Data-driven Control of Hypergraphs: Leveraging THIS to Damp Noise in Diffusive Hypergraphs

Robin Delabays[1], Yuanzhao Zhang[2], Florian Dörfler[3], and Giulia De Pasquale[4].

Abstract— **Controllability determines whether a system's state can be guided toward any desired configuration, making it a fundamental prerequisite for designing effective control strategies. In the context of networked systems, controllability is a well-established concept. However, many real-world systems, from biological collectives to engineered infrastructures, exhibit higher-order interactions that cannot be captured by simple graphs. Moreover, the way in which agents interact and influence one another is often unknown and must be inferred from partial observations of the system. Here, we close the loop between a hypergraph representation and our recently developed hypergraph inference algorithm, THIS, to infer the underlying multibody couplings. Building on the inferred structure, we design a parsimonious controller that, given a minimal set of controllable nodes, steers the system toward a desired configuration. We validate the proposed system identification and control framework on a network of Kuramoto oscillators evolving over a hypergraph.**

I. Introduction

Motivation. Networked systems provide a powerful framework for describing the collective dynamics of interacting agents, from power grids [1] to biological [2] and social systems [3]. However, in many real systems, interactions that naturally occur among groups of agents cannot be reduced to a set of pairwise interactions, for example, in neuronal assemblies [4], ecological communities [5], and social groups [6] where the state of an individual depends simultaneously on multiple others. These higher-order interactions are more accurately captured by hypergraphs or simplicial complexes, which generalize networks by allowing multi-body connections [7]. The ability to understand and control such systems is crucial for ensuring stability and robustness. Yet, in many practical scenarios, the underlying interaction structure is not directly observable and must be inferred from limited data. This lack of full structural knowledge challenges traditional control approaches, which typically assume a known network topology.

By integrating structure inference and control design within a unified framework, we aim to develop principled tools for identifying and steering higher-order dynamical systems, enabling effective interventions even when the governing interactions are only partially known.

More concretely, extensions of the well studied Kuramoto model to include three-body or higher-order interactions have revealed that these couplings can make the synchronized state's basin of attraction smaller yet deeper, raising the critical coupling threshold while enhancing robustness once synchronization is achieved [8]. These findings highlight that higher-order structures fundamentally reshape the landscape of collective dynamics.

The coupling strength plays a central role in determining the collective behavior of oscillator networks [9]. It governs the transition from incoherence to synchrony and critically shapes the stability, robustness, and transient dynamics of the ensemble. However, in most practical scenarios, coupling strengths are not directly measurable, as the interactions between oscillators are often mediated by latent physical, chemical, or informational processes. Instead, only phase measurements are typically accessible. This limitation motivated us to design a closed loop algorithm between our proposed Taylor-based Hypergraph Inference using SINDy (THIS) algorithm [10], able to estimate effective coupling parameters and interaction structures from time series data together with a control law in order to drive the system towards a desired state. Such approach, allows to identify hidden connectivity patterns, assess synchronization regimes, and design control strategies in real-world networks where the underlying coupling mechanisms remain partially unknown.

Literature Review. The notion of structural controllability for linear time invariant systems evolving over network traces back to the 70s [11] with the work of Lin on "structural controllability". Both algebraic and graph theoretic conditions were provided to ensure that a system can be driven towards any point of the state space based solely on topological considerations. Further analytical tools to certify structural controllability of a network have been provided in [12], where the authors identify the set of driver nodes with time-dependent control that can drive the system dynamics. The extension of the notion of controllability to hypergraphs is much more recent with Refs. [13], [14], [15], which relies on tensor algebra and polynomial control theory [16]. They provide theoretical results about the minimum number of control nodes to be controlled given specific hypergraph structures.

*Contributions.*We leverage our recently proposed identification algorithm, Taylor-based Hypergraph Inference using

*RD was supported by the Swiss National Science Foundation under grand nr. 200021_215336

[1]Robin Delabays is with the University of Applied Sciences of Western Switzerland (HES-SO), Sion, Switzerland. robin.delabays@hevs.ch

[2]Yuanzhao Zhang is with the Santa Fe Institute, Santa Fe, New Mexico, USA.yzhang@santafe.edu

[3]Florian Dörfler is with the Department of Information Technology and Electrical Engineering at ETH Zürich, Zürich, Switzerland. dorfler@ethz.ch

[4]Giulia De Pasquale is with the Control Systems group in the Electrical Engineering Department and the Eindhoven AI Systems Institute at Eindhoven Univeristy of Technology, Eindhoven, The Netherlands.g.de.pasquale@tue.nl

SINDy (THIS) [10], to infer coupling strengths in a network of coupled oscillators. Building on this inferred model, we design a simple yet effective proportional control strategy that enables any system evolving over a hypergraph to reach a desired fixed point. In a similar spirit as Ref. [12], we introduce a notion of a *minimal set of controllable nodes*, allowing us to design a parsimonious controller that acts only on a minimal subset of nodes to drive the system's dynamics. Finally, we demonstrate the proposed identification and control framework on a hypergraph of Kuramoto oscillators, successfully steering them toward the synchronous state.

Article Organization. The paper is organized as follows. Section II introduces the notation and preliminary concepts used throughout the paper. Section III presents the problem formulation for the design of a parsimonious state-feedback controller. Section IV details the proposed algorithm design, while Section V demonstrates its effectiveness on a hypergraph of Kuramoto oscillators. Finally, Section VI concludes the manuscript.

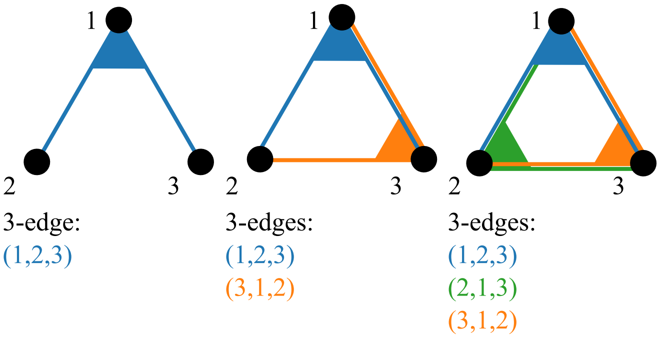

Fig. 1. Examples of hypergraphs. In the left panel, nodes 2 and 3 are leaves and need to be controlled. In the middle panel, only node 2 is a leaf and needs to be controlled. In the right panel, there is no leaf node and one needs to control any one of the three nodes.

II. Notation and Preliminaries

We provide the notation and preliminary concepts that will be used throughout the manuscript.

A. Hypergraphs

For a matrix $A \in \mathbb{R}^{n\times n}$, let $a_{i,j}$ be its (i,j)-th entry. A *p-th order hypergraph* $\mathscr{H}$ is a tuple $\mathscr{H}(V,E,A^{(2)},...,A^{(p)})$, where $V = \{1,\dots,n\}$ indicates the set of nodes, $E^{(k)} \subset V^k$ is the set of k-th order (hyper)edges and $E = \bigcup_k E^{(k)}$ is the set of all (hyper)edges, and $A^{(k)} \in \mathbb{R}^{n\times\cdots\times n}$ is the k-th order *adjacency tensor*, such that a k-hyperedge $(i_1,i_2,\dots,i_k)$ is present in the hypergraph if and only if $A^{(k)}_{i_1,i_2,\dots,i_k} \neq 0$.

In this work, we consider *directed* hypergraphs, meaning that each permutation of k indices $(i_1,\dots,i_k)$ denotes a distinct directed hyperedge. The tuple $e = (i_1,i_2,\dots,i_k)$ indicates that nodes $i_2,\dots,i_k$ all influence node i_1's dynamics. We say that the node at the first index is the *head* of a hyperedge and the other nodes are the *tails*, e.g., in $e = (i_1,i_2,\dots,i_k)$, node i_1 is the head of e, and $i_2,\dots,i_k$ are the tails.

We define a sequence of edges $(e_1,...,e_\ell)$ is a *directed path* from node i to node j if i is the head of e_1, j is a tail of e_ℓ, and the head of e_k is one of the tails of e_{k-1}, for each $k \in \{2,...,\ell\}$. A node is said to be a *leaf* if it is not the head of any edge. For instance, in the middle panel of Figure 1, node 2 is a leaf, while nodes 1 and 3 are not. A hypergraph is said to be *connected* if for any two nodes i and j, there is either a path from i to j or a path from j to i.

B. Higher-order Kuramoto model

Let us consider the generalization of the Kuramoto model to higher-order interactions. In this work, we illustrate our results on the third-order Kuramoto model where the interactions among n phase oscillators is modeled by a hypergraph $\mathscr{H}(V,E,A^{(2)},A^{(3)})$. The dynamics of the i-th phase is governed by

$$\dot{x}_i = \omega_i + \sum_{j=1}^{n} A^{(2)}_{ij}\sin(x_j - x_i) + \sum_{j,k=1}^{n} A^{(3)}_{ijk}\sin(x_j + x_k - 2x_i) \tag{1}$$

where $x_i \in S^1$ represents the phase of oscillator i and $\omega_i \in \mathbb{R}$ is its natural frequency.

III. Hypergraph Stabilization via Noise Rejection

Consider a hypernetworked system of arbitrary order subject to additive noise described as

$$\dot{x}_i = \xi_i + f(x_i) + \sum_j a_{ij} f^{(2)}(x_i,x_j) + \sum_{j,k} a_{ijk} f^{(3)}(x_i,x_j,x_k) + \cdots \tag{2}$$

where $f^{(k)}$ is the k-th-order interaction function and ξ_i is the noise term at node i. we do not have any knowledge about the functions $f^{(k)}$, and we only have access to time-series trajectories $x(t)$ of (2). We will assume that the autonomous system in (2) admits at least one stable equilibrium point and that it is operating in the proximity of one of them, as formalized by the following assumption.

Assumption 1: The autonomous system in (2) admits at least one locally stable equilibrium point x^*.

Note that, due to the noise term ξ_i, the system evolves randomly around the fixed point $\boldsymbol{x}^*$. For a strong enough disturbance, the system may even escape the basin of attraction of $\boldsymbol{x}^*$ and either reach another basin of attraction, or lose stability and diverge. Our goal in this manuscript is to design an algorithm that, given the state trajectories $x(t)$, identifies the hyperedges in the system and a parsimonious droop control that dampens the excursions due to noise and prevents destabilization of the system. By *parsimonious*, we mean that we aim at minimizing the number of nodes that need to be controlled. Our state feedback control problem is formally stated as follows:

Problem 1: Given state trajectories of the system (2), design a parsimonious state-feedback controller $u(x)$, so that (2) converges to the closest equilibrium point x^*.

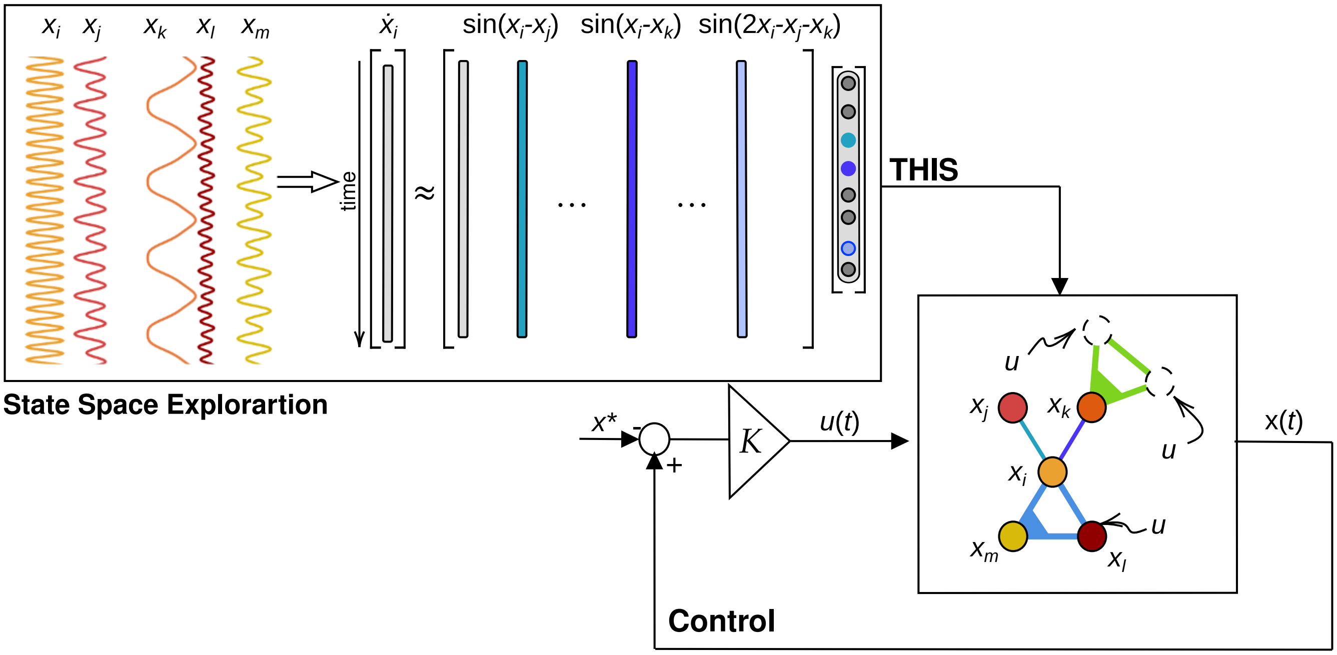

Fig. 2. Algorithm process pipeline: In phase 1) trajectories are collected for a system operating in the vicinity of a locally stable equilibrium point for a time interval $[t_0, t_1]$; at time instant t_2 phase 2 starts, where THIS is applied to infer the hypergraph topology, after which, at phase 3) the minimal set of controllable nodes is detected and the system driven towards the desired equilibrium point x^*.

This problem is of particular relevance in the context of chemical reactions, where the whole interaction structure may be complicated to grasp. Furthermore, the ability to drive the reaction towards a desired state, using minimal resources is valuable for many situations, such as optimizing catalytic processes in industrial chemistry to increase yield while reducing waste [17].

IV. Identification and Control Algorithm

In order to solve Problem 1, we first need to come up with a formal definition of *structural controllability* of an hypergraph and *minimal set of controllable nodes*, made up of the minimal number of nodes that need to be controlled to drive the system towards any point of the state space given no explicit knowledge about the hyper-edge weights. To do so, we extend the definition of structural controllabiliry and of minimal set of driver nodes, provided in Ref. [11], for networks of pair-wise interactions.

Definition 1 (Structural Controllability): A dynamical system with higher-order interactions desribed by a p-th order hypergraph $\mathscr{H}(V, E, A^{(2)}, \ldots, A^{(p)})$ is *structurally controllable* by the node set $S \subset V$, if controlling the nodes in S enables to steer the system towards any point in the state space.

Definition 2 (Minimal Set of Controllable Nodes): The minimal set of controllable nodes in a hypergraph is given by the set of leaf nodes. We denote it by S^*.

Analogously to what is observed in [11], once all leaf nodes are controlled, there is a directed path from the input signal to any node in the system. On the other hand, leaves are not reachable by any direct path by definition, and therefore the set of leaves constitute a minimalistic choice.

For instance in the left panel of Figure 1, the input signal cal propagate from node 2 to node 1 and from node 3 to node 1. However, the input signal cannot propagate to node 2, neither from node 1 nor from node 3. All together, both nodes 2 and 3 have to be controlled. In the middle panel of Figure 1, the input signal propagate to node 1 and node 3. It is therefore enough to control node 2. Finally, in the right panel, the input signal propagate from any node to any other. Therefore, any of the node can be controlled in order to control the whole system.

Notice that according to our definition, a system could be disconnected, while all nodes are reachable (e.g., two copies of the right panel of Figure 1). In such a case, one would need to control at least one node in each of the connected components.

Given the notion of minimal set of controllable nodes, we are now ready to design our system identification and control algorithm. The proposed algorithm follows the following process pipeline, summarized in Figure 2:

A. Initialization and State Space Exploration
B. System Identification via THIS
C. Droop Controller Design

All steps are explained in detail in the following dedicated subsections.

A. Initialization and State Space Exploration

The system is initialized in the vicinity of a locally stable equilibrium point $\boldsymbol{x}^*$ and recording of the nodes' time series begins at time $t_0 = 0$. Given the noise, the system explores

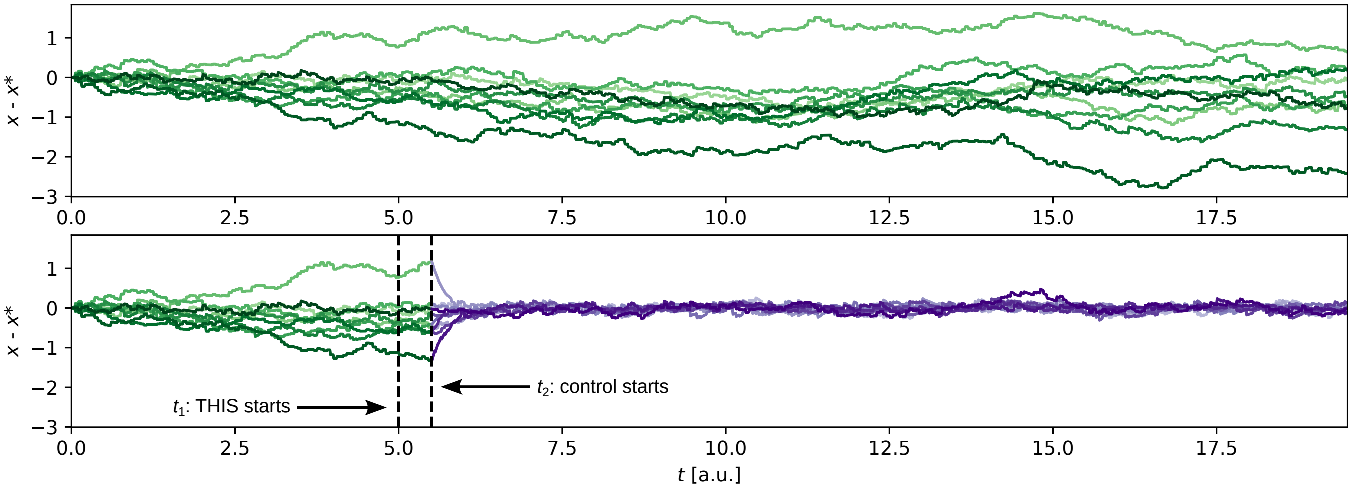

Fig. 3. Top panel: Trajectory of the 10 agents of a 3rd-order Kuramoto model on a random 3-hypergraph. Each agent deviates from its steady state due to noise. Bottom panel: The time series of the nodes over the first 5 time units are used to infer the hypergraph structure using THIS. Within this hypergraph, the leaf nodes are identified and a proportional control is applied to them, starting at time $t_2 = 5.5$. From $t_0 = 0$ to $t_2 = 5.5$, the two time series are exactly the same.

the neighborhood of $\boldsymbol{x}^*$ and the trajectories, after a certain time interval $[t_0, t_1]$, are rich enough to serve as input to the identification algorithm THIS. No control is applied within the time interval $[t_0, t_1]$.

B. System Identification via THIS

At time $t_1 > t_0$, the recorded time series are provided as input to THIS to infer the hypergraph structure. We recall that THIS is a model-free hypergraph inference algorithm designed to uncover causal relationships among system variables. It enables robust hypergraph reconstruction with minimal prior knowledge about the underlying dynamics, requiring only that the system can be represented as a set of coupled differential equations.

In terms of the exploration domain, noise plays a constructive role in THIS: the explored region should be sufficiently large to capture the system's nonlinear behavior (beyond the locally linear regime), yet not excessively broad to maintain the validity of the Taylor approximation, which is crucial for THIS. When this balance is achieved, THIS remains fully agnostic to the specific regions of the state space from which data are collected.

The run time of THIS depends on the size of the system and on the length of the time series considered. In our pipeline, THIS runs from t_1 to t_2, while the system evolves freely.

C. Droop Controller Design

Given the inferred hypergraph, a parsimonious droop control is designed and applied to the nodes in the minimal set S^* of controllable nodes. The state-feedback control follows a linear proportional control law $u_k = K(x_k - x_k^*)$ with $k \in S^*$ and K chosen such that stability of $\boldsymbol{x}^*$ is preserved. At time instant $t_2 \geq t_1$, the droop control is turned on and noise is damped throughout the system. The effectiveness of this control scheme is showcased in Section V.

V. Mathematical Simulations

Let us consider the third order Kuramoto model as in Equation (1), with $\boldsymbol{x}^*$ being its synchronous state. In our example, the system is made up of 10 nodes interacting in a random third order hypergraph. As shown in Figure 3 each state deviates from the synchronous state due to noise. After the state space exploration in the time interval $t \in [0, 5]$ all the state space trajectories are collected and the identification step is initiated. THIS infers the hypergraphs in the time interval $t \in [5, 5.5]$. Here we virtually allocate 0.5 time units to the inference for illustrative purpose. In practice, the inference time strongly depends on the system size and the time series length. Starting at $t_2 = 5.5$, the proportional control law $u_k = K(x_k - x_k^*)$ with $K = 5$ is applied to each leaf node identified by THIS, which in turn drive the overall system towards the desired state.

Interestingly, we notice that THIS does not infer exactly the actual hypergraph underlying the dynamics. Indeed, in this particular example, THIS achieves a true positive rate of 55% on the inference of 3-edges. In other words, only 55% of the existing 3-edges were correctly identified by the algorithm. At the same time, THIS infers some 2-edges, while there was none in the actual hypergraph. Nevertheless, THIS perfectly identifies the set of leaf nodes and hence does a perfect job in designing a parsimonious and effective controller.

This imperfect inference is generic to our example and with our choice of parameters, according to our simulations. The main reason for these errors in the inferred edges stems from the limited radius of exploration allowed by the chosen noise, over the chosen time interval. The inference would have been better should the magnitude of the noise be larger, or the exploration time be longer.

VI. Conclusions

We proposed an explicit implementation of a data-driven controller on a diffusive hypernetwork system. Based on measured time series of the system's nodes, THIS infers the underlying hypernetwork structure and designs a minimal controller accordingly.

So far, the proposed approach applies mostly to diffusive couplings where a stable steady state exists. The strength of the approach lies in the fact that it is mostly system agnostic and does not need any prior on the studied system.

Most interestingly, it appears that, despite a partially incorrect inference of the hypergraph, the set of parsimonious control nodes is perfectly identified. This last observation opens promising avenues for the application of (hyper)network inference. Indeed, it appears that inference algorithms like THIS are capable of extracting the key information needed for the hypergraph controllability, despite model mismatches.

References

[1] B. Lesieutre, P. Sauer, and M. Pai, "Existence of solutions for the network/load equations in power systems," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 46, no. 8, pp. 1003–1011, 1999.

[2] D. A. Drubin, J. C. Way, and P. A. Silver, "Designing biological systems," *Genes & Development*, vol. 21, pp. 242–254, 2007.

[3] A. V. Proskurnikov and R. Tempo, "A tutorial on modeling and analysis of dynamic social networks. part i," *Annual Reviews in Control*, vol. 43, p. 65–79, 2017.

[4] L. Xiao, J. Wang, P. H. Kassani, Y. Zhang, Y. Bai, J. M. Stephen, T. W. Wilson, V. D. Calhoun, and Y.-P. Wang, "Multi-hypergraph learning-based brain functional connectivity analysis in fmri data," *IEEE Transactions on Medical Imaging*, vol. 39, no. 5, pp. 1746–1758, 2020.

[5] D. J. Higham and H.-L. de Kergorlay, "Epidemics on hypergraphs:spectral thresholds forextinction," *Proceedings Royal Society A*, vol. 477, no. 2252, pp. 1–18, 2021.

[6] U. Alvarez-Rodriguez, F. Battiston, G. F. de Arruda, Y. Moreno, M. Perc, and V. Latora, "Evolutionary dynamics of higher-order interactions in social networks," *Nature Human Behaviour*, vol. 5, no. 5, pp. 586–595, 2021.

[7] G. Bianconi, *Higher-Order Networks*, ser. Elements in the Structure and Dynamics of Complex Networks. Cambridge University Press, 2021.

[8] Y. Zhang, M. Lucas, and F. Battiston, "Higher-order interactions shape collective dynamics differently in hypergraphs and simplicial complexes," *Nature Communications*, vol. 14, no. 1, p. 1605, 2023.

[9] F. Dörfler, M. Chertkov, and F. Bullo, "Synchronization in complex oscillator networks and smart grids," *Proceedings of the National Academy of Sciences*, vol. 110, no. 6, pp. 2005–2010, 2013.

[10] R. Delabays, G. De Pasquale, Y. Zhang, and F. Dörfler, "Hypergraph reconstruction from dynamics," *Nauture Communications*, vol. 16, no. 2691, pp. 1–9, 2025.

[11] C. T. Lin, "Structural controllability," *IEEE Transactions on Automatic Control*, vol. 19, no. 3, pp. 201–208, 1974.

[12] Y. Liu, J. Slotine, and A. L. Barábasi, "Controllability of complex networks," *Nature*, vol. 473, pp. 167–173, 2011.

[13] C. Chen, A. Surana, A. M. Bloch, and I. Rajapakse, "Controllability of hypergraphs," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 2, pp. 1646–1657, 2021.

[14] W. Ma, X. Bao, and C. Ma, "Controllability of higher-order networks," *Physica A: Statistical Mechanics and its Applications*, vol. 653, p. 130108, 2024.

[15] A. Dong, X. Mao, R. Vasudevan, and C. Chen, "Controllability and Observability of Temporal Hypergraphs," *IEEE Control Systems Letters*, vol. 8, pp. 2571–2576, 2024.

[16] V. Jurdjevic and I. Kupka, "Polynomial control systems," *Mathematische Annalen*, vol. 272, no. 3, pp. 361–368, 1985.

[17] R. A. Sheldon, "Catalysis: the key to waste minimization," *Journal of Chemical Technology & Biotechnology: International Research in Process, Environmental AND Clean Technology*, vol. 68, no. 4, pp. 381–388, 1997.