka-Volterra model of population dynamics [17], the susceptible-infected-susceptible model of epidemic spreading [4] and the Kuramoto model of synchronization phenomena [18] could be accurately predicted with the observations of dynamics on an unknown network while the functions f and g in the governing (1) are known. This prediction was achieved by a surrogate network, which was derived by fitting past observations of nodal states to the dynamical process. Counterintuitively, despite the significant divergence between the surrogate topology and the network topology, the prediction remained accurate. This method relied

Received 21 June 2024; revised 17 September 2024 and 14 November 2024; accepted 2 December 2024. Date of publication 11 December 2024; date of current version 31 March 2025. This work was supported in part by the National Natural Science Foundation of China under Grant U23A20331 and Grant 62173095 and in part by the Natural Science Foundation of Shanghai under Grant 21ZR1404700. The work of Piet Van Mieghem was supported by European Research Council (ERC) under European Union's Horizon 2020 Research and Innovation Program under Agreement 101019718. This article was recommended by Associate Editor A. Ascoli. *(Corresponding author: Cong Li.)*

Zili Zhou and Cong Li are with the Adaptive Networks and Control Laboratory, Department of Electronic Engineering, School of Information Science and Technology, Fudan University, Shanghai 200433, China (e-mail: 21210720046@m.fudan.edu.cn; cong_li@fudan.edu.cn).

Piet Van Mieghem is with the Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: P.F.A.VanMieghem@tudelft.nl).

Xiang Li is with the Institute of Complex Networks and Intelligent Systems, Shanghai Research Institute for Intelligent Autonomous Systems, the Frontiers Science Center for Intelligent Autonomous Systems, and the State Key Laboratory of Intelligent Autonomous Systems, Tongji University, Shanghai 201210, China (e-mail: lix@fudan.edu.cn).

Digital Object Identifier 10.1109/TCSI.2024.3513406

on a powerful tool for discovering low-dimensional structures in dynamics, the proper orthogonal decomposition (POD) [19] and applied to static networks.

The aforementioned study assumed that dynamic processes are on networks with pairwise interactions. Recent studies [20], [21], [22], [23], [24] have indicated that pairwise interactions may inadequately capture the intricate dependencies among nodes in certain systems, and the existence of higher-order interactions has been proved to profoundly influence the dynamics of networked systems, from the cascading failure problem in power systems [25], diffusion [26] and synchronization [27], to social processes [28]. To capture higher-order interactions among nodes, more advanced mathematical structures, such as hypergraphs and simplicial complexes [20] were developed (See detailed definition of hypergraph in Section II-A). In the field of circuits, higher-order structures were used to construct topological insulators [29], [30]. The multiple modules in a circuit netlist can be represented as a hypergraph, where the nodes correspond to the modules and the hyperedges correspond to the interconnections among modules, and the hypergraph model has been used in the circuit partitioning problem [31], [32], [33], which is one of the central problems in very large scale integration circuit (VLSI) system design. However, the prediction of dynamics on hypergraphs has not been fully studied. Whether the prediction of higher-order dynamics is similar to the prediction of network dynamics, which can be carried out without network topology [16], is a question deserving extensive research.

In this work, the prediction of higher-order dynamics on hypergraphs refers to the prediction of future nodal states of a hypergraph. In fact, the temporal evolution of a nodal state is also a time series. From this perspective, the prediction of higher-order dynamics on hypergraphs is a multivariate time series forecasting problem, where each variate is the state of a node in the hypergraph. In recent years, various deep learning models [34], [35], [36], [37] have been developed to solve the time-series forecasting problem. Most of them are categorized as data-driven methods [38], as they rely exclusively on training datasets as input to predict other datasets, disregarding the underlying processes that generate the time series. However, when specific knowledge about the generation of the time series is available (e.g., the governing equations of a dynamical system), a question arises consequently: how can such prior knowledge be integrated into the prediction method?

The approach proposed in [39] was an attempt to implement this idea. This approach extended the method in [16] to hypergraphs, but only applied to limited-size synthetic hypergraphs with 100 nodes, consisting of size-2 and size-3 hyperedges. Due to the space complexity of $O(2^n)$, the scalability of this method was severely constrained for larger networks. However, hyperedges that contain more than 3 nodes are abundantly present in real-world hypergraphs, and most of the hypergraphs contain far more than 100 nodes (see Table I for structural features of real-world hypergraphs). Therefore, it remains a challenge to predict the future states of nodes in real-world hypergraphs with unknown topological structures, where hyperedges of any orders may exist.

In this work, we propose a novel method, called **T**opology-**a**gnostic **Hi**gher-order Dynamics **P**rediction (TaHiP) algorithm, to predict the dynamics of a contagion processes [40] on general hypergraphs, including real-world hypergraphs and synthetic hypergraphs with different structural features. The main contributions of this work can be summarized as follows:

- Unlike most of the purely data-driven deep learning models, the proposed TaHiP algorithm incorporates the governing equation of dynamical process and predict the future states of nodes (with the form of time-series) on a hypergraph with unknown structure. In various real-world hypergraphs, TaHiP greatly outperforms three latest data-driven time-series forecasting methods used as baselines [35], [36], [37]. Furthermore, TaHiP needs less data input (an $N \times 1$ initial state vector) to predict than the baseline methods, and its single-layer structure leads to less training time and less storage space than the multi-layer baseline models.
- Specifically, as the topological structure of the hypergraph is unknown, we constructed a surrogate matrix to replace the unknown incidence matrix. We embedded the elements of the surrogate matrix of a hypergraph in exponential forms into the governing equation of the system. The elements in this matrix are then continuously optimized using the node state observation data of the target hypergraph. Finally, the optimised surrogate matrix is used to predict future states of nodes of the hypergraph.
- Following the definition of hyperdegree, we define the *predicted hyperdegree* of node based on the optimized surrogate matrix generated by TaHiP. We calculate the Pearson correlation coefficients between metrics including he hyperdegree (HD), the predicted hyperdegree (PHD), the higher-order H-index (HOH) and the higher-order PageRank (HOP), and find that PHD is highly correlated with HD and HOP when the prediction is accurate.

The rest of the paper is organized as follows. Section II gives the problem statement and introduces the proposed TaHiP algorithm. Section III first reports a few structural features of the adopted hypergraphs, and then shows the performances of TaHiP on real-world hypergraphs, which are compared with other prediction methods. Section IV analyses the relation between the surrogate matrix obtained by TaHiP and the hypergraph topology. Finally, Section V concludes the paper and stipulates future directions.

II. Topology-Agnostic Higher-Order Dynamics Prediction

A. Problem Statement

A hypergraph $\mathcal{H} = (\mathcal{V}, \mathcal{E})$ is defined as a set of nodes $\mathcal{V} = \{v_i\}$ and a set of hyperedges $\mathcal{E} = \{e_j\}$, with $N = |\mathcal{V}|$ being the number of nodes and $E = |\mathcal{E}|$ being the number of hyperedges. We denote $\mathcal{E}_i$ as the set of hyperedges that contain the node i, and the cardinality of a hyperedge e_j as $|e_j|$. Formally, the incidence matrix $\mathbf{B} \in \mathbb{R}^{N \times E}$ of hypergraph

$\mathcal{H}$ can be denoted as

$$B_{ij} = \begin{cases} 1, & \text{if } v_i \in e_j \\ 0. & \text{if } v_i \notin e_j \end{cases} \tag{2}$$

By its definition, each column in the incidence matrix $\mathbf{B}$ represents a hyperedge in the corresponding hypergraph, and each row in $\mathbf{B}$ shows the specific hyperedges to which a node belongs. We refer to Appendix A and [41] for more properties of the incidence matrix.

In this work, we adopt a contagion process on hypergraphs in [40] as our target dynamics to be predicted. In this process, the nodal state $x_i(t)$ represents the probability of an individual to be active at time t. The contagion process can be written as the form of (3) in Section I, with the self-dynamics of node i being $f(x_i(t)) = -\delta x_i(t)$, in which δ denotes the curing rate of an individual, and the interaction function $g(x_i(t), \ x_j(t)) = \beta\,(1 - x_i(t)) \sum_{e_j \in \mathcal{E}_i} \prod_{\substack{v_k \in e_j \\ k \neq i}} x_k(t)$ is the product of the probability that node i is inactive and the other nodes in the hyperedge e_j are active times a contact rate, β. Hence, we have

$$\frac{dx_i(t)}{dt} = -\delta x_i(t) + \beta\,(1 - x_i(t)) \sum_{e_j \in \mathcal{E}_i} \prod_{\substack{v_k \in e_j \\ k \neq i}} x_k(t). \tag{3}$$

For a hypergraph $\mathcal{H}$ with unknown topology consisting of N nodes, we are given an initial nodal state vector $\mathbf{x}(0) = [x_1(0), x_2(0), \cdots, x_N(0)]^T \in \mathbb{R}^N$ following a specific distribution and the observed nodal state matrix for training

$$\mathbf{X}_{train} = [\mathbf{x}(0), \mathbf{x}(1), \cdots, \mathbf{x}(T-1)] \in \mathbb{R}^{N \times T}\ , \tag{4}$$

which describes a contagion process on this hypergraph, with $\mathbf{x}(0)$ being the initial value. In matrix $\mathbf{X}_{train}$, each column $\mathbf{x}(t) \in \mathbb{R}^N$ stores the states of all the nodes observed at time t, with the total number of time steps being T.

To test the accuracy of the prediction algorithm, we obtained another nodal state vector $\mathbf{x}'(0)$ by sampling from the same distribution as x(0), and then construct $\mathbf{X}_{test} \in \mathbb{R}^{N \times T}$ by generating a contagion process on the same hypergraph $\mathcal{H}$ with $\mathbf{x}'(0)$ as the initial value

$$\mathbf{X}_{test} = [\mathbf{x}'(0), \mathbf{x}'(1), \cdots, \mathbf{x}'(T-1)] \in \mathbb{R}^{N \times T}. \tag{5}$$

The prediction algorithm is supposed to output a matrix

$$\mathbf{X}_{pred} = [\mathbf{x}'(0), \mathbf{x}_p(1), \cdots, \mathbf{x}_p(T-1)] \in \mathbb{R}^{N \times T}\ , \tag{6}$$

given the initial nodal state vector $\mathbf{x}'(0)$, which is the same as $\mathbf{X}_{test}$. Finally, the accuracy of the prediction algorithm is measured by different metrics computing the error between $\mathbf{X}_{test}$ and $\mathbf{X}_{pred}$. We refer to Fig. 1 for more details of the generation of the aforementioned datasets.

B. Iterative Equation of the Contagion Process

The equation (3) of the contagion process defines the influence of other nodes on node i by first calculating the product of the states of other nodes that share the same hyperedge as node i. Then, the resulting products are summed based on the hyperedge to which node i belongs. However, (3) must be transformed to a general form of $\mathbf{x}(t+1) = F(\mathbf{x}(t), \ \mathbf{B})$ to compute the observed nodal state matrix $\mathbf{X}_{train}$ of a contagion process, so that given the nodal state vector $\mathbf{x}(t) \in \mathbb{R}^N$ at any time t and the incidence matrix $\mathbf{B}$, the nodal state vector at next time step, $\mathbf{x}(t+1)$ can be directly calculated.

First, we approximate the derivative $\frac{dx_i(t)}{dt}$ in (3) by a difference quotient,

$$\frac{x_i(t+1) - x_i(t)}{\Delta t} = -\delta x_i(t) + \beta\,(1 - x_i(t)) \sum_{e_j \in \mathcal{E}_i} \prod_{\substack{v_k \in e_j \\ k \neq i}} x_k(t)\ , \tag{7}$$

where Δt denotes the time interval. (7) can be rewritten as

$$x_i(t+1) = (1 - \delta \Delta t) x_i(t) + \Delta t \beta (1 - x_i(t)) \sum_{e_j \in \mathcal{E}_i} \prod_{\substack{v_k \in e_j \\ k \neq i}} x_k(t). \tag{8}$$

To transform (8) to the form of $\mathbf{x}(t+1) = F(\mathbf{x}(t), \ \mathbf{B})$, the term $\sum_{e_j \in \mathcal{E}_i} \prod_{\substack{v_k \in e_j \\ k \neq i}} x_k(t)$ that describes the influence of other nodes on node i must be simplified to a general term that applies to any node in the hypergraph, regardless of its location.

The incidence matrix $\mathbf{B}$ allows to compute the product of nodal states of any hyperedge e_j in the hypergraph as

$$\prod_{v_k \in e_j} x_k(t) = \prod_{k=1}^{N} x_k^{B_{kj}}(t). \tag{9}$$

If $B_{kj} = 0$, implying that node v_k is not in hyperedge e_j, then we have $x_k^{B_{kj}}(t) = x_k^0(t) = 1$, which will not affect the product in (9). If $B_{kj} = 1$ and node k is in hyperedge e_j, then $x_k^{B_{kj}}(t) = x_k^1(t) = x_k(t)$, which will be included in the product. Applying (9) to all the hyperedges yields a vector

$$e_{prod}^T = [\prod_{k=1}^{N} x_k^{B_{k1}}(t), \ \prod_{k=1}^{N} x_k^{B_{k2}}(t), \ \cdots, \ \prod_{k=1}^{N} x_k^{B_{kE}}(t)] \in \mathbb{R}^{1 \times E}\ . \tag{10}$$

We denote the sum of products of nodal states within all the hyperedges that node v_k belongs to as

$$s_k = \sum_{e_j \in \mathcal{E}_k} \prod_{k=1}^{N} x_k^{B_{kj}}(t)\ , \tag{11}$$

and the vector consisting of $s(i)$ of all the nodes

$$\mathbf{s}^T = [s_1, s_2, \cdots, s_N] \in \mathbb{R}^{1 \times N}\ , \tag{12}$$

which can be written in matrix form as

$$\mathbf{s} = \mathbf{B}\mathbf{e}_{prod}\ . \tag{13}$$

Moreover, the state of node i itself need to be eliminated from the product, and we obtain the general term that describes the influence of other nodes on any node i

$$\sum_{e_j \in \mathcal{E}_i} \prod_{\substack{v_k \in e_j \\ k \neq i}} x_k(t) = \frac{1}{x_i(t)} s_i = \frac{1}{x_i(t)} (\mathbf{B}\mathbf{e}_{prod})_i\ , \tag{14}$$

where $(Be_{prod})_i$ denotes the ith element of the vector Be_{prod}. (14) is then substituted back into (8)

$$x_i(t+1) = (1-\delta\Delta t)x_i(t) + \Delta t\beta \frac{1-x_i(t)}{x_i(t)}(\mathbf{Be}_{prod})_i \ , \tag{15}$$

and written in matrix form as

$$\mathbf{x}(t+1) = (1-\delta\Delta t)\mathbf{x}(t) + \Delta t\beta \frac{1-\mathbf{x}(t)}{\mathbf{x}(t)}\mathbf{Be}_{prod}. \tag{16}$$

Equation (16) transforms (8) to the general form of

$$\mathbf{x}(t+1) = F(\mathbf{x}(t),\ \mathbf{B}) \ , \tag{17}$$

which leads to an immediate calculation of the nodal state vector $\mathbf{x}(t+1)$ given $\mathbf{x}(t)$ and the incidence matrix $\mathbf{B}$. We refer to (16) as the iterative equation of contagion process, which will be repeatedly used in the generation of the observed data and the prediction algorithm.

C. TaHiP Algorithm

As the topology of the target hypergraph is unknown, we aim to find a surrogate matrix $\hat{\mathbf{B}}$ to replace the incidence matrix of the hypergraph. The surrogate matrix can be substituted into (16) along with any initial nodal state vector to generate predictions for future states of nodes in the hypergraph. The framework of TaHiP algorithm is illustrated in Fig. 1. The algorithm consists of three steps:

1) **Initialization of the surrogate matrix $\hat{\mathbf{B}}$.**
To initialize the surrogate matrix $\hat{\mathbf{B}} \in \mathbb{R}^{N\times E}$ from a zero matrix, we first generate a set, $e_{sample} = \{|e_1|, |e_2|, \cdots, |e_E|\}$ by sampling from a Poisson distribution with a fixed parameter λ, $f(n;\lambda) = Pr[X = n] = \frac{\lambda^n e^{-\lambda}}{n!}$. Then, for the jth column of the surrogate matrix, we randomly select $|e_j|$ elements to be replaced by 1. After performing this operation for all columns, we obtain an initialized binary matrix $\hat{\mathbf{B}}_0$. The choice of other distributions for initialization will be discussed in Section III-D.
2) **Training, with the optimization of the surrogate matrix $\hat{\mathbf{B}}$.**
The optimization of the surrogate matrix is the core of the proposed algorithm. As is shown in the left of Fig. 1, the matrix $\mathbf{X}_{train} \in \mathbb{R}^{N\times T}$ obtained by the observation of nodal states in the unknown hypergraph is the input of the algorithm. For each training epoch, a submatrix $\mathbf{X}_{sample} \in \mathbb{R}^{N\times k}$ is randomly sampled from $\mathbf{X}_{train}$, consisting of k consecutive nodal state vectors. Here, k is referred to as the random window length. For each vector in the first $k-1$ vectors in $\mathbf{X}_{sample}$, namely, $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_{k-1}$, we apply Eq. (16) with the current surrogate matrix $\hat{\mathbf{B}}$, resulting in vectors $\tilde{\mathbf{x}}_2, \tilde{\mathbf{x}}_3, \cdots, \tilde{\mathbf{x}}_k$. Then, we use the MSE (mean square error) loss to measure the discrepancy between the prediction $\tilde{\mathbf{x}}_k$ and the ground truth $\mathbf{x}_k$. The loss of each vector pair, denoted as $MSE(\mathbf{x}_i, \tilde{\mathbf{x}}_i)$, is gathered to get the overall objective loss $\sum_{i=2}^{k} MSE(\mathbf{x}_i, \tilde{\mathbf{x}}_i)$, which is used for the backward process. The elements in $\hat{\mathbf{B}}$ will be updated in the end of each training epoch by $\hat{B}_{ij(updated)} = \hat{B}_{ij} + \alpha * (\partial Loss/\partial\hat{\mathbf{B}})_{ij}$, in which α denotes the learning rate. We used the Adam algorithm [42] as the optimizer. The maximum number of epochs is set as 1000 to assure the convergence of loss. We have tested the impact of k have on the prediction performance. The results showed that different values of k had no significant effect on the performance of TaHiP. Therefore, we set k as 100 to get an acceptable time of training. We refer to Appendix B for the convergence analysis of TaHiP, and Appendix C for the complexity analysis.
3) **Prediction with the surrogate matrix $\hat{\mathbf{B}}$.**
We start with the initial nodal state vector $\mathbf{x}'(0)$, which follows the same distribution as the initial nodal state vector $\mathbf{x}(0)$ of $\mathbf{X}_{train}$. The predicted nodal state matrix $\mathbf{X}_{pred} \in \mathbb{R}^{N\times E}$ is obtained by iterating (16) with the optimized surrogate matrix $\hat{\mathbf{B}}$ and the initial value of $\mathbf{x}'(0)$ for T times. Meanwhile, the matrix $\mathbf{X}_{test} \in \mathbb{R}^{N\times E}$ used as the ground truth is obtained by iterating (16) with the incidence matrix $\mathbf{B}$ of the hypergraph and the initial value of $\mathbf{x}'(0)$ for T times. Finally, we calculate the MSE and MAE (mean absolute error) between the matrix $\mathbf{X}_{test}$ and the matrix $\mathbf{X}_{pred}$ as metrics, to evaluate the performance of the proposed TaHiP algorithm.

The pseudocode of optimization of the surrogate matrix is provided in **Algorithm 1**.

Algorithm 1 Training Part of TaHiP

Input:
$\hat{\mathbf{B}}_0 \in \mathbb{R}^{N\times E}$: the initialized surrogate matrix;
$\mathbf{X}_{train} \in \mathbb{R}^{N\times T}$: the observed nodal state matrix for training;
M: training epochs;
α: learning rate;
F: the function for iteration, defined in Eq.(16) and Eq.(17).
Output:
$\hat{\mathbf{B}} \in \mathbb{R}^{N\times E}$: the optimized surrogate matrix.

```
while m ≤ M do
    X_sample ← randomly select N × k submatrix from X_train;
    q ← 1;
    Loss ← 0;
    while q ≤ k − 1 do
        x̃_{q+1} ← F(x_q, B̂)
        Loss ← Loss + MSE(x̃_{q+1}, x_{q+1})
        q ← q + 1
        Backward with Loss, update elements in B̂ by
        B̂_{ij(updated)} = B̂_ij + α * (∂Loss/∂B̂)_ij using Adam
        optimizer
        m ← m + 1
    end while
end while
```

There are a few remarks worth emphasizing. The initialized surrogate matrix $\hat{\mathbf{B}}_0$ is binary. Thus, it can be regarded as an incidence matrix of a random unweighted hypergraph.

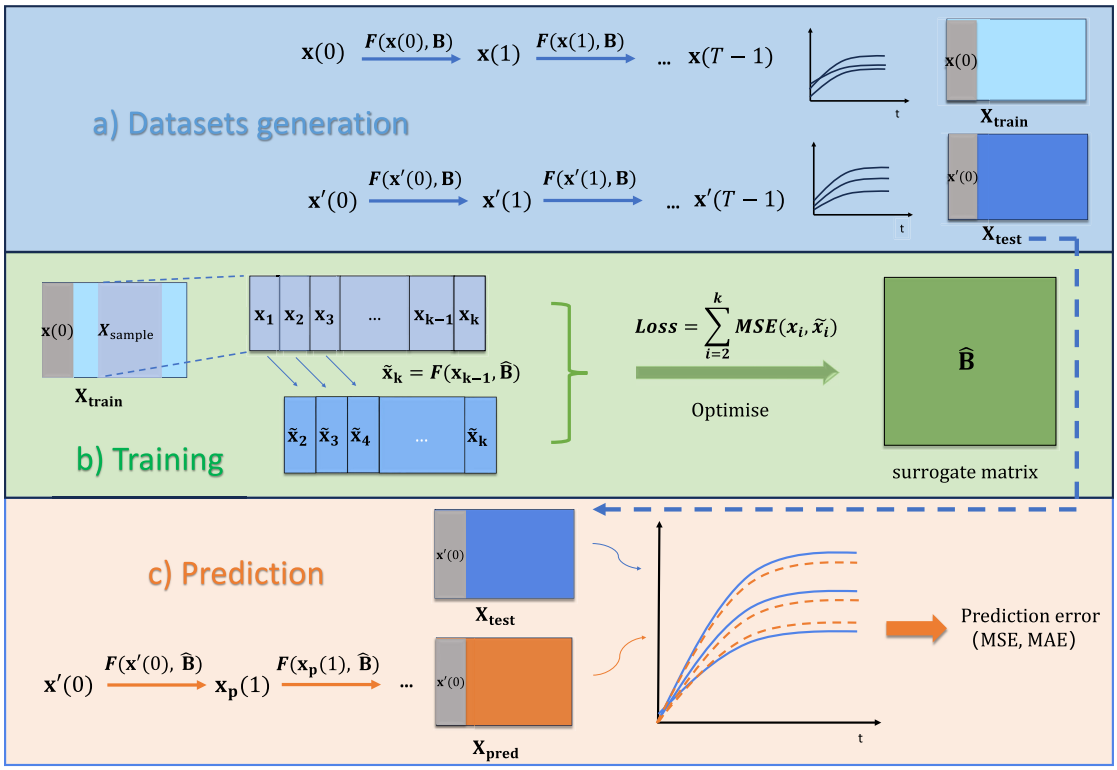

Fig. 1. TaHiP architecture. **(a) Datasets generation**. The training dataset $\mathbf{X}_{train}$ and the test dataset $\mathbf{X}_{test}$ were obtained by applying function F defined in (17) to the corresponding initial vector $\mathbf{x}(0)$ and $\mathbf{x}'(0)$ recursively with the incidence matrix $\mathbf{B}$. **(b) Training**. We first conducted one-step inference with the current surrogate matrix by $\tilde{\mathbf{x}}_k = F(\mathbf{x}_{k-1}, \hat{\mathbf{B}})$, and then updated the elements of $\hat{\mathbf{B}}$ according to the constructed loss. **(c) Prediction**. Applying function F defined in (17) to the initial vector $\mathbf{x}'(0)$ recursively with the optimised surrogate matrix $\hat{\mathbf{B}}$ gave the final prediction matrix $\mathbf{X}_{pred}$.

However, during the optimization process, the elements of $\hat{\mathbf{B}}_0$ will be updated according to the calculated gradients, leading to a real matrix $\hat{\mathbf{B}}$, which no longer corresponds to the topology of any hypergraph as an incidence matrix.

In the definition of incidence matrix (2), a hyperedge is modelled by a vector consisting of binary elements that determine whether a node is in the hyperedge. We consider this way of representing as a discrete form of modeling higher-order interactions between nodes. In a more general sense, the intensity of interactions between nodes should be more precisely described with real numbers, including negative values. Therefore, the proposed TaHiP algorithm does not impose any constraint on the elements of the surrogate matrix. This setting distinguishes the TaHiP algorithm from existing studies on hypergraph structure inference [43], [44], [45] or hyperedge prediction [46], [47], [48]. We do not focus on the topology of the target hypergraph, but on constructing a surrogate matrix that can be used for predicting dynamics in the hypergraph.

III. Experiments and Results

A. Structural Features of Real-World Hypergraphs

The 6 real-world hypergraphs in this work are derived from empirical data from various domains, and were collected by the author of [49]. The *contact-high-school* and *contact-primary-school* datasets are hypergraphs of groups of people in contact at a high/primary school. The *email-Enron* and *email-Eu* datasets are hypergraphs of sets of email addresses on emails. The *senate-bills* and *house-bills* datasets are hypergraphs modelling bill cosponsorship in the US House/Senate of Representatives. For each hypergraph, we report a diverse range of structural properties such as number of nodes, hyperedges and their sizes, as detailed in Table I. The definitions of some features are listed below.

Definition 1 Hyperdegree: For a node i in a hypergraph, the set of all the hyperedges that contain node i is denoted as $\mathcal{E}_i$. The hyperdegree $d_H(i)$ of node i is defined as the number of elements contained in $\mathcal{E}_i$ and can be calculated by $d_H(i) = \sum_{j=1}^{E} B_{ij}$. Furthermore, $\langle d_H \rangle$ denotes the mean hyperdegree of all nodes in the hypergraph, given by $\langle d_H \rangle = \frac{1}{N}\sum_{i=1}^{N} d_H(i)$.

Definition 2 Hyperedge Size: The hyperedge size $|e_j| = \sum_{i=1}^{N} B_{ij}$ is the number of nodes contained in the hyperedge e_j. Additionally, $|e_j|_{max}$ is the maximum hyperedge size in the hypergraph, $\langle |e_j| \rangle$ denotes the mean hyperedge size with $\langle |e_j| \rangle = \frac{1}{E}\sum_{j=1}^{E} |e_j|$, and $\%|e_j| = 2$ is defined as the ratio of the number of size-2 hyperedges to the number of all hyperedges, which indicates the proportion of low-order interactions among all interactions.

Definition 3 Hyperedge Density [50]: The hyperedge density ρ is defined as the ratio of the number of hyperedges to the number of nodes, *i.e.*, $\rho = E/N$.

Table I shows that the selected real-world hypergraphs exhibit differences not only in basic parameters such as the number of nodes and hyperedges, but also in higher-order structural features such as mean hyperdegree and mean hyperedge size.

TABLE I
STRUCTURAL FEATURES OF 6 REAL-WORLD HYPERGRAPHS

	N	E	$\langle d_H \rangle$	ρ	$\lvert e_j \rvert_{max}$	$\langle \lvert e_j \rvert \rangle$	$\%\lvert e_j \rvert = 2$
contact-high-school	327	7818	55.63	23.91	5	2.3	0.703
contact-primary-school	242	12799	127.0	52.89	5	2.4	0.61
email-Enron	148	1512	30.74	10.22	18	3	0.535
email-Eu	998	25027	85.31	25.08	25	2.33	0.510
senate-bills	294	21721	731.8	73.88	99	9.90	0.151
house-bills	1494	54933	814.3	36.77	399	22.1	0.139

TABLE II
CONTACT RATE β SET FOR EACH REAL-WORLD HYPERGRAPH

hypergraph	β
contact-high-school	0.1
contact-primary-school	0.05
email-Enron	0.25
email-Eu	0.05
senate-bills	0.075
house-bills	0.2

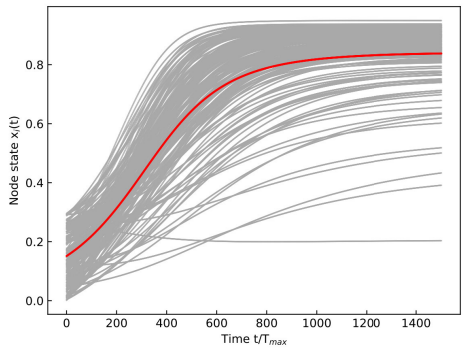

Fig. 2. Nodal states of contagion process with $\beta = 0.1$ and $\delta = 1$ in hypergraph *contact-high-school*. Each grey curve is the temporal evolution of a nodal state, and the red curve shows the change of mean nodal state $\overline{x_i(t)}$ over time.

B. Experimental Settings

The observed nodal state matrix $\mathbf{X}_{train}$ is generated the same way as $\mathbf{X}_{test}$, by iterating (16) with the incidence matrix $\mathbf{B}$ of the hypergraph and the initial value of $\mathbf{x}(0)$ for T times. The elements of all the initial nodal vectors ($\mathbf{x}(0)$ of $\mathbf{X}_{train}$, $\mathbf{x}'(0)$ of $\mathbf{X}_{test}$ and $\mathbf{X}_{pred}$) are sampled from a uniform distribution $U(0,\ 0.3)$, and the total number of time steps T is set as 1500. The time interval Δt is set as 0.001.

We fix the curing rate δ as 1 and adjust the contact rate β to control the relative strength between the curing (deactivation) process and the infection (activation) process. The contact rate β directly affects the contagion process in the hypergraph, which can be quantitatively described by the temporal evolution of the mean nodal state $\overline{x_i(t)} = \frac{1}{N}\sum_{i=1}^{N} x_i(t)$. A larger β enables all nodes in the hypergraph to reach different steady states more quickly, while a smaller β may hinder the contagion process, with the mean node state $\overline{x_i(t)}$ monotonically decreasing over time. We set β for each real-world hypergraph in Table II to ensure that $\overline{x_i(t)}$ reaches a steady state in time $[T/2, T]$.

To illustrate, we plot the contagion process with $\beta = 0.1$ in hypergraph *contact-high-school* in Fig. 2.

We use two metrics, namely, MSE and MAE to measure the performance of the TaHiP algorithm. Specifically, we compute the error between the matrix $\mathbf{X}_{test}$ and the matrix $\mathbf{X}_{pred}$ by

$$MSE(\mathbf{X}_{test},\ \mathbf{X}_{pred}) = \frac{\sum_{i=1}^{N}\sum_{j=1}^{T}(\mathbf{X}_{test}(i,\ j) - \mathbf{X}_{pred}(i,\ j))^2}{N \times T}, \tag{18}$$

$$MAE(\mathbf{X}_{test},\ \mathbf{X}_{pred}) = \frac{1}{N \times T}\sum_{i=1}^{N}\sum_{j=1}^{T}|\mathbf{X}_{test}(i,\ j) - \mathbf{X}_{pred}|. \tag{19}$$

C. Comparing the Performance of TaHiP With Other Prediction Methods

1) Comparing With Network Dynamics Prediction Method: In recent years, an increasing body of research from various domains have supported the existence of higher-order interactions in different systems. On the one hand, there is evidence [51], [52] suggesting the modelling of higher-order interactions might in some cases be redundant, and may be fully described by combination of pairwise interactions. In the contagion process on hypergraphs studied in this work, is it necessary to explicitly model the higher-order interactions among nodes?

The SIS (Susceptible-Infected-Susceptible) model in networks is one of the dynamics that can be accurately predicted without knowing the network topology in [16]. The governing equation of SIS model is shown as

$$\frac{dx_i(t)}{dt} = -\delta x_i(t) + (1 - x_i(t))\sum_{j=1}^{N} A_{ij}x_j(t). \tag{20}$$

The equation of the contagion process on hypergraphs, i.e., (3) extends (20) to hypergraphs, by replacing the $x_j(t)$ term of single node state (according to the definition of pairwise interaction) by the products of nodes in the same hyperedge, $\prod_{\substack{k\in e_j \\ k\neq i}} x_k(t)$. Suppose we are given the observed data of a contagion process on hypergraphs $\mathbf{X}_{train} \in \mathbb{R}^{N\times T}$, without knowing the underlying process. We adopt the method proposed in [16] to predict the nodal state matrix $\mathbf{X}_{pred} \in \mathbb{R}^{N\times T}$, assuming the underlying process is the SIS epidemic process without higher-order interaction. If the prediction is accurate, then the modelling of higher-order interaction in the contagion model will be unnecessary.

The prediction of dynamics on hypergraph *contact-high-school* is illustrated in Fig. 3, which shows that the network dynamics prediction method is incapable of predicting the higher-order contagion process. Predictions on

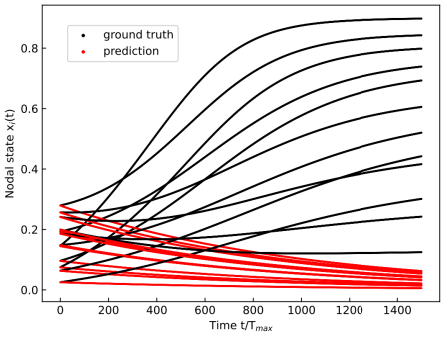

Fig. 3. Prediction of dynamics on hypergraph *contact-high-school* by the network dynamics prediction method proposed in [16]. The black curves are the true nodal states $x_i(t)$, and red curves are the predictions of the corresponding nodes. For clarity, 12 randomly chosen nodal states are depicted.

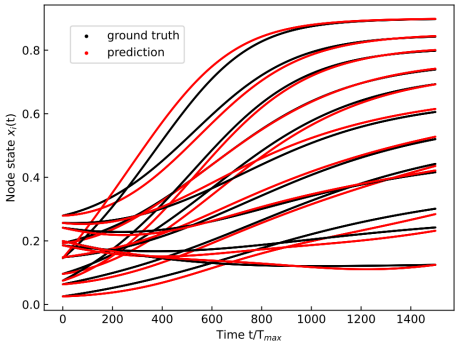

Fig. 4. Prediction of dynamics on hypergraph *contact-high-school* by the TaHiP algorithm. The black curves are the true nodal states $x_i(t)$, and red curves are the predictions of the corresponding nodes. For clarity, 12 randomly chosen nodal states are depicted.

other 5 hypergraphs are similar to the prediction on hypergraph *contact-high-school*. Thus, we demonstrate that the higher-order contagion process cannot be predicted with unknown topology by the method designed for networks, assuming only pairwise interactions between nodes. The higher-order interactions among nodes must be modelled and studied explicitly.

2) Comparing With Deep Learning Prediction Algorithm: We compared the performance of TaHiP to three latest Transformer-based models for multivariate time series prediction:

- **Zeng et al. [35] (LTSF, 2022)**: They extracted the temporal relations in an ordered set of continuous points, and introduced a simple one-layer linear model named LTSF-Linear for the long-term time series prediction task.
- **Nie et al. [36] (PatchTST, 2023)**: They proposed an efficient design of Transformer-based model for multivariate time series prediction, with segmentation of time series into subseries-level patches and channel-independence where each channel contains a single univariate time series that shares the same embedding and Transformer weights across all the series.
- **Yi et al. [37] (FreTS, 2023)**: They explored a novel direction of applying MLPs in the frequency domain for time series prediction. They investigated the learned patterns of frequency-domain MLPs, and proposed FreTS, an effective architecture built upon frequency-domain MLPs for time series prediction.

In this work, we set the parameters of each baseline method according to the setting on the ILI dataset (adopted by all the baseline models), which has a similar size (966 time steps) to the training matrix $\mathbf{X}_{\mathbf{train}} \in \mathbb{R}^{N\times T}$ used in TaHiP algorithm (1500 time steps). We set the number of input features in the baseline models according to the number of nodes N in each hypergraph. The prediction results on various hypergraphs of TaHiP and baseline models are presented in Table III.

Overall, the TaHiP algorithm outperforms all the baseline models by metrics of MSE and MAE in all the six real-world hypergraphs. Furthermore, TaHiP only needs an $N \times 1$ initial node state vector to start prediction, while the baseline models require continuous data input (called 'look-back window' [35] with length of L) to predict future data. Unlike the baseline models, which rely exclusively on training datasets and disregard the underlying process that generates the time series, we have successfully integrated the governing equations of a dynamical system into the prediction method in TaHiP, and achieved a better performance. This finding highlights the importance of combining the prior knowledge of a specific domain with the training data, and these methods with prior knowledge may outperform the data-driven methods developed for general prediction purpose.

We plot the prediction on hypergraph *contact-primary-school* in Fig. 4 to show the accuracy of the prediction. As Table III shows, the TaHiP algorithm predicts with extremely small MSE errors (around 10^{-4}) on hypergraphs *contact-high-school*, *contact-primary-school*, and *email-Eu*. The 3 hypergraphs have small mean hyperedge sizes of 2.3, 2.4, and 2.33, respectively, and high proportions of size-2 hyperedge ($\%|e_j| = 2 > 50\%$). In contrast, the hypergraphs *senate-bills* and *house-bills* have larger mean hyperedge sizes (9.9 and 22.1) and lower proportions of size-2 hyperedges ($\%|e_j| = 2 < 20\%$). To summarize, the TaHiP algorithm predicts with higher accuracy on hypergraphs with smaller mean hyperedge sizes and higher proportions of size-2 hyperedges. The mean hyperedge size is one of the important indicators of the complexity of the hypergraph topology, and hypergraphs with larger mean hyperedge size contain more hyperedges with higher-order. The existence of higher-order hyperedges, which contain far more nodes than 2 (the minimum size of a hyperedge), makes the accurate prediction based on observed dynamics among nodes more difficult.

To quantitatively study the impact of mean hyperedge size of a hypergraph on the prediction accuracy, we test the TaHiP algorithm in different synthetic hypergraphs in Section III-D.

D. Predicting Higher-Order Dynamics With Known Hyperedge Size Distribution

In the previous experiments, we predict the dynamics of hypergraphs with completely unknown topology. In this subsection, we conduct more experiments to explore the following questions:

TABLE III

PREDICTION RESULTS OF TAHIP AND BASELINE MODELS ON HYPERGRAPHS. THE BEST RESULTS OF MSE ARE IN **BOLD**. **IMP.** DENOTES THE IMPROVEMENT ON MSE OF TAHIP COMPARED TO THE CORRESPONDING METHOD, WITH **IMP.** $= 1 - \frac{MSE(TaHiP)}{MSE(Model)}$

Hypergraph	N	TaHiP		LTSF			PatchTST			FreTS		
		MSE	MAE	MSE	MAE	**IMP.**	MSE	MAE	**IMP.**	MSE	MAE	**IMP.**
contact-high-school	327	**0.00022**	0.0102	0.00644	0.0591	96.6%	0.00456	0.0454	95.2%	0.00612	0.053	96.4%
contact-primary-school	242	**0.00017**	0.0089	0.00556	0.0535	96.9%	0.00589	0.0524	97.1%	0.00516	0.0443	96.7%
email-Enron	148	**0.00806**	0.0631	0.01581	0.0928	49.0%	0.01143	0.0703	29.5%	0.01075	0.0751	25.0%
email-Eu	998	**0.00031**	0.0124	0.00887	0.0621	96.5%	0.00284	0.0265	89.1%	0.00224	0.0297	86.2%
senate-bills	294	**0.00227**	0.0215	0.0283	0.1285	92.0%	0.01046	0.0663	78.3%	0.01259	0.0686	82.0%
house-bills	1494	**0.00773**	0.0611	0.03021	0.1330	74.4%	0.02992	0.1201	74.2%	0.01888	0.0761	59.1%

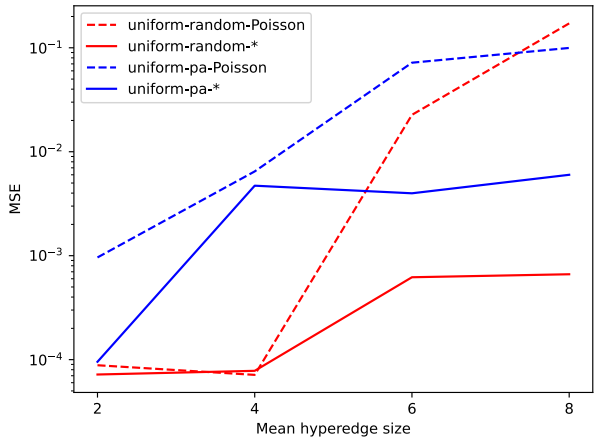

Fig. 5. Prediction errors measured by MSE of TaHiP on synthetic uniform hypergraphs with different mean hyperedge sizes. The label 'uniform-random' or 'uniform-pa' indicates the generation method of the hypergraph, while the suffix '-Poisson' or '-*' refers to the surrogate matrix initialization method of Poisson distribution or hyperedge size distribution, respectively.

- In real-world applications of the algorithm, limited structural information of the target hypergraph (for instance, the hyperedge size distribution) may be known, will the known hyperedge size distribution (the topology of the hypergraph is still unknown) improve the prediction accuracy of TaHiP?
- We have observed in Table III that the TaHiP algorithm predicts with higher accuracy on real-world hypergraphs with smaller mean hyperedge sizes, it is necessary to validate this finding in more synthetic hypergraphs.

We test the TaHiP algorithm in synthetic uniform hypergraphs generated by two distinct methods. The first method adds each hyperedge by randomly selecting m nodes from the hypergraph. The second method selects nodes to form hyperedges with the principle of preferential attachment. Specifically, the probability of a node i being selected is defined as

$$p(i) = \frac{d_H(i)}{\sum_{j=1}^{N} d_H(j)}. \tag{21}$$

In both methods, duplicate hyperedges are not allowed. We refer to the two classes of uniform hypergraph as *uniform-random* and *uniform-pa* ('pa' is short for preferential attachment), respectively. The sizes of all the uniform hypergraphs are set as $N = 500$ and $E = 10000$.

For each $m-$uniform hypergraph, we use two distinct methods of initialize the surrogate matrix. In the first case, we assume the hyperedge size distribution of the target hypergraph is known, and generate samples from this distribution, $\{m, m, \cdots, m\}$ to initialize the surrogate matrix, which guarantees that the surrogate matrix and the target hypergraph share the same hyperedge size distribution. In the second case, where the hyperedge size distribution of the target hypergraph is unknown, we use a Poisson distribution with $\lambda = 2$ for initialization, similar to the experiments in Section II-C.

We continue to use MSE as the measurement of prediction. The results are showed in Fig. 5. Each MSE is the mean value of 10 experiments. The method of hypergraph generation is labeled as 'uniform-random' or 'uniform-pa', as noted above. The suffix '-Poisson' or '-*' refers to the surrogate matrix initialization method of Poisson distribution or hyperedge size distribution, respectively.

First, in both classes of uniform hypergraphs with different mean hyperedge sizes, the knowledge of hyperedge size distribution reduces the prediction error MSE (comparing 'uniform-random-Poisson' with 'uniform-random-*', and 'uniform-pa-Poisson' with 'uniform-pa-*' in Fig. 5), especially in hypergraphs with larger mean hyperedge size ($< |e_j| >= 6$ or 8). This finding demonstrates that the hyperedge size distribution of the unknown hypergraph for initialization can improve the prediction accuracy in uniform hypergraphs. Second, the prediction error increases with mean hyperedge size of the hypergraph generally (except for the 'uniform-pa-*' case, where the hypergraph with $< |e_j| >= 6$ slightly outperforms hypergraph with $< |e_j| >= 4$), which aligns with the results of experiments in real-world hypergraphs (see Table III). We attribute this finding to the complicated topology of hypergraphs with higher mean hyperedge size, which adds to the difficulty of prediction.

To provide more evidence, we test the performances of the proposed TaHiP algorithm in real-world hypergraphs, given the hyperedge size distribution of each hypergraph. The results are compared with the predictions without hyperedge size distribution in Table IV, which shows that in all the hypergraphs except for hypergraph *email-Eu*, the knowledge of hyperedge size distribution reduces the prediction errors, especially in hypergraph *senate-bills* and hypergraph *house-bills*, where the improvements are 83.3% and 76.4%.

Thus, the results in both synthetic hypergraphs and real-world hypergraphs demonstrate that, the knowledge of hyperedge size distribution of the unknown hypergraph can

TABLE IV
PREDICTION RESULTS OF THE PROPOSED TAHIP ALGORITHM WITHOUT/WITH HYPEREDGE SIZE DISTRIBUTION IN REAL-WORLD HYPERGRAPHS. THE BESTRESULTS OF MSE ARE IN **BOLD**. **IMP.** DENOTES THE IMPROVEMENT ON MSE OF TAHIP W/E COMPARED TO TAHIP, WITH **IMP.** $= 1 - \frac{MSE(TaHiPw/e)}{MSE(TaHiP)}$

Hypergraph	N	TaHiP		TaHiP w/e[1]		**IMP.**
		MSE	MAE	MSE	MAE	
contact-high-school	327	0.00022	0.0102	**0.00019**	0.0093	13.6%
contact-primary-school	242	0.00017	0.0089	**0.00010**	0.0068	41.2%
email-Enron	148	0.00806	0.0636	**0.00673**	0.0629	16.5%
email-Eu	998	0.00031	0.0124	0.00031	0.0122	0%
senate-bills	294	0.00227	0.0215	**0.00038**	0.0124	83.3%
house-bills	1494	0.01594	0.0546	**0.00376**	0.0394	76.4%

[1] w/e denotes 'with hyperedge size distribution'.

improve the prediction accuracy, and the prediction error of the TaHiP algorithm increases with mean hyperedge size of the hypergraph.

IV. TOPOLOGY OF THE SURROGATE MATRIX

In Section II-C, we provided detailed description of how the proposed TaHiP algorithm predicts the dynamics of an unknown hypergraph accurately by optimizing a surrogate matrix, $\hat{\mathbf{B}}$. Does the prediction accuracy imply a similarity of the surrogate matrix with the true incidence matrix of the hypergraph?

We here denote the incidence matrix of any hypergraph by $\mathbf{B}$. The hyperdegree of any node i in this hypergraph can be computed by summing all the elements of the ith row in matrix $\mathbf{B}$ (see **Definition 1**).

Definition 4 Predicted Hyperdegree (PHD): Similar to hyperdegree, we can sum all the elements of the ith row in matrix $\hat{\mathbf{B}}$, and refer to this sum as the predicted hyperdegree of node i, denoted as $\hat{d_H}(i)$

$$\hat{d_H}(i) = \sum_{j=1}^{E} \hat{B}_{ij}. \tag{22}$$

To quantitatively study the similarity between the surrogate matrix $\hat{\mathbf{B}}$ and the incidence matrix $\mathbf{B}$, we introduce two node centrality metrics, including the higher-order H-index (HOH) [22], and higher-order PageRank (HOP) [22]. The H-index [53] is originally used to measure the citation impact of a scholar or a journal, and is adopted in [54] as a node centrality metric in networks. In this work, we use an extension of H-index for hypergraphs defined in [22].

Definition 5 Higher-Order H-index (HOH): The HOH of a node i is the maximum value H, such that there exists at least H neighbors of the node i with hyperdegrees no less than H. Here the neighbor of node i is defined as the node that belongs to the same hyperedge with node i.

Definition 6 Higher-Order PageRank (HOP): Similarly to PageRank [55], the element of the transition matrix $\mathbf{P}$ of a hypergraph is defined as

$$P_{ij} = \begin{cases} \frac{(A_H)_{ij}}{\sum_i (A_H)_{ij}}, & \text{if } i \neq j, \\ 0, & \text{if } i = j. \end{cases}$$

where $\mathbf{A_H}$ is the adjacency matrix of hypergraph defined in the Appendix A. The stationary distribution $\mathbf{P}^\infty$ is defined by

$$\mathbf{P}^\infty = \lim_{t\to\infty} \mathbf{P}^t.$$

The basic PageRank at time t is defined by

$$pr(t) = \mathbf{P}^T pr(t-1) \quad \in \mathbb{R}^N, \tag{23}$$

where $pr_i(t)$ is the PageRank value of the ith node in the hypergraph. Considering that there might be a group of interconnected nodes, the PageRank values remain the same within the group and will not be changed. A damping factor s is introduced to avoid this case, and we obtain a revised PageRank. The steady-state value of the revised PageRank for each node is defined by

$$pr(t) = s(\mathbf{P}^\infty)^T pr(0) + \frac{u(1-s)}{N}, \tag{24}$$

where $u \in \mathbb{R}^N$ is a vector with every entry equal to 1, and each element of $pr(0)$ is $1/N$, $s \in (0, 1)$ is the damping factor.

We calculate the Pearson correlation coefficients between each pair of metrics, including the hyperdegree (HD), the predicted hyperdegree (PHD), the higher-order H-index (HOH) and the higher-order PageRank (HOP) in the 6 real-world hypergraphs. The Pearson correlation coefficient reflects the linear correlation between variables, and has value between -1 and 1. The results are provided in Fig. 6. In all the real-world hypergraphs in Fig. 6, the Pearson correlation coefficients between HOH and the other three metrics are relatively low, revealing its different nature of defining influential nodes from the other metrics. On the other hand, the Pearson correlation coefficients between HD and HOP are close to 1 in all the hypergraphs, reflecting the strong correlation between these 2 metrics.

The PHD metric is computed from the surrogate matrix generated during the prediction for each hypergraph. In hypergraph *contact-high-school*, *contact-primary-school*, *email-Eu* and *senate-bills*, where the prediction is accurate (with MSE$<$ 0.003, see Table IV), the Pearson correlation coefficients between HD and PHD, HOP and PHD (see Fig. 6(a),(b),(d),(e)) are close to 1, and larger than the corresponding Pearson correlation coefficients in hypergraph *email-Enron* and *house-bills* (see Fig. 6(c),(f)), where the prediction is inaccurate. The results in Fig. 6 demonstrates that, if the prediction of the unknown hypergraph dynamics

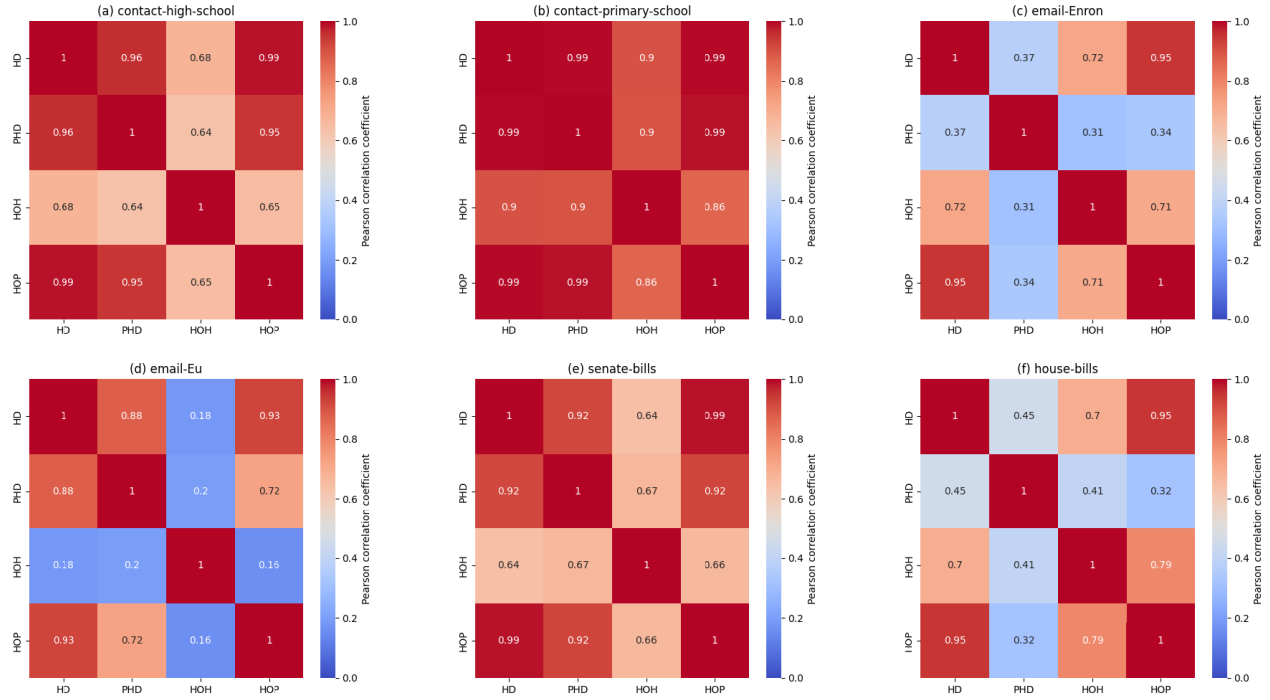

Fig. 6. Pearson correlation coefficients between each pair of metrics, including the hyperdegree (HD), the predicted hyperdegree (PHD), the higher-order H-index (HOH) and the higher-order PageRank (HOP) in the 6 real-world hypergraphs.

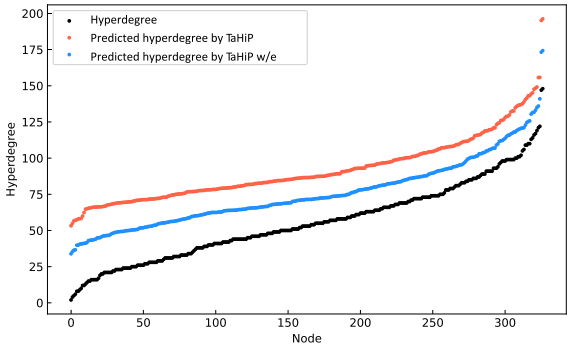

Fig. 7. Hyperdegree distribution and the predicted hyperdegree distributions of hypergraph *contact-high-school*. The predicted hyperdegree distributions are obtained by TaHiP without/with hyperedge size distribution, denoted as 'TaHiP' and 'TaHiP w/e', respectively.

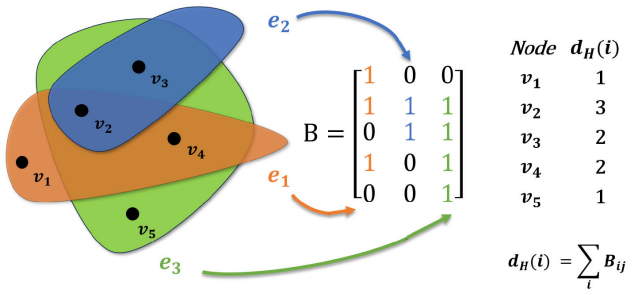

Fig. 8. A hypergraph of 5 nodes and 3 hyperedges and its corresponding 5×3 incidence matrix $\mathbf{B}$. The hyperdegree of a node i, denoted as $d_H(i)$, is defined in Section III-A.

is accurate, then we can obtain the predicted hyperdegree of that hypergraph, which is highly correlated with the true hyperdegree of each node in the hypergraph.

For instance, we plot the hyperdegree distribution and the predicted hyperdegree distribution of the hypergraph *contact-high-school* in Fig. 7. Each dot in Fig. 7 is the hyperdegree or predicted hyperdegree of a node, and the distributions are obtained by sorting the hyperdegrees of all the nodes by value. There is a remarkable similarity across the distributions illustrated in Fig. 7, which shows that the predicted hyperdegree can be regarded as the inference of the hyperdegree of a node, when the prediction of TaHiP is accurate.

V. Conclusion and Future Directions

The prediction of contagion dynamics on unknown hypergraphs is studied, based on observations of the dynamics. We propose a prediction framework which consists of two steps. First, we obtain a surrogate matrix by fitting the dynamical model to the observations of a contagion process on a hypergraph with unknown topology. Second, we predict the dynamics of any contagion process on the same hypergraph using the surrogate matrix, given initial values of all the nodes. The proposed TaHiP algorithm outperforms three Transformer-based deep learning models [35], [36], [37] in different real-world hypergraphs, and requires less data input, training time and storage space. Moreover, experiments in synthetic and real-world hypergraphs show that the prediction accuracy of TaHiP can be improved, if the hyperedge size distribution of the hypergraph is used to initialize the surrogate matrix. Furthermore, we studied the surrogate matrix obtained by TaHiP, and found that when the prediction is accurate, the Pearson correlation coefficient between hyperdegree and predicted hyperdegree defined in this work is close to 1, which shows that the hyperdegree distribution of the unknown

hypergraph could be inferred by calculating the predicted hyperdegree distribution from the surrogate matrix.

In terms of future directions, first, it is an open question whether our prediction method applies to other higher-order dynamics on hypergraphs, including diffusion, synchronization, etc. Theoretically, this method works if the incidence matrix of hypergraph can be embedded into the equation of the dynamical process. Second, the proposed TaHiP algorithm predicts with low accuracy on hypergraphs with higher mean hyperedge size ($\langle|e_j|\rangle > 10$), how to improve the performance of TaHiP on hypergraphs with higher mean hyperedge size needs further works.

APPENDIX A
HYPERGRAPH AND INCIDENCE MATRIX

Fig. 8 shows a hypergraph of 5 nodes and 3 hyperedges and its corresponding incidence matrix. Note that the size of the hyperedge e_2 that contains node v_2 and v_3 is 2, which means this hyperedge models pairwise interaction between nodes, similar to an edge in a network. Moreover, a hyperedge is not equivalent to a complete graph containing the same number of nodes, as higher-order interactions cannot be accurately modelled by linear combinations of pairwise interactions [20].

For an undirected graph $\mathcal{G}$ of N nodes and L edges, the unsigned incidence matrix $\mathbf{R} \in \mathbb{R}^{N\times L}$ is defined as

$$R_{il} = \begin{cases} 1, & \text{if node } i \text{ and node } j \text{ is linked by edge } e_l, \\ 0, & \text{otherwise.} \end{cases}$$

The $N \times N$ adjacency matrix $\mathbf{A}$ of the graph G can be written [41] in terms of the unsigned incidence matrix $\mathbf{R}$ as

$$\mathbf{A} = \mathbf{R}\mathbf{R}^T - \Delta, \tag{25}$$

where $\Delta = diag(d_1,\ d_2\ \cdots,\ d_N)$ is the degree matrix.

Analogous the definition of adjacency matrix of graph, the adjacency matrix of a hypergraph, $\mathcal{H}$ with N nodes and E hyperedges can be defined as

$$(\mathbf{A_H})_{ij} = \begin{cases} e_s, & \text{if node } i \text{ and node } j \text{ share } e_S \text{ hyperedges,} \\ 0, & \text{otherwise.} \end{cases} \tag{26}$$

Similarly, we have

$$\mathbf{A_H} = \mathbf{B}\mathbf{B}^T - \Delta_\mathbf{H}, \tag{27}$$

where $\mathbf{B}$ is the incidence matrix of hypergraph $\mathcal{H}$, as we have defined in Section II-A, and $\Delta_\mathbf{H} = diag(d_H(1),\ d_H(2)\ \cdots,\ d_H(N))$ is the hyperdegree matrix, where $d_H(i)$ is the hyperdegree of node i.

We also find that in the $E \times E$ matrix

$$\mathbf{M} = \mathbf{B}^T\mathbf{B}, \tag{28}$$

the element M_{ij} is the number of nodes that belong to hyperedge i and hyperedge j simultaneously, and each diagonal element M_{ii} is the size of each hyperedge i.

APPENDIX B
CONVERGENCE ANALYSIS OF TAHIP

We analyze the convergence of TaHiP using the framework proposed in [42]. First, the pseudocode of Adam [42] is provided in **Algorithm 2**.

Algorithm 2 Adam.

Input:
Stepsize α, with $\beta_1,\ \beta_2 \in [0, 1)$ being exponential decay rates for the moment estimates. Stochastic objective function $f(\theta)$, with initial parameter vector being θ_0.

Output:
Resulting parameters θ_t.

$m_0 \leftarrow 0$ (Initialize 1^{st} moment vector)
$v_0 \leftarrow 0$ (Initialize 2^{nd} moment vector)
$t \leftarrow 0$ (Initialize timestep)
while θ_t not converged **do**
  $t \leftarrow t + 1$
  $g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$ (Get gradients w.r.t. objective at t)
  $m_t \leftarrow \beta_1 m_{t-1} + (1 - \beta_1)g_t$ (Update biased first moment)
  $v_t \leftarrow \beta_2 v_{t-1} + (1 - \beta_2)g_t^2$ (Update biased second raw moment)
  $\hat{m}_t \leftarrow m_t(1 - \beta_1^t)$ (Bias-corrected first moment)
  $\hat{v}_t \leftarrow v_t(1 - \beta_2^t)$ (Bias-corrected second raw moment)
  $\theta_t \leftarrow \theta_{t-1} - \alpha \cdot \hat{m}_t/(\sqrt{\hat{v}_t} + \epsilon)$ (Update parameters)
end while

Given an arbitrary, unknown sequence of ***convex*** cost functions $f_1(\theta), f_2(\theta), \cdots, f_t(\theta)$, the goal of Adam is to predict parameter θ_t, which is then evaluated on f_t. The regret $R(T)$ is used to prove the convergence of an optimization algorithm. $R(T)$ is the sum of all the previous differences between the prediction $f_t(\theta_t)$ and the best parameter $f_t(\theta^*)$ from a feasible set $\mathcal{X}$ for all the previous steps. Specifically, $R(T)$ is defined as

$$R(T) = \sum_{t=1}^{T}[f_t(\theta_t) - f_t(\theta^*)]\ , \tag{29}$$

where $\theta^* = argmin_{\theta\in\mathcal{X}} \sum_{t=1}^{T} f_t(\theta)$. The convergence of an algorithm is ensured if the average regret, i.e., $R(T)/T$ of the algorithm converges.

Theorem 1 (Corollary 4.2. in [42]): Assume the function f_t has bounded gradients, $||\nabla f_t(\theta)||_2 \le G, ||\nabla f_t(\theta)||_\infty \le G_\infty$ for all $\theta \in R^d$ and the distance between any θ_t generated by Adam is bounded, $||\theta_n - \theta_m||_2 \le D, ||\theta_n - \theta_m||_\infty \le D_\infty$ for any $m, n \in \{1,\ 2, \cdots,\ T\}$. Adam achieves the following guarantee, for all $T \ge 1$.

$$\frac{R(T)}{T} = O(\frac{1}{\sqrt{T}}),\ lim_{T\to\infty}\frac{R(T)}{T} = 0. \tag{30}$$

In TaHiP, first, we prove that the cost function of mean squared error, namely, $f(\mathbf{x}) = \frac{1}{N}\sum_{i=2}^{k}(\mathbf{x}_i - \tilde{\mathbf{x}}_i)^2$ (See Section II-C 2) and Fig. 1(b)) is convex.

Lemma 1 (Affine Transformations Preserve Convexity [56]): If f is convex, then $g(\mathbf{x}) = f(\mathbf{A}\mathbf{x} + \mathbf{b})$ is also convex.

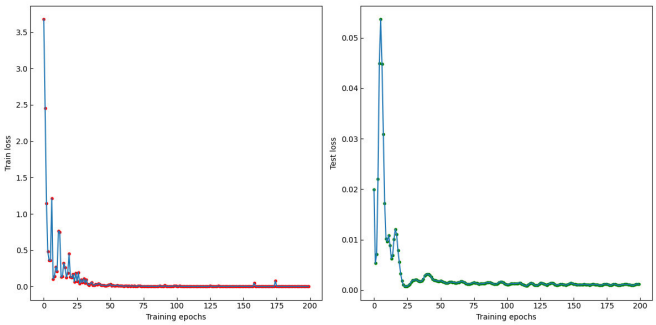

Fig. 9. Train loss and test loss vs training epochs of TaHiP applied to hypergraph *contact-high-school*.

Lemma 2 (Non-Negative Weighted Sum Preserve Convexity [56]): If $f_1,\ f_2,\ \cdots,\ f_k$ *are convex, then* $f = \beta_1 f_1 + \beta_2 f_2 + \cdots + \beta_k f_k$ *is also convex, when* $\beta_1,\ \beta_2,\ \cdots,\ \beta_k \geq 0$.

As $\frac{1}{N}\sum_{i=2}^{k} ||\mathbf{x}||^2$is convex, the MSE $f(\mathbf{x}) = \frac{1}{N}\sum_{i=2}^{k}(\mathbf{x}_i - \tilde{\mathbf{x}}_i)^2$ is convex by **Lemma 1** and **Lemma 2**, which satisfy the requirement of Adam that the cost function is convex. Second, we use the gradient norm clipping method to ensure that the gradients during the training process are bounded, which satisfy the conditions of **Theorem 1**. Therefore, the convergence of TaHiP is guaranteed. We plot the curves of train loss and test loss during the training processes of TaHiP in Fig. 9.

Fig. 9 shows that, during the training process of TaHiP, the test loss decreased simultaneously with the train loss and converged quickly, proving the effectiveness of the algorithm. The training process of TaHiP being applied to other hypergraphs is similar.

APPENDIX C
COMPLEXITY ANALYSIS OF TAHIP

The time complexity of TaHiP could be estimated as

$$\begin{aligned} TC_{train}(N,E) &= O(k \cdot M \cdot batch\_size \cdot (2N + 3N \cdot E)) \\ &= O(N \cdot E), \\ TC_{prediction}(N,E) &= O(T \cdot N \cdot E) = O(N \cdot E), \end{aligned}$$

where TC_{train} is the time complexity of the training part of TaHiP, $TC_{prediction}$ is the time complexity of the prediction part, the random window length $k = 100$, the number of training epochs $M = 1000$, the size of a batch of training samples $batch\_size = 4$, prediction time steps $T = 1500$, the number of nodes N and the number of hyperedges E vary with specific hypergraph. The training time of TaHiP increases linearly with the number of nodes of the hypergraph for prediction.

The storage space of TaHiP is

$$Space(train) \approx batch\_size \cdot N \cdot [4 \cdot E \cdot (k+1) + 1] \tag{31}$$

$$Space(prediction) \approx N \cdot E \cdot (T+1) \tag{32}$$

where $batch\_size = 4$, $k = 100$, $T = 1500$, the number of nodes N and the number of hyperedges E vary with specific hypergraph. The space complexity of TaHiP is $O(N \cdot E)$, which increases linearly with the number of nodes of the hypergraph. We attribute the cost-effectiveness of TaHiP to the limited size of training parameters (the elements of an $N \times E$ matrix).

REFERENCES

[1] M. Newman, A.-L. Barabási, and D. J. Watts, *The Structure and Dynamics of Networks*. Princeton, NJ, USA: Princeton Univ. Press, 2011.
[2] X. Fan Wang and G. Chen, "Complex networks: Small-world, scale-free and beyond," *IEEE Circuits Syst. Mag.*, vol. 3, no. 1, pp. 6–20, Sep. 2003.
[3] J. Song, E. Cotilla-Sanchez, G. Ghanavati, and P. D. H. Hines, "Dynamic modeling of cascading failure in power systems," *IEEE Trans. Power Syst.*, vol. 31, no. 3, pp. 2085–2095, May 2016.
[4] R. Pastor-Satorras, C. Castellano, P. Van Mieghem, and A. Vespignani, "Epidemic processes in complex networks," *Rev. Mod. Phys.*, vol. 87, no. 3, pp. 925–979, Aug. 2015.
[5] A. Pikovsky, M. Rosenblum, and J. Kurths, *Synchronization: A Universal Concept in Nonlinear Sciences*. Cambridge, U.K.: Cambridge Univ. Press, 2001.
[6] J. Wu and X. Li, "Collective synchronization of Kuramoto-oscillator networks," *IEEE Circuits Syst. Mag.*, vol. 20, no. 3, pp. 46–67, Aug. 2020.
[7] X. Wang, X. Li, and J. Lu, "Control and flocking of networked systems via pinning," *IEEE Circuits Syst. Mag.*, vol. 10, no. 3, pp. 83–91, 3rd Quart., 2010.
[8] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, "Complex networks: Structure and dynamics," *Phys. Rep.*, vol. 424, nos. 4–5, pp. 175–308, 2006.
[9] X. Li, X. Wang, and G. Chen, "Pinning a complex dynamical network to its equilibrium," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 10, pp. 2074–2087, Oct. 2004.
[10] I. Mishkovski, M. Mirchev, S. Šcepanovic, and L. Kocarev, "Interplay between spreading and random walk processes in multiplex networks," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 64, no. 10, pp. 2761–2771, Oct. 2017.
[11] R. A. Shuvro, P. Das, M. M. Hayat, and M. Talukder, "Predicting cascading failures in power grids using machine learning algorithms," in *Proc. North Amer. Power Symp. (NAPS)*, Oct. 2019, pp. 1–6.
[12] M. Timme and J. Casadiego, "Revealing networks from dynamics: An introduction," *J. Phys. A, Math. Theor.*, vol. 47, no. 34, Aug. 2014, Art. no. 343001.
[13] B. Barzel and A.-L. Barabási, "Universality in network dynamics," *Nature Phys.*, vol. 9, no. 10, pp. 673–681, Oct. 2013.
[14] T.-T. Gao and G. Yan, "Autonomous inference of complex network dynamics from incomplete and noisy data," *Nature Comput. Sci.*, vol. 2, no. 3, pp. 160–168, Mar. 2022.
[15] J. Liu, G. Mei, X. Wu, and J. Lü, "Robust reconstruction of continuously time-varying topologies of weighted networks," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 65, no. 9, pp. 2970–2982, Sep. 2018.
[16] B. Prasse and P. Van Mieghem, "Predicting network dynamics without requiring the knowledge of the interaction graph," *Proc. Nat. Acad. Sci. USA*, vol. 119, no. 44, Nov. 2022, Art. no. e2205517119.
[17] R. MacArthur, "Species packing and competitive equilibrium for many species," *Theor. Population Biol.*, vol. 1, no. 1, pp. 1–11, May 1970.
[18] Y. Kuramoto and Y. Kuramoto, *Chemical Turbulence*. Cham, Switzerland: Springer, 1984.
[19] G. Kerschen, J.-C. Golinval, A. F. Vakakis, and L. A. Bergman, "The method of proper orthogonal decomposition for dynamical characterization and order reduction of mechanical systems: An overview," *Nonlinear Dyn.*, vol. 41, nos. 1–3, pp. 147–169, Aug. 2005.
[20] F. Battiston et al., "Networks beyond pairwise interactions: Structure and dynamics," *Phys. Rep.*, vol. 874, pp. 1–92, Aug. 2020.
[21] F. Battiston et al., "The physics of higher-order interactions in complex systems," *Nat. Phys.*, vol. 17, no. 10, pp. 1093–1098, 2021.
[22] Y. Zhao, C. Li, D. Shi, G. Chen, and X. Li, "Ranking cliques in higher-order complex networks," *Chaos, Interdiscipl. J. Nonlinear Sci.*, vol. 33, no. 7, Jul. 2023, Art. no. 073139.
[23] Y. Zhu, C. Li, and X. Li, "Epidemic spreading on coupling network with higher-order information layer," *New J. Phys.*, vol. 25, no. 11, Nov. 2023, Art. no. 113043, doi: 10.1088/1367-2630/ad0920.

[24] Y. Zhang, J. Li, J. Ding, and X. Li, "A graph transformer-driven approach for network robustness learning," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 71, no. 5, pp. 1992–2005, May 2024.

[25] A. Ghasemi and H. Kantz, "Higher-order interaction learning of line failure cascading in power networks," *Chaos, Interdiscipl. J. Nonlinear Sci.*, vol. 32, no. 7, Jul. 2022, Art. no. 073101.

[26] M. T. Schaub, A. R. Benson, P. Horn, G. Lippner, and A. Jadbabaie, "Random walks on simplicial complexes and the normalized Hodge 1-Laplacian," *SIAM Rev.*, vol. 62, no. 2, pp. 353–391, Jan. 2020.

[27] A. P. Millán, J. J. Torres, and G. Bianconi, "Synchronization in network geometries with finite spectral dimension," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 99, no. 2, Feb. 2019, Art. no. 022307.

[28] I. Iacopini, G. Petri, A. Barrat, and V. Latora, "Simplicial models of social contagion," *Nat. Commun.*, vol. 10, no. 1, p. 2485, 2019.

[29] W. Zhang et al., "Experimental observation of higher-order topological Anderson insulators," *Phys. Rev. Lett.*, vol. 126, no. 14, Apr. 2021, Art. no. 146802.

[30] Y. Wang, H. M. Price, B. Zhang, and Y. D. Chong, "Circuit implementation of a four-dimensional topological insulator," *Nature Commun.*, vol. 11, no. 1, p. 2356, May 2020.

[31] M. Ouyang, M. Toulouse, K. Thulasiraman, F. Glover, and J. S. Deogun, "Multilevel cooperative search for the circuit/hypergraph partitioning problem," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 21, no. 6, pp. 685–693, Jun. 2002.

[32] B. Wang et al., "LHNN: Lattice hypergraph neural network for VLSI congestion prediction," in *Proc. 59th ACM/IEEE Design Autom. Conf.*, Jan. 2022, pp. 1297–1302.

[33] W.-K. Mak and D. F. Wong, "A fast hypergraph min-cut algorithm for circuit partitioning," *Integration*, vol. 30, no. 1, pp. 1–11, Nov. 2000.

[34] C. Fan et al., "Multi-horizon time series forecasting with temporal attention learning," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2019, pp. 2527–2535.

[35] A. Zeng, M. Chen, L. Zhang, and Q. Xu, "Are transformers effective for time series forecasting?" in *Proc. AAAI Conf. Artif. Intell.*, vol. 37, Jun. 2023, pp. 11121–11128.

[36] Y. Nie, N. H. Nguyen, P. Sinthong, and J. Kalagnanam, "A time series is worth 64 words: Long-term forecasting with transformers," in *Proc. Int. Conf. Learn. Represent.*, 2023, pp. 1–24.

[37] K. Yi et al., "Frequency-domain MLPs are more effective learners in time series forecasting," in *Proc. 37th Conf. Neural Inf. Process. Syst.*, Jan. 2023, pp. 76656–76679.

[38] B. Lim and S. Zohren, "Time-series forecasting with deep learning: A survey," *Phil. Trans. Roy. Soc. A*, vol. 379, no. 2194, Feb. 2021, Art. no. 20200209.

[39] Z. Zhou, C. Li, B. Qu, and X. Li, "Predicting higher-order dynamics without network topology by ridge regression," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2024, pp. 1–5.

[40] G. F. de Arruda, M. Tizzani, and Y. Moreno, "Phase transitions and stability of dynamical processes on hypergraphs," *Commun. Phys.*, vol. 4, no. 1, p. 24, Feb. 2021.

[41] P. Van Mieghem, *Graph Spectra for Complex Networks*, 2nd ed., Cambridge, U.K.: Cambridge Univ. Press, 2023.

[42] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[43] M. Contisciani, F. Battiston, and C. De Bacco, "Inference of hyperedges and overlapping communities in hypergraphs," *Nature Commun.*, vol. 13, no. 1, p. 7229, Nov. 2022.

[44] F. Musciotto, F. Battiston, and R. N. Mantegna, "Detecting informative higher-order interactions in statistically validated hypergraphs," *Commun. Phys.*, vol. 4, no. 1, p. 218, Sep. 2021.

[45] H. Wang, C. Ma, H.-S. Chen, Y.-C. Lai, and H.-F. Zhang, "Full reconstruction of simplicial complexes from binary contagion and ising data," *Nature Commun.*, vol. 13, no. 1, p. 3043, Jun. 2022.

[46] L. Pan, H.-J. Shang, P. Li, H. Dai, W. Wang, and L. Tian, "Predicting hyperlinks via hypernetwork loop structure," *EPL (Europhys. Lett.)*, vol. 135, no. 4, p. 48005, Aug. 2021.

[47] C. Chen and Y.-Y. Liu, "A survey on hyperlink prediction," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 11, pp. 15034–15050, 2023.

[48] M. Zhang, Z. Cui, S. Jiang, and Y. Chen, "Beyond link prediction: Predicting hyperlinks in adjacency space," in *Proc. AAAI Conf. Artif. Intell.*, Apr. 2018, vol. 32, no. 1, pp. 4430–4437.

[49] A. R. Benson, R. Abebe, M. T. Schaub, A. Jadbabaie, and J. Kleinberg, "Simplicial closure and higher-order link prediction," *Proc. Nat. Acad. Sci.*, vol. 115, no. 48, pp. E11221–E11230, Nov. 2018.

[50] S. Hu, X. Wu, and T.-H.-H. Chan, "Maintaining densest subsets efficiently in evolving hypergraphs," in *Proc. ACM Conf. Inf. Knowl. Manage.*, Nov. 2017, pp. 929–938.

[51] E. Ganmor, R. Segev, and E. Schneidman, "Sparse low-order interaction network underlies a highly correlated and learnable neural population code," *Proc. Nat. Acad. Sci. USA*, vol. 108, no. 23, pp. 9679–9684, Jun. 2011.

[52] L. Merchan and I. Nemenman, "On the sufficiency of pairwise interactions in maximum entropy models of networks," *J. Stat. Phys.*, vol. 162, no. 5, pp. 1294–1308, Mar. 2016.

[53] J. E. Hirsch, "An index to quantify an individual's scientific research output," *Proc. Nat. Acad. Sci. USA*, vol. 102, no. 46, pp. 16569–16572, Nov. 2005.

[54] L. Lü, T. Zhou, Q.-M. Zhang, and H. E. Stanley, "The H-index of a network node and its relation to degree and coreness," *Nature Commun.*, vol. 7, no. 1, p. 10168, Jan. 2016.

[55] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Comput. Netw. ISDN Syst.*, vol. 30, nos. 1–7, pp. 107–117, Apr. 1998.

[56] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[57] A. Barrat, M. Barthelemy, and A. Vespignani, *Dynamical Processes on Complex Networks*. Cambridge, U.K.: Cambridge Univ. Press, 2008.

[58] M. Morrison and J. N. Kutz, "Nonlinear control of networked dynamical systems," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 1, pp. 174–189, Jan. 2021.

[59] T. Carletti, F. Battiston, G. Cencetti, and D. Fanelli, "Random walks on hypergraphs," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 101, no. 2, Feb. 2020, Art. no. 022308.

[60] Q. Chen, Y. Zhao, C. Li, and X. Li, "Robustness of higher-order networks with synergistic protection," *New J. Phys.*, vol. 25, no. 11, Nov. 2023, Art. no. 113045.

[61] P. S. Skardal and A. Arenas, "Abrupt desynchronization and extensive multistability in globally coupled oscillator simplexes," *Phys. Rev. Lett.*, vol. 122, no. 24, Jun. 2019, Art. no. 248301.

[62] G. F. de Arruda, G. Petri, and Y. Moreno, "Social contagion models on hypergraphs," *Phys. Rev. Res.*, vol. 2, no. 2, Apr. 2020, Art. no. 023032.

Zili Zhou received the B.S. degree in electrical engineering from Fudan University, Shanghai, China, in 2021, where he is currently pursuing the master's degree with the School of Information Science and Technology. His research interests include higher-order dynamics and network dynamics prediction.

Cong Li (Member, IEEE) received the Ph.D. degree in intelligent systems from Delft University of Technology (TU Delft), Delft, The Netherlands, in 2014. She is currently an Associate Professor with the Electronic Engineering Department, Fudan University, where she involves in complex network theory and applications. Her research interests include analysis and modeling of complex networks, including network properties, dynamic processes, and network of networks. Her work on these subjects include 40 international journal articles and conference papers.

Piet Van Mieghem (Fellow, IEEE) received the master's and Ph.D. degrees in electrical engineering from KU Leuven, Belgium, in 1987 and 1991, respectively. He has been a Professor with Delft University of Technology and the Chairperson of the Section Network Architectures and Services (NAS) since 1998. Before joining Delft, he was with the Interuniversity Microelectronics Centre (IMEC) from 1987 to 1991. From 1993 to 1998, he was a member of the Alcatel Corporate Research Center, Antwerp. He was a Visiting Scientist at MIT, from 1992 to 1993; and a Visiting Professor at UCLA in 2005, Cornell University in 2009, Stanford University in 2015, and Princeton University in 2022. He is the author of four books: *Performance Analysis of Communications Networks and Systems*, *Data Communications and Networking*, *Graph Spectra for Complex Networks*, and *Performance Analysis of Complex Networks and Systems*. He is a Board Member of The Netherlands Platform Complex Systems, a Steering Committee Member of the Dutch Network Science Society, an External Faculty Member of the Institute for Advanced Study (IAS), University of Amsterdam. He was awarded an Advanced ERC grant 2020 for ViSiON and Virus Spread in Networks. He serves on the editorial board for the OUP Journal of Complex Networks. He was member of the editorial board of *Computer Networks* from 2005 to 2006, the IEEE/ACM TRANSACTIONS ON NETWORKING from 2008 to 2012, the *Journal of Discrete Mathematics* from 2012 to 2014, and *Computer Communications* from 2012 to 2015.

Xiang Li (Senior Member, IEEE) received the B.S. and Ph.D. degrees in control theory and control engineering from Nankai University, China, in 1997 and 2002, respectively. He was with the City University of Hong Kong, University Bremen, Shanghai Jiao Tong University, and Fudan University, as a Post-Doctoral Research Fellow, a Humboldt Research Fellow, an Associate Professor, a Professor/Distinguished Professor in 2002–2004, 2005–2006, 2004–2007, and 2008–2021, respectively. He is currently a Distinguished Professor with Tongji University, Shanghai, China, and the Founding Director of the Institute of Complex Networks and Intelligent Systems, Shanghai Research Institute for Intelligent Autonomous Systems, Tongji University. He has (co-)authored six research monographs, seven book chapters, and more than 150 peer-refereed journal publications and more than 100 indexed conference papers. His research interests include network science and systems control in both theory and applications. He received the IEEE Guillemin-Cauer Best Paper Award from the IEEE Circuits and Systems Society in 2005, Shanghai Natural Science Award (First Class) in 2008, Shanghai Science and Technology Young Talents Award in 2010, the National Science Foundation for Distinguished Young Scholar of China in 2014, the National Natural Science Award of China (Second Class) in 2015, the Ten Thousand Talent Program of China in 2017, the TCCT CHEN Han-Fu Award of Chinese Automation Association in 2019, the Excellent Editor Award of the IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING in 2021, and among other awards and honors. He served/serves as an Associate Editor for IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS from 2010 to 2015, a Researcher for the *Journal of Complex Networks* and the IEEE Circuits and Systems Society Newsletter, and an Associate Editor (2018–2021) and the Area Editor (since 2022) for IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING.