



Delft University of Technology

Document Version

Final published version

Licence

Dutch Copyright Act (Article 25fa)

Citation (APA)

Leus, G., Han, Y., Isufi, E., & Marques, A. G. (2025). Directed Acyclic Hypergraphs: Topology Identification from Nodal Data. In M. B. Matthews (Ed.), *Proceedings of the 2025 59th Asilomar Conference on Signals, Systems, and Computers* (pp. 176-181). IEEE. <https://doi.org/10.1109/IEEECONF67917.2025.11443873>

Important note

To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright

In case the licence states "Dutch Copyright Act (Article 25fa)", this publication was made available Green Open Access via the TU Delft Institutional Repository pursuant to Dutch Copyright Act (Article 25fa, the Taverne amendment). This provision does not affect copyright ownership.
Unless copyright is transferred by contract or statute, it remains with the copyright holder.

Sharing and reuse

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.

Directed Acyclic Hypergraphs: Topology Identification from Nodal Data

Geert Leus[†], Yongsheng Han[†], Elvin Isufi[†], and Antonio G. Marques^{*}

Abstract—Inferring higher-order network structures from nodal data is an emerging challenge across fields such as signal processing, machine learning, and causal inference. While directed acyclic graphs (DAGs) provide a powerful framework for modeling causal or functional dependencies, they only capture pairwise interactions. This paper introduces a new directed acyclic hypergraph (DAH) signal model that generalizes DAG-based structural equation modeling to multi-node (higher-order) relationships. Our approach begins by lifting a directed hypergraph (DH) into an equivalent bipartite directed graph (DG), where virtual nodes represent source-node sets of hyperedges. Nodal data are assigned to both original and virtual nodes, and an SEM is defined over the DG. By imposing a smooth acyclicity constraint on this bipartite graph, we obtain a continuous and scalable formulation for DAH estimation from nodal observations. The proposed framework unifies hypergraph learning and DAG inference under a common optimization perspective, enabling interpretable higher-order dependency discovery. Numerical experiments demonstrate the ability of the method to recover meaningful DAH structures from simulated nodal data.

Index Terms—Hypergraphs, directed acyclic graph (DAG), graph learning, hyperedge identification

I. INTRODUCTION

Inferring the structure of a directed acyclic graph (DAG) from nodal data is a fundamental problem not only in data science disciplines such as machine learning (ML), signal processing (SP), and causal inference, but also in fields such as biology, genetics, epidemiology, and transportation. Learning a DAG from data is typically a combinatorial problem; however, recent advances have reformulated it as a continuous optimization task by introducing a smooth *acyclicity* constraint [1]. This continuous characterization enables the imposition of a differentiable DAG constraint and the optimization of a suitable score function—often based on a structural equation model (SEM)—to fit the observed nodal signals. This perspective aligns with graph SP approaches to *graph topology identification*, where the goal is to learn an underlying graph that explains the observed signals [2].

However, classical graph models, including DAGs, capture only *pairwise* relationships between nodes. In many real-world datasets, interactions occur among groups of nodes (multi-way relations) rather than solely between node pairs [3], [4]. *Higher-order networks* such as hypergraphs, simplicial

complexes, and cell complexes have been introduced to represent these multi-way relationships and have recently gained attention in network science, SP, and ML. Prominent examples in SP and ML include *topological* SP over simplicial complexes [5], [6] and *hypergraph* SP [7], [8]. One of the critical tasks within this field is inferring *higher-order network structures*, e.g., learning a hypergraph topology from nodal observations. Several related works have started exploring this direction. For instance, when only nodal data are available, *graph Volterra models* have been proposed to capture non-pairwise interactions on networks by augmenting SEMs with higher-order (nonlinear) terms [9], [10]. These models leverage Volterra series expansions to account for multi-node influences and have shown improved performance in topology identification and link prediction tasks by modeling group interactions beyond simple edges.

While these advances enable the modeling and inference of higher-order relationships, they lack a clear connection to *acyclicity* in hypergraphs—the natural higher-order generalization of a DAG. In other words, although directed acyclic hypergraphs (DAHs) have been studied in the literature [11], methods for inferring such structures directly from nodal signals remain unclear. Motivated by the expressive power of DAHs, in this paper we pursue the inference of a DAH from nodal observations as a parsimonious model to explain the data in an SEM framework. Formally, given observed signals on a set of nodes, our goal is to learn a directed hypergraph structure (i.e., hyperedges with specified sets of source and target nodes) that best fits the data under a suitable SEM-based cost, while containing no directed cycles in the hypergraph sense.

This problem is significantly more challenging than inferring a simple graph (DAG) for three main reasons. First, there is no unique or universally accepted notion of directedness, edge weight, or even acyclicity in hypergraphs. We must therefore define what it means for a hypergraph to be “directed” and “acyclic” in our context to impose a meaningful DAG-like constraint. Second, a single hyperedge can connect an arbitrary number of nodes, which vastly enlarges the search space of possible structures beyond pairwise connections, ultimately making continuous formulations essential for scalability. Third, the expressive power of hypergraphs introduces a large number of degrees of freedom. Without additional structural restrictions or regularization, inferring a hypergraph from data risks overfitting or becoming ill-posed, especially for typical sample sizes. Stronger constraints or model assumptions are thus required to ensure tractability.

[†]Delft University of Technology, Delft, The Netherlands. ^{*}King Juan Carlos University, Madrid, Spain. We would like to thank Gonzalo Mateos for his contributions. This paper is supported by Netherlands Organization for Scientific Research (NWO) grant GraSPA (No. 19497); the Spanish AEI (AEI/10.13039/501100011033) grant PID2023-149457OB-I00; and the Community of Madrid via grant IDEA-CM (TEC-2024/COM-89) and the Ellis Madrid Unit.

To address these challenges, we develop a principled framework that exploits the advantages of continuous optimization (as in DAG learning) while controlling the structural complexity of the hypergraph model. Our main contributions are described next.

- We first define a *dedicated class of DHs* suited to our objectives, where each directed hyperedge originates from one or more source nodes and points to a single target node. Such DHs are sometimes also labeled as backward hypergraphs [11], [12].
- We introduce an equivalent *functional dependency directed graph* (FD-DG) representation of the hypergraph, which is an adaptation of the FD-graph first presented in [13]. The FD-DG is a bipartite directed graph composed of two disjoint node sets: the original nodes and a set of virtual nodes, each representing a distinct source-node set from which one or more hyperedges emerge. An original node connects to a virtual node if and only if it belongs to that virtual node's source set, while a virtual node connects to an original node whenever a corresponding hyperedge exists. This bipartite construction compactly encodes all hyperedge relationships.
- We impose an *acyclicity constraint* on the FD-DG to ensure that it forms a DAG. Critically, this condition guarantees that the underlying hypergraph is acyclic, since any directed cycle in the hypergraph would manifest as a cycle in the FD-DG through its virtual nodes.
- We propose a *DAH-based data model* that integrates an SEM with a graph Volterra formulation. In this model, each original node's value is expressed as a combination of contributions from its associated source-node sets (virtual nodes), analogous to a standard SEM on a DAG. The graph Volterra component enables continuous modeling of multi-node interaction effects, effectively lifting node signals onto the virtual nodes.
- Finally, we formulate a *DAH learning* (topology inference) problem that estimates the underlying DAH from signal observations. This formulation reduces hypergraph inference to the estimation of an acyclic bipartite graph (the FD-DG), allowing us to leverage continuous optimization techniques developed for DAG learning. We promote tractability and prevent overfitting by imposing sparsity-inducing penalties on the hypergraph connections, and we enforce acyclicity through a smooth DAG constraint on the FD-DG. The resulting optimization problem jointly estimates the hyperedges and their corresponding weights by minimizing an SEM-based loss subject to the DAH constraint.

II. DIRECTED ACYCLIC HYPERGRAPHS

In this section, we introduce the concept of a directed hypergraph along with its bipartite graph representation and how we could consider acyclicity in this setting.

A. Directed Hypergraphs

A directed hypergraph (DH) is a natural extension of a directed graph (DG) [11], [12], [14]–[16]. It consists of a set of

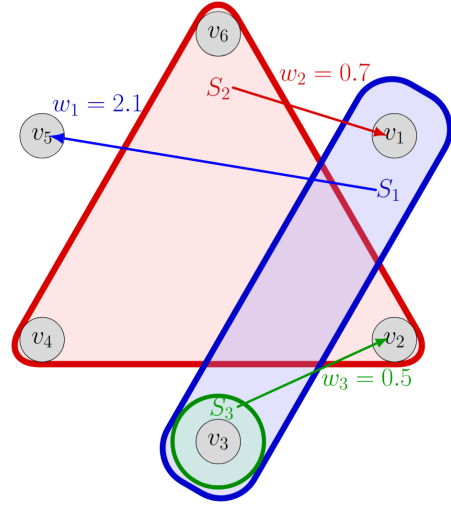


Fig. 1: A directed hypergraph with 6 nodes and 3 hyperedges. The nodes $v_1, \dots, v_6$ are shown with gray circles. There are 3 directed weighted hyperedges. The first hyperedge $h_1 = (S_1 = \{v_1, v_3\}, v_5, w_1 = 2.1)$ is in blue. The second hyperedge $h_2 = (S_2 = \{v_2, v_4, v_6\}, v_1, w_2 = 0.7)$ is in red. The third one, $h_3 = (S_3 = \{v_3\}, v_2, w_3 = 0.5)$, is in green.

nodes $\mathcal{V} = \{v_1, \dots, v_N\}$ and a set of (weighted) hyperedges $\mathcal{H} = \{h_1, \dots, h_M\}$, where each hyperedge is defined as the triplet $h_j = (S_j, t_j, w_j)$. Here, the *source* is a subset of nodes $S_j \subset \mathcal{V}$ (called the source-node set), the *target* is a single¹ node $t_j \in \mathcal{V} \setminus S_j$ (thus avoiding self-loops), and $w_j \in \mathbb{R}$ is the weight² associated with the hyperedge. A hyperedge h_j can therefore be interpreted as an arrow pointing from the source node set S_j to the target node t_j with weight w_j ; see Fig. 1 for an illustrative example. Note that this definition includes hyperedges of order one, where both the source and target consist of single nodes—as exemplified by the green hyperedge h_3 in Fig. 1.

Weights quantify the strength, reliability, or importance of the relationship between a source and its target. Their inclusion allows multiple hyperedges to originate from the same source-node set but point to different targets with distinct weights. This flexibility enables the modeling of heterogeneous interactions emanating from the same source, where each weighted hyperedge represents a unique source–target relation rather than grouping all targets into a single hyperedge. It also naturally allows a node to have multiple outgoing conventional edges with different weights. We assume there are K distinct source-node sets, with $K \leq M$. For simplicity, we order the hyperedges such that the first K hyperedges correspond to distinct source-node sets, i.e., $S_1, \dots, S_K$ are all unique. Furthermore, for each hyperedge h_j , $j = 1, \dots, M$, we

¹While there exist DH models where the target of a hyperedge can be a subset of nodes, we focus on the single-node case.

²While many works in graph theory consider unweighted graphs, incorporating hyperedge weights yields a more general formulation and facilitates several downstream tasks, as will become evident in subsequent sections.

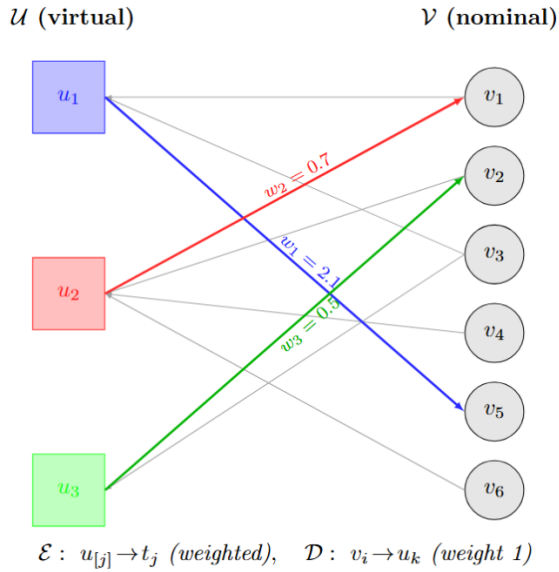


Fig. 2: The functional dependency directed graph of the hypergraph example in Fig. 1. The three source-node sets $\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3$ are represented respectively by the virtual nodes (in boxes) u_1, u_2, u_3 . The nominal nodes $v_1, \dots, v_6$ are still represented with the gray circles. The set of edges $\mathcal{E}$ collects the colored weighted edges, i.e., the former hyperedges h_j between the *virtual* nodes and the nominal nodes. The edge set $\mathcal{D}$ collects the unweighted gray edges, i.e., between the nominal nodes v_i and the virtual nodes u_k , that exist only if node v_i was part of the set $\mathcal{S}_k$ in the hypergraph.

define an index mapping $[j] = k$, with $k = 1, \dots, K$, whenever $\mathcal{S}_j = \mathcal{S}_k$.

B. Directed Graph Representation

It is often beneficial to represent a DH by means of a DG, so that existing graph algorithms can be used. Although many such graph representations exist, we focus on the so-called functional dependency directed graph (FD-DG), adapted from [13], where the DH is lifted to a larger bipartite DG. More specifically, we extend the *nominal* set of nodes $\mathcal{V}$ with a set of K additional *virtual* nodes $\mathcal{U} = \{u_1, \dots, u_K\}$, one for every distinct source-node set $\mathcal{S}_k$. Between the nominal and the virtual node sets, we will consider two sets of directed edges. First, every hyperedge h_j is represented by a (weighted) edge from the *virtual* node $u_{[j]} \in \mathcal{U}$ to the *nominal* node $t_j \in \mathcal{V}$ with weight w_j , denoted as $e_j = (u_{[j]}, t_j, w_j)$. The set of all these edges is denoted as $\mathcal{E} \subset \mathcal{U} \times \mathcal{V} \times \mathbb{R}$. Second, for every distinct source node set $\mathcal{S}_k$, all the *nominal* nodes $v_i \in \mathcal{S}_k \subset \mathcal{V}$ are connected to the *virtual* node $u_k \in \mathcal{U}$ with an arbitrary weight (let us assume it is 1). This leads to the second edge set $\mathcal{D} \subset \mathcal{V} \times \mathcal{U} \times 1$. The resulting FD-DG representation is then denoted as $(\mathcal{V}, \mathcal{U}, \mathcal{E}, \mathcal{D})$.

Let us now see how the adjacency matrix $\mathbf{W}$ of this FD-DG looks like. Assuming the nodes are ordered with set $\mathcal{V}$ at the

top and set $\mathcal{U}$ at the bottom, $\mathbf{W}$ has size $(N + K) \times (N + K)$ and can be written as

$$\mathbf{W} = \begin{bmatrix} \mathbf{0}_{N \times N} & \mathbf{W}_{12} \\ \mathbf{W}_{21} & \mathbf{0}_{K \times K} \end{bmatrix},$$

where $\mathbf{W}_{12}$ is of size $N \times K$ with entries

$$[\mathbf{W}_{12}]_{ik} = \begin{cases} w_j & \text{if } \exists j : v_i = t_j \text{ and } k = [j] \\ 0 & \text{otherwise} \end{cases},$$

and where $\mathbf{W}_{21}$ is of size $K \times N$ with entries

$$[\mathbf{W}_{21}]_{ki} = \begin{cases} 1 & \text{if } v_i \in \mathcal{S}_k \\ 0 & \text{otherwise} \end{cases}.$$

Since the FD-DG is bipartite, we have two zero blocks on the diagonal. So we have no edges within $\mathcal{V}$ and $\mathcal{U}$.

If the hypergraph contains hyperedges of order one, i.e., conventional edges, then we could replace these hyperedges consisting of a pair of edges in the FD-DG (one edge between $\mathcal{V}$ and $\mathcal{U}$ and one between $\mathcal{U}$ and $\mathcal{V}$) by a single directed edge within $\mathcal{V}$. In that case, the set $\mathcal{U}$ will exclude source node sets of size 1 and the top left corner of $\mathbf{W}$ will consist of the adjacency matrix of the conventional edges of the hypergraph. This will slightly reduce the size of the overall $\mathbf{W}$. Related to the example in Figs. 1 and 2, this implies that that hyperedge $h_3 = (\mathcal{S}_3 = \{v_3\}, v_2, w_2 = 0.5)$ in the DH can be represented with a conventional directed edge between nodes v_3 and v_2 rather than by the two edges $e_3 = (u_3, v_2, w_3)$ and (v_3, u_3) . However, for simplicity, we will not consider this representation here.

C. Introducing Acyclicity

The notion of acyclicity depends on the way we define a hyperpath. Here, we consider the simplest definition of a hyperpath [11], [12], i.e., an alternating sequence of nodes and hyperedges, $v_{i_1}, h_{j_1}, v_{i_2}, h_{j_2}, \dots, h_{j_L}, v_{i_{L+1}}$, where we assume that v_{i_l} is a source node of the hyperedge h_{j_l} , i.e., $v_{i_l} \in \mathcal{S}_{j_l}$, and $v_{i_{l+1}}$ is the target node of h_{j_l} , i.e., $v_{i_{l+1}} = t_{j_l}$, and this for all $l = 1, \dots, L$. Such a hyperpath is a cycle if the last target node ends up in the first source-node set, i.e., $v_{i_{L+1}} \in \mathcal{S}_{j_1}$. A hyperpath is acyclic (or cycle-free) if it does not contain a subpath that is a cycle, or in other words, for a hyperpath to be acyclic, we need $v_{i_{l+1}} \notin \{\mathcal{S}_{j_1}, \dots, \mathcal{S}_{j_l}\}$ for $l = 1, 2, \dots, L$. The DH is acyclic, if all hyperpaths are acyclic. Conceptually, a DAH can be seen as the extension of directed acyclic graphs to the hypergraph setting, where we cannot start from a node and by traversing the directed hyperedges return to it.

It is easy to show that this definition of acyclicity of the DH is equivalent to acyclicity of the related FD-DG. In other words, the FD-DG of a DAH is a DAG, hence its adjacency matrix $\mathbf{W}$ belongs to the set of DAG adjacency matrices. Recently, it has been shown that this set can be represented by the zero level set of a continuous smooth nonconvex function of $\mathbf{W}$. Many such so-called acyclicity functions have been proposed. Because of the favorable properties outlined in [17, Sec. 3.2], we consider the function

$$f(\mathbf{W}) = (N + K) \log s - \log(\det(s\mathbf{I} - \mathbf{W} \circ \mathbf{W})), \quad (1)$$

where $s \in \mathbb{R}^+$ can be chosen freely. Because of the block structure of $\mathbf{W}$, we can write

$$s\mathbf{I} - \mathbf{W} \circ \mathbf{W} = \begin{bmatrix} s\mathbf{I}_N & -(\mathbf{W}_{12} \circ \mathbf{W}_{12}) \\ -(\mathbf{W}_{21} \circ \mathbf{W}_{21}) & s\mathbf{I}_K \end{bmatrix}.$$

Applying the block determinant identity based on the Schur complement, we obtain that $\det(s\mathbf{I} - \mathbf{W} \circ \mathbf{W})$ is

$$s^{N+K} \det(\mathbf{I}_K - s^{-2}(\mathbf{W}_{21} \circ \mathbf{W}_{21})(\mathbf{W}_{12} \circ \mathbf{W}_{12})).$$

Substituting this result into (1) yields

$$f(\mathbf{W}) = -\log(\det(\mathbf{I} - s^{-2}(\mathbf{W}_{21} \circ \mathbf{W}_{21})(\mathbf{W}_{12} \circ \mathbf{W}_{12}))), \quad (2)$$

which captures the structure of our DAG. In conclusion, since the FD-DG of a DAH is a DAG, we can guarantee that a DH is acyclic if the adjacency matrix of its associated FD-DG $\mathbf{W}$ satisfies $f(\mathbf{W}) = 0$, with f having the form in (2).

III. STRUCTURAL EQUATION MODEL

So far, we have only discussed the structure of the DH (or DAH) and how to represent this by means of its FD-DG. However, how can we link this structure to the signals that are observed at the nodal level? One way to do this is by means of a so-called SEM. This model has been derived for conventional graphs and expresses a graph signal as the product of the adjacency matrix with the graph signal itself, up to some additive exogenous noise signal. In other words, it expresses a node signal by means of a linear combination of neighboring node signals.

Applying this model to our FD-DG, we first need to introduce nodal signals for the set of *virtual* nodes $\mathcal{U}$ which are related to the distinct source-node sets. Suppose the *nominal* node $v_i \in \mathcal{V}$ measures a signal x_i , and suppose we assign signal y_k to *virtual* node $u_k \in \mathcal{U}$ related to source-node set $\mathcal{S}_k$. Introducing $\mathbf{x} = [x_1, \dots, x_N]^\top$ and $\mathbf{y} = [y_1, \dots, y_K]^\top$ and stacking the two into the vector $\mathbf{z}$, i.e., $\mathbf{z}^\top = [\mathbf{x}^\top, \mathbf{y}^\top]^\top$, the SEM for the FD-DG can be written as

$$\mathbf{z} = \mathbf{W}\mathbf{z} + \boldsymbol{\zeta},$$

where $\boldsymbol{\zeta} = [\boldsymbol{\epsilon}^\top, \boldsymbol{\eta}^\top]^\top$ represents the exogenous noise with $\boldsymbol{\epsilon}$ the noise on $\mathbf{x}$ and $\boldsymbol{\eta}$ the noise on $\mathbf{y}$, although we will set this latter to zero, i.e., $\boldsymbol{\eta} = \mathbf{0}$. Due to the bipartite structure of the FD-DG, the SEM for $\mathbf{x}$ and $\mathbf{y}$ separately can then be written as

$$\mathbf{x} = \mathbf{W}_{12}\mathbf{y} + \boldsymbol{\epsilon}, \quad (3)$$

$$\mathbf{y} = \mathbf{W}_{21}\mathbf{x}. \quad (4)$$

It is clear that if we substitute (4) into (3), we obtain a conventional SEM for the nominal nodes, i.e., $\mathbf{x} = \mathbf{W}_{12}\mathbf{W}_{21}\mathbf{x} + \boldsymbol{\epsilon}$, and thus lose the higher-order information. The reason for this is that (4) enforces the signal y_k on the virtual node $u_k \in \mathcal{U}$ to be constructed as the *sum* of the signals on the nominal nodes in the related source-node set $\mathcal{S}_k \subset \mathcal{V}$.

As a result, we will drop (4) from our hypergraph SEM, and we instead assume that $y_k = \phi(\mathbf{x}_{\mathcal{S}_k})$, with $\mathbf{x}_{\mathcal{A}}$ the subvector of $\mathbf{x}$ related to the index set $\mathcal{A}$ and $\phi(\mathbf{a})$ any permutation invariant

function of the related elements in the vector $\mathbf{a}$ (excluding the sum in (4)). The considered hypergraph SEM can thus be expressed as

$$\mathbf{x} = \mathbf{W}_{12}\mathbf{y} + \boldsymbol{\epsilon}, \quad (5)$$

$$y_k = \phi(\mathbf{x}_{\mathcal{S}_k}), \text{ for } k = 1, \dots, K. \quad (6)$$

Note that this model is similar to the autoregressive graph Volterra model introduced in [9], but now viewed as a SEM on the FD-DG related to the DH.

The hypergraph SEM has different applications such as denoising, signal prediction, and topology identification. In the remainder of this paper, we focus on this latter application, thereby specifically paying attention to DAHs.

IV. TOPOLOGY IDENTIFICATION

In this section, we will consider multiple samples of $\mathbf{x}$, i.e., $\mathbf{x}_t$, $t = 1, 2, \dots, T$, to estimate an acyclic hypergraph from (5) and (6). Since we do not know which source-node sets are active, we need to consider all possible sets, up to some order P . So we will consider $\bar{K} = \sum_{p=1}^P \binom{N}{p}$ source-node sets and denote them as $\mathcal{S}_{\bar{k}}$, with $\bar{k} = 1, \dots, \bar{K}$. For every virtual node $u_{\bar{k}}$ related to $\mathcal{S}_{\bar{k}}$, we further construct the signals

$$\bar{y}_{t,\bar{k}} = \phi(\mathbf{x}_{t,\mathcal{S}_{\bar{k}}}), \quad \bar{k} = 1, \dots, \bar{K}, \quad t = 1, \dots, T, \quad (7)$$

yielding the virtual node signal $\bar{\mathbf{y}}_t = [\bar{y}_{t,1}, \dots, \bar{y}_{t,\bar{K}}]^\top$, $t = 1, 2, \dots, T$. We also do not know yet to which targets all the $\bar{K}$ source-node sets are pointing, and hence we need to consider all nodes (except the ones in the source-node set itself to exclude self-loops) as targets for every possible source-node set. In other words, we consider $N - p$ hyperedges for every source set $\mathcal{S}_{\bar{k}}$ of size p , one for every node $v_i \in \mathcal{V} \setminus \mathcal{S}_{\bar{k}}$. We call the resulting hypergraph, the complete hypergraph of order P . It has $\bar{M} = \sum_{p=1}^P (N - p) \binom{N}{p}$ hyperedges, although some hyperedge weights can still be zero. The adjacency matrix of the FD-DG related to this complete hypergraph is given by the $(N + \bar{K}) \times (N + \bar{K})$ matrix

$$\bar{\mathbf{W}} = \begin{bmatrix} \mathbf{0}_{N \times N} & \bar{\mathbf{W}}_{12} \\ \bar{\mathbf{W}}_{21} & \mathbf{0}_{\bar{K} \times \bar{K}} \end{bmatrix}.$$

Here, $\bar{\mathbf{W}}_{12} \in \mathbb{R}^{N \times \bar{K}}$ is similarly defined as $\mathbf{W}_{12}$ but now considering all possible source-node sets, up to order P . Note that we explicitly set to zero all entries $[\bar{\mathbf{W}}_{12}]_{i\bar{k}}$ for which $v_i \in \mathcal{S}_{\bar{k}}$ (since we do not allow self-loops). Furthermore, $\bar{\mathbf{W}}_{21} \in \mathbb{R}^{\bar{K} \times N}$ indicates which nodes are in a particular source-node set. This matrix $\bar{\mathbf{W}}_{21}$ is fully known, since it includes all source-node sets and we know how they are constructed. Even if some of these source-node sets turn out not to be active, leaving them in $\bar{\mathbf{W}}_{21}$ will not change the acyclicity condition $f(\bar{\mathbf{W}}) = 0$.

So the only unknown in $\bar{\mathbf{W}}$ is $\bar{\mathbf{W}}_{12}$ which contains the hyperedge weights (remember that some of these weights are explicitly set to zero to avoid self-loops). A non-zero column in $\bar{\mathbf{W}}_{12}$ means the related source-node set is active and the non-zero elements in that column tell us which target nodes this source-node set points at (and hence how many

hyperedges are related to that source-node set). Since only a limited number of source-node sets will be active, we can adopt a column-wise sparsity constraint on $\bar{\mathbf{W}}_{12}$. Furthermore, since an active source-node set will only point to one or a few target nodes, we can also use an element-wise sparsity constraint on $\bar{\mathbf{W}}_{12}$.

As a result, a natural way to estimate $\bar{\mathbf{W}}_{12}$ from $\mathbf{x}_t$, $t = 1, 2, \dots, T$ and the related $\bar{\mathbf{y}}_t$, $t = 1, 2, \dots, T$ [see (7)], under the considered sparsity constraints is by solving

$$\min_{\bar{\mathbf{W}}_{12}} \frac{1}{T} \sum_{t=1}^T \|\mathbf{x}_t - \bar{\mathbf{W}}_{12} \bar{\mathbf{y}}_t\|_2^2 + \lambda_1 \|\bar{\mathbf{W}}_{12}\|_{2,1} + \lambda_2 \|\bar{\mathbf{W}}_{12}\|_1, \quad (8)$$

where λ_1 and λ_2 are the sparsity regularizing weights, $\|\mathbf{A}\|_{2,1} = \sum_j \|\mathbf{A}_{:,j}\|_2$, and $\|\mathbf{A}\|_1 = \sum_{i,j} |\mathbf{A}_{i,j}|$. From [18], we know that when $P = 1$ and ϕ is the identity function (i.e., the HG is just a DG and $\mathbf{y}_t = \mathbf{x}_t$), the solution for $\lambda_1 = \lambda_2 = 0$ does not correspond to the true graph. However, when in addition acyclicity is assumed and the noise ϵ satisfies certain conditions, problem (8) for $\lambda_1 = \lambda_2 = 0$ is identifiable [19], [20]. This example shows that acyclicity can improve identifiability.

Similarly, we believe that introducing acyclicity (as well as sparsity) for the more general $P > 1$ case will lead to a better solution. As such, we will boost problem (8) by adding the acyclicity constraint $f(\bar{\mathbf{W}}) = 0$ as a regularizer. Our final DAH estimation problem is then given by

$$\min_{\bar{\mathbf{W}}_{12}} \frac{1}{T} \sum_{t=1}^T \|\mathbf{x}_t - \bar{\mathbf{W}}_{12} \bar{\mathbf{y}}_t\|_2^2 + \lambda_1 \|\bar{\mathbf{W}}_{12}\|_{2,1} + \lambda_2 \|\bar{\mathbf{W}}_{12}\|_1 - \nu \log \det (\mathbf{I}_{\bar{K}} - s^{-2} (\bar{\mathbf{W}}_{21} \circ \bar{\mathbf{W}}_{21}) (\bar{\mathbf{W}}_{12} \circ \bar{\mathbf{W}}_{12})). \quad (9)$$

where we again note that $\bar{\mathbf{W}}_{21}$ is a fully known matrix and the only optimization variable is $\bar{\mathbf{W}}_{12}$ (except for the entries that are explicitly set to zero).

V. EXPERIMENTS

We evaluate the creation-and-recovery pipeline on synthetic DAHs, which are constructed as follows. We fix $N = 10$ nominal nodes and choose $P = 2$, so the $\bar{K} = 55$ source-node sets are all size-1 and size-2 subsets of $\mathcal{V}$. Among these we activate $K = 10$ distinct source sets; for each active $\mathcal{S}$ we select up to 2 targets in $\mathcal{V} \setminus \mathcal{S}$ and assign nonzero weights.

Acyclicity is enforced by imposing a global order on the nodes. Let $\pi : \mathcal{V} \rightarrow \{1, \dots, N\}$ be a permutation. We only admit a hyperedge $h = (\mathcal{S}, t, w)$ if

$$\max_{u \in \mathcal{S}} \pi(u) < \pi(t), \quad (10)$$

so source nodes always precede their targets in the order. Under this rule, every alternating path in the FD-DG strictly increases the order index, hence no cycle can form and the synthesized directed hypergraph is acyclic. For the weights on $\bar{\mathbf{W}}_{21}$ we follow DAGMA [17] and draw each nonzero entry uniformly from $[-2, -0.5] \cup [0.5, 2]$.

Given a synthesized DAH, we generate T samples by a single forward pass along the order π . For sample t , we

compute for each target node v_j in increasing $\pi(v_j)$, the following signals:

$$y_{t,k} = \prod_{v_i \in \mathcal{S}_k} x_{t,i}, \quad \text{for all } k \text{ with } \max_{v_i \in \mathcal{S}_k} \pi(v_i) < \pi(v_j), \quad (11)$$

$$x_{t,j} = \sum_{k=1}^K [\mathbf{W}_{12}]_{j,k} y_{t,k} + \epsilon_{t,j}, \quad \epsilon_{t,j} \sim \mathcal{N}(0, \sigma^2), \quad (12)$$

where the noise variance σ^2 is set to 1 for simplicity.

Our algorithm then solves (9) over $\bar{\mathbf{W}}_{12}$, with $\bar{\mathbf{W}}_{21}$ fixed and assuming the product aggregator in (7). Recall that we consider a hard source-exclusion mask, i.e., we enforce $[\bar{\mathbf{W}}_{12}]_{i,\bar{k}} = 0$ whenever $v_i \in \mathcal{S}_{\bar{k}}$. Following DAGMA [17], we gradually decrease s and enlarge the acyclicity weight ν during the optimization to obtain a DAH.

For different sample sizes T we draw 10 random DAHs and corresponding datasets, run the estimator, and compute the normalized squared error of $\hat{\bar{\mathbf{W}}}_{12}$ against $\bar{\mathbf{W}}_{12}$. We report the average over the 10 DAHs as shown in Figure 3. We clearly observe that the acyclicity constraint improves the performance significantly.

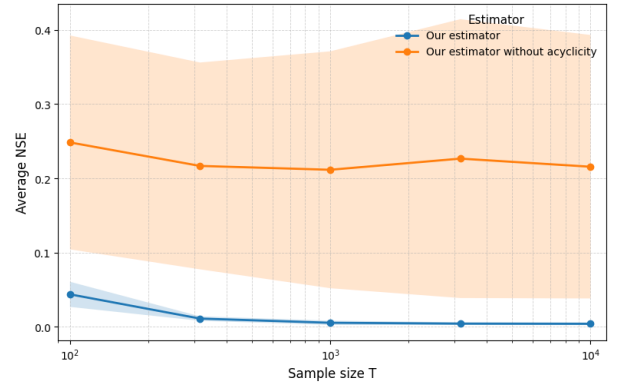


Fig. 3: Average normalized squared error of $\hat{\bar{\mathbf{W}}}_{12}$ versus the sample size T

In Figure 4, we consider a particular DAH realization and plot the energy related to the different estimated source-node sets, i.e., $\|[\bar{\mathbf{W}}_{12}]_{:, \bar{k}}\|_2$. Here, it is clear that only a few source-node sets are discovered (due to the sparsity constraint) and that considering the acyclicity constraint leads to a better and more sparse fit.

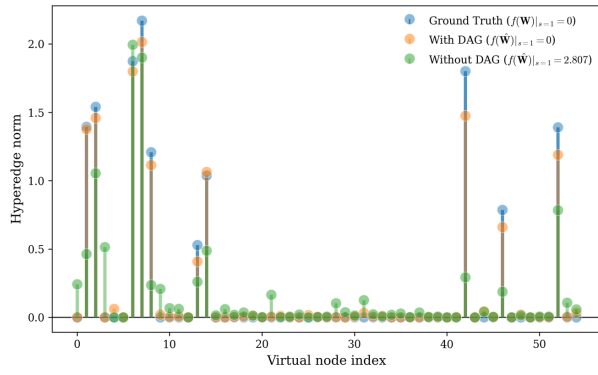


Fig. 4: Energy of estimated source-node sets, i.e., $\|[\hat{\mathbf{W}}_{12}]_{:, \bar{k}}\|_2$, versus the source-node set index $\bar{k}$.

VI. CONCLUSIONS

We reviewed a dedicated class of DHs suited to our setup and introduced the corresponding class of DAHs. We estimated a DAH from nodal data by i) lifting the DH to a bipartite FD-DG; ii) imposing a continuous acyclicity (DAG) constraint on the FD-DG, which enforces a DAH; iii) assuming data can be explained by a nonlinear Volterra-based SEM and iv) solving the related constrained hypergraph SEM data fitting problem. Synthetic simulation results show that the acyclicity constraint leads to improved DAH estimates.

Future work will focus on a study of the identifiability of the problem. Furthermore, we will work on reducing the complexity by exploiting a low-rank tensor model and on extending our method to dynamic online settings. We will finally test our proposed technique on real data from, e.g., brain science, gene regulatory networks, or chemical reactions.

REFERENCES

- [1] X. Zheng, B. Aragam, P. Ravikumar, and E. P. Xing, "DAGs with NO TEARS: Continuous optimization for structure learning," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2018.
- [2] G. Mateos, S. Segarra, A. G. Marques, and A. Ribeiro, "Connecting the dots: Identifying network structure via graph signal processing," *IEEE Signal Process. Mag.*, vol. 36, no. 3, pp. 16–43, 2019.
- [3] F. Battiston, G. Cencetti, I. Iacopini, V. Latora, A. Patania, J.-G. Young, and G. Petri, "Networks beyond pairwise interactions: Structure and dynamics," *Nat. Phys.*, vol. 17, no. 10, pp. 1093–1098, 2020.
- [4] A. R. Benson, D. F. Gleich, and J. Leskovec, "Higher-order organization of complex networks," *Science*, vol. 353, no. 6295, pp. 163–166, 2016.
- [5] S. Barbarossa and S. Sardellitti, "Topological signal processing over simplicial complexes," *IEEE Transactions on Signal Processing*, vol. 68, pp. 2992–3007, 2020.
- [6] E. Isufi, G. Leus, B. Beferull-Lozano, S. Barbarossa, and P. Di Lorenzo, "Topological signal processing and learning: Recent advances and future challenges," *Signal Processing*, p. 109930, 2025.
- [7] M. T. Schaub, Y. Zhu, J. B. Seby, T. M. Roddenberry, and S. Segarra, "Signal processing on higher-order networks: Livin' on the edge... and beyond," *Signal Process.*, vol. 187, p. 108149, 2021.
- [8] S. Zhang, Z. Ding, and S. Cui, "Introducing hypergraph signal processing: Theoretical foundation and practical applications," *IEEE Internet Things J.*, vol. 7, no. 1, pp. 639–660, Jan. 2020.
- [9] Q. Yang, M. Coutino, G. Leus, and G. Giannakis, "Autoregressive graph Volterra models and applications," *EURASIP J. on Adv. in Signal Process.*, vol. 2023, no. 4, 2023.
- [10] A. Buciulea, E. Isufi, G. Leus, and A. G. Marques, "Learning graphs and simplicial complexes from data," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*. IEEE, 2024, pp. 9861–9865.

- [11] M. Thakur and R. Tripathi, "Linear connectivity problems in directed hypergraphs," *Theoretical Computer Science*, vol. 410, no. 27, pp. 2592–2618, 2009.
- [12] G. Gallo, G. Longo, S. Pallottino, and S. Nguyen, "Directed hypergraphs and applications," *Discrete Appl. Math.*, vol. 42, pp. 177–201, 1993.
- [13] G. Ausiello, A. D'Atri, and D. Saccà, "Graph algorithms for functional dependency manipulation," *J. ACM*, vol. 30, pp. 752–766, 1983.
- [14] A. R. Berg, B. Jackson, and T. Jordán, "Edge splitting and connectivity augmentation in directed hypergraphs," *Discrete Math.*, vol. 273, no. 1–3, pp. 71–84, 2003.
- [15] A. Frank, T. Király, and Z. Király, "On the orientation of graphs and hypergraphs," *Discrete Appl. Math.*, vol. 131, no. 2, pp. 385–400, Sep. 2003.
- [16] G. Ausiello and L. Laura, "Directed hypergraphs: Introduction and fundamental algorithms—a survey," *Theoretical Computer Science*, vol. 658, pp. 293–306, 2017.
- [17] K. Bello, B. Aragam, and P. Ravikumar, "DAGMA: Learning DAGs via M-matrices and a log-determinant acyclicity characterization," in *Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 8226–8239.
- [18] Y. Han, A. Natali, and G. Leus, "Graph topology identification based on covariance matching," in *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2025.
- [19] S. Shimizu, P. O. Hoyer, A. Hyvärinen, A. Kerminen, and M. Jordan, "A linear non-gaussian acyclic model for causal discovery," *Journal of Machine Learning Research*, vol. 7, no. 10, 2006.
- [20] J. Peters and P. Bühlmann, "Identifiability of Gaussian structural equation models with equal error variances," *Biometrika*, vol. 101, no. 1, pp. 219–228, 2013.