



Contents lists available at ScienceDirect

Engineering

journal homepage: www.elsevier.com/locate/engResearch
Control Engineering–Article

The Immense Impact of Reverse Edges on Large Hierarchical Networks

Haosen Cao^{a,#}, Bin-Bin Hu^{a,#}, Xiaoyu Mo^b, Duxin Chen^c, Jianxi Gao^d, Ye Yuan^e, Guanrong Chen^f,
Tamás Vicsek^g, Xiaohong Guan^{h,*}, Hai-Tao Zhang^{a,*}^a Key Laboratory of Image Processing and Intelligent Control, School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China^b School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore 639798, Singapore^c Jiangsu Key Laboratory of Networked Collective Intelligence, School of Mathematics, Southeast University, Nanjing 210096, China^d Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY 12180, USA^e State Key Laboratory of Digital Manufacturing Equipment and Technology, Huazhong University of Science and Technology, Wuhan 430074, China^f Department of Electronic Engineering, City University of Hong Kong, Hong Kong 999077, China^g Department of Biological Physics, Eötvös University, Budapest 1117, Hungary^h MOE Key Laboratory for Intelligent Networks and Network Security, Xi'an Jiaotong University, Xi'an 710049, China

ARTICLE INFO

Article history:
Available online xxxxKeywords:
Synchronizability
Large hierarchical networks
Reverse edges
Information flows
Complex networks

ABSTRACT

Hierarchical networks are frequently encountered in animal groups, gene networks, and artificial engineering systems such as multiple robots, unmanned vehicle systems, smart grids, wind farm networks, and so forth. The structure of a large directed hierarchical network is often strongly influenced by reverse edges from lower- to higher- level nodes, such as lagging birds' howl in a flock or the opinions of lower-level individuals feeding back to higher-level ones in a social group. This study reveals that, for most large-scale real hierarchical networks, the majority of the reverse edges do not affect the synchronization process of the entire network; the synchronization process is influenced only by a small part of these reverse edges along specific paths. More surprisingly, a single effective reverse edge can slow down the synchronization of a huge hierarchical network by over 60%. The effect of such edges depends not on the network size but only on the average in-degree of the involved subnetwork. The overwhelming majority of active reverse edges turn out to have some kind of "bunching" effect on the information flows of hierarchical networks, which slows down synchronization processes. This finding refines the current understanding of the role of reverse edges in many natural, social, and engineering hierarchical networks, which might be beneficial for precisely tuning the synchronization rhythms of these networks. Our study also proposes an effective way to attack a hierarchical network by adding a malicious reverse edge to it and provides some guidance for protecting a network by screening out the specific small proportion of vulnerable nodes.

© 2023 The Authors. Published by Elsevier LTD on behalf of Chinese Academy of Engineering and Higher Education Press Limited Company. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

1. Introduction

Synchronization is an essential coordinated behavior of complex networks that is often encountered in fields such as physics, biology, ecology, and climatology [1–4]. That is, all network nodes agree upon certain features of interest, such as positions, velocities, tendencies, and so forth, or periodically move with the same rhythms, phases, and amplitudes. Synchronization phenomena are evolutionarily advantageous, and members gain benefits in

the form of group protection and survival, mate choice, and foraging [5]. Typical examples include organized fish schools avoiding predator attacks [6,7] and formatted bird flocks saving kinetic energy [8]. A typical synchronization process often encountered in smart electrical power grids is matching different generators to obtain a uniform frequency and angle. In fact, efficient grid synchronization remains a challenging issue for distributed power-generation systems. Abundant manifestations of spontaneous synchronization can be observed in self-coordinated wireless sensors, bursting neurons, pace-matching chemical oscillators, and frequency-locked power generators as well.

By analogy to other forms of collective phenomena, synchronization strongly depends on the topological properties of the underlying interaction network [9]. Due to the appealing nature

* Corresponding authors.

E-mail addresses: xhguan@mail.xjtu.edu.cn (X. Guan), zht@mail.hust.edu.cn (H.-T. Zhang).

These authors equally contributed to this work.

<https://doi.org/10.1016/j.eng.2023.06.011>2095-8099/© 2023 The Authors. Published by Elsevier LTD on behalf of Chinese Academy of Engineering and Higher Education Press Limited Company. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).Please cite this article as: H. Cao, B.-B. Hu, X. Mo et al., The Immense Impact of Reverse Edges on Large Hierarchical Networks, Engineering, <https://doi.org/10.1016/j.eng.2023.06.011>

of networked synchronization, significant efforts [10–13] have been invested lately in exploring the relationship between inter-agent connection and network synchronizability with the assistance of biological experiments, real network data, statistical physics theories, and mathematical/system science analytical tools. So far, due to the widespread presence of hierarchical networks in the natural world, it is common wisdom that the synchronization of hierarchical networks is substantially decelerated or inhibited by extra reverse edges from lower-level nodes to higher-level ones (the hierarchical levels of a network are depicted in Fig. S1 in Appendix A). For example, circadian clocks synchronize biological processes with the day/night clock using molecular mechanisms that include interlocked, transcriptional feedback loops (or reverse edges). A negative reverse edge (i.e., autoregulation) of the evening complex genes constitutes the clock's evening loop, which represses the morning genes [14]. Our study proves this expectation to be invalid by demonstrating that the overwhelming majority of extra reverse edges cannot touch the synchronization rhythms at all. More intriguingly, once made possible, the synchronization of a huge network can be sharply decelerated by over 60% with no more than a single reverse edge overpassing just two nodes. In modern power systems, for example, a disruption in synchronization may cause the malfunction of generators and the outages of power grids with cascading catastrophic failures of power plants, as have been observed in the Western American network in 1996 [15], the Italy-Switzerland network in 2003 [16], and the North America in 2003 [17]. During these large-scale outages, huge reverse power flows and excessive phase angle desynchronization were respectively observed and believed to be the key factors in the aforementioned system collapses [17]. The aim of the current study is to explore the role of reverse edges on abundant natural, social, and engineering hierarchical networks. Our investigation provides some insights into analyzing the mechanism by which the synchronization rhythms can be quantitatively tuned and, reversely, into protecting hierarchical networks from being attacked via the addition of bottom-up links.

To reveal the role of reverse edges, we ignored the loops of hierarchical networks, focusing instead on directed acyclic networks. Hierarchical networks are usually used to describe causal or hierarchical relationships between different systems interacting with each other; correspondingly, there are many kinds of definitions of hierarchy [18,19]. For example, the leader-follower structures of migrating bird flocks [20–22] form hierarchical communication networks when flying in V-formation [23]. The generating stations, substations, loads, and so forth of smart grids in different hierarchies form a network in which directed power or information flows connect the network components. Moreover, in military multivehicle formations, only a few vehicles are manned agents that act as leaders, while the rest are unmanned followers. Each unmanned vehicle follows its front neighbors by means of onboard sensors (e.g., cameras); thus, the communication network forms a hierarchical network [24].

Typical topologies of hierarchical networks include chains, trees, grids, and more general single-rooted hierarchical graphs. Widely existing biological hierarchical network examples include migrating bird flocks [20–23], mammal groups [25], and even microbiological yeast regulation networks and gene ontology (GO) [26,27]. Moreover, there are abundant industrial and engineering hierarchical networks, such as water-supply networks [28], wind farms [29], smart grids [30], instruction scheduling in low-level program optimization [31], unmanned vehicle formations [32], combinational circuits [33], compiler design [34], data-flow programming languages [35], and efficient search algorithms [36]. The literature even contains examples of huge-scale hierarchical networks in fields ranging from biology [37–39], neuroscience [40], and genetics [41] to economics [42].

It is inevitable that the performance and features of hierarchical networks are often affected by additional reverse edges, which sometimes significantly influence the dynamics of the entire network. In social networks, feedback opinions from lower-level individuals to leaders favor the optimization of a group decision or democracy. In power networks, redundant cycles produced by link addition are practical for protecting fragile nodes and strengthening network connectivity and robustness. On the other hand, in what is known as Braess's paradox, adding new links may not promote but rather destroy the synchrony of specific networks, incurring power outages or traffic congestion [43,44]. In many natural hierarchical networks, such as pigeon flocks [22] and horse herds [45], the followers may detect food resources, new habitats, or clues indicating predators. Thus, bottom-up emergent information flows, that is, reverse edges help the leaders make better decisions for the benefit of the whole network. In supply chains, feedback from retailers and customers is indispensable for producers and suppliers in order to enhance circulation efficiency. Therefore, such real-life situations motivate investigations into the role of reverse edges in the huge volume of hierarchical networks.

To this end, we conduct an extensive investigation on several typical medium- or large-scale real hierarchical networks. For some of these cases, counterintuitively, our extensive analytical and numerical investigation revealed that the majority of the massive possible reverse edges do not affect the entire hierarchical network at all, so we denote them as silent reverse edges. As an example, for a 2373-node/20 812-edge/89-level hierarchical Bitcoin network slim [46], the proportion of silent reverse edges is more than 64.48%; in another denser giant 29 383-node/56 377-edge/17-level hierarchical network, (i.e., a GO [26,27]), the proportion is as high as 99.92%. In contrast, the reverse edges are denoted as active if they have an influence on the synchronization process. We have identified some specific classes of network structures along which the active reverse edges can be promptly extracted without difficulty.

Next, we focus on the quantitative role of reverse edges on real natural/social/engineering hierarchical networks. Again, counterintuitively, the effect of reverse edges turns out to be independent of their locations or the network size. Instead, their effect is solely determined by the local subnetwork containing the additional reverse edges, especially the in-degrees of the landing nodes (i.e., the ending nodes of the reverse edges). More precisely, a larger in-degree landing node (i.e., the ending nodes of the reverse edges) implies a slower synchronization. The underlying reason is that merging more information flows from other branches into the landing nodes contributes to diluting the effect of reverse edges.

From a scientific point of view, this study refines our knowledge, making it possible to accurately screen out the very few active reverse edges with significant influence within abundant large-scale hierarchical networks. With respect to engineering applications, the present investigation might shed some light onto protecting, attacking, or regulating large-scale natural/social/engineering hierarchical networks, by means of precisely located additional reverse edges. More specifically, this work reveals an effective way to attack a network (in terms of slowing down its synchronization process) by precisely positioning a few specific malicious reverse edges in a huge network. Conversely, our work also provides guidance for protecting a network by screening out specific paths that are vulnerable to reverse edge attacks but usually constitute a minuscule proportion of the network paths. In brief, our paper contributes to paving the way for the newly revealed principle of hierarchical networks toward a large variety of promising applications in real natural, social, and engineering hierarchical networked systems.

2. Materials and methods

2.1. Experiment

Using a group of 17 unmanned surface vehicles (USVs) to mimic a horse-herd network, we were able to experimentally check the quantitative role of reverse edges on hierarchical networks. Therein, for simplicity, the USV dynamic was regarded as a first-order dynamic system by accurately regulating the kinematic model [47]. In this way, the USV group synchronization performance of moving directions could be quantified and the revealed role of additional reverse edges could accordingly be verified.

As shown in Figs. S2-S5 in Appendix A, the multi-USV system consists of 17 40 cm-long USVs, a motion-capture module, a monitoring and control module, and an indoor pool. The motion-capture module consists of eight Optitrack Flex 3 infrared cameras (NaturalPoint Corporation, USA) and Motive 2.2 data-collecting software (NaturalPoint Corporation) which are used to detect, trace, and analyze the moving trajectories of the USVs with an accuracy of ± 0.3 mm. Main components and hardware in this experiment are listed in Table S1 in Appendix A. With the communication topology of the horse herd shown in Fig. 1 [45], each USV receives the velocities and positions of its neighboring USVs sent by an embedded monitoring module via a wireless module with a sampling frequency of 100 Hz. Furthermore, the module calculates the directional synchronization control signal.

Mathematically, we denote $\varphi_i \in [0, 2\pi], i = 1, 2, \dots, 17$ as the direction of the i th USV, where i is the index of a specific USV. Using the kinematic model, the directions of all USVs will converge

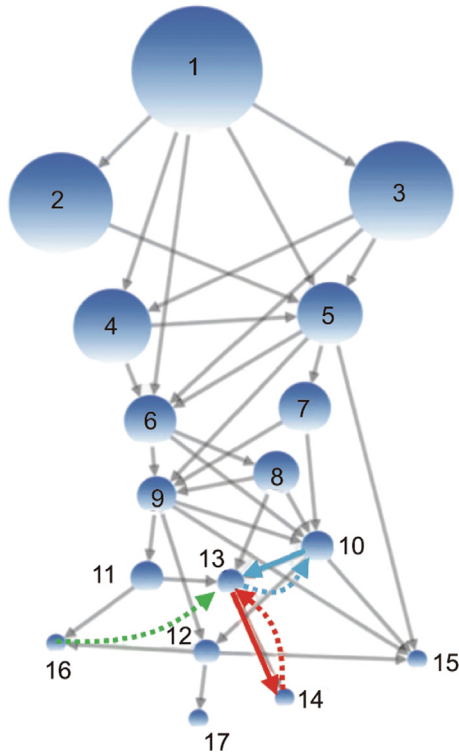


Fig. 1. The inter-communication network of a horse herd [45]. The proximity of a node's level to the root node is indicated by its size. Here, the necessary condition for active reverse edges—that is, the formation of loops—is illustrated. For example, the reverse edge $16 \rightarrow 13$ (the green dashed arrow) could not form any new loop, so the Laplacian spectrum remains unchanged, and it is a silent edge. Both edge $14 \rightarrow 13$ (the red dashed arrow) and edge $13 \rightarrow 10$ (the blue dashed arrow) form loops. However, $14 \rightarrow 13$ is active, whereas $13 \rightarrow 10$ is not. Therefore, forming loops is not a sufficient condition for the activity of a reverse edge.

to the initial direction φ_0 of the leading USV, representing the root node in the horse-herd network. We consider the USV group to have achieved synchronization when the difference in direction between any USV and the leading USV is less than a certain threshold 0.05 rad—that is, $\forall i, |\varphi_i - \varphi_1| \leq 0.05$ rad and φ_1 denotes the direction of leading USV.

As is shown in Figs. 2(a) and (b) and Fig. S6 in Appendix A, during the experiments, the USVs were initially randomly distributed on a rectangle section of $[-3000, 1000] \text{ mm} \times [-1000, 1000] \text{ mm}$ of the pool with velocities of zero. The moving directions of all the USVs were then synchronized. To exhibit the influence of reverse edges more vividly, we conducted experiments under three scenarios—namely, ① no reverse edges, ② a silent reverse edge ($17 \rightarrow 11$), and ③ an active reverse edge ($17 \rightarrow 12$), as represented in Figs. 2 (c)–(e). In order to calculate the convergence times of the synchronization processes, each was taken as an average over three independent runs. It was observed that an active reverse edge (Fig. 2(e) with red trajectory ends) effectively slowed down the synchronization speed, whereas others did not influence it (Figs. 2(c) and (d) with green trajectory ends). In other scenarios, the synchronization processes are compared in Fig. S7 in Appendix A.

2.2. Algorithm

We formally consider an N -node single-rooted hierarchical network \mathbb{G} in topological ordering, with a lower triangular Laplacian $\mathbf{L}_0 \in \mathbb{R}^{N \times N}$, where \mathbb{R} is the set of real numbers. We assume that all networks mentioned here are unweighted. Let \mathcal{V} be the node set of \mathbb{G} and $\mathcal{H}(v) \in \mathbb{R}^+$, $v \in \mathcal{V}$ be the hierarchical level of node v ; moreover, in this work, we define $\mathcal{H}(v)$ as the function that maps node v to the longest distance from the root node to it (Fig. S1 and Section S1 in Appendix A). We add a reverse edge from node ℓ to node h with $r = \mathcal{H}(\ell) - \mathcal{H}(h) > 0$; $h, \ell \in \mathcal{V}, h \neq \text{root}$, where r denotes the surpassing range, and “root” denotes the root node of \mathbb{G} . For this reverse edge, denoted as (h, ℓ) , a subnetwork \mathbb{G}_1 is uniquely determined whose node set contains h, ℓ , and the nodes of all paths from h to ℓ . Let Ω be the node set of \mathbb{G}_1 and $n := |\Omega|$ the cardinality of set Ω . Mathematically, once the reverse edge (h, ℓ) is added, the Laplacian matrix \mathbf{L}_0 of network \mathbb{G} becomes $\bar{\mathbf{L}}_0$, and subnetwork \mathbb{G}_1 corresponds to a block matrix $\mathbf{L} = [l_{ij}] \in \mathbb{R}^{n \times n}$ on the diagonal of $\bar{\mathbf{L}}_0$, with l being some entry of \mathbf{L} . That is:

$$\bar{\mathbf{L}}_0 = \begin{bmatrix} L_1 & 0 & 0 \\ * & L & 0 \\ * & * & L_2 \end{bmatrix} \quad (1)$$

$$\mathbf{L} = \begin{bmatrix} l_{1,1} & 0 & \cdots & \cdots & -1 \\ -1 & l_{2,2} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \cdots & \cdots & \cdots & l_{n-1,n-1} & 0 \\ \cdots & \cdots & \cdots & -1 & l_{n,n} \end{bmatrix} \quad (2)$$

where the square matrix \mathbf{L} contains all path nodes from h to ℓ . It thus becomes an interesting problem to screen out active reverse edges merely according to the in-degrees of the nodes in the involved subnetwork \mathbb{G}_1 , in order to avoid the time-consuming spectrum calculation of huge Laplacian matrices. To this end, we have designed the active reverse edge searching Algorithm 1, which is composed of two steps: path searching and node searching.

Step 1: path searching. Search the path(s) from node h to node ℓ in subgraph \mathbb{G}_1 . Denote Γ_i as the i th such path and c as the number of such paths. Then $c \geq 1$ is a necessary condition for determining the active reverse edge.

Step 2: node searching. Search the node(s) with waist bunching and funnel bunching in subgraph \mathbb{G}_1 , corresponding to three conditions in Algorithm 1. As is discussed in Section 3.2, these nodes indicate the information flow within \mathbb{G}_1 and determines the activity of a reverse edge.

With Algorithm 1 (more details are provided in Section S2 in Appendix A), more than 99.4% of the active reverse edges in the real networks (Section 3.3) can be screened out without calculating the eigenvalues of the massive matrix $\bar{\mathbf{L}}_o$. In essence, the computational complexity of computing and evaluating a discriminant matrix has been sharply decreased from $\mathcal{O}(N^3)$ to $\mathcal{O}(N)$ with the assistance of a typical classical QR algorithm [48]. Still, the remaining few active edges ($\leq 0.6\%$) create a dilemma, due to the sophisticated counterbalance between the exterior and interior information flows endowed by the in-degrees in \mathbb{G}_1 . Fortunately, in most application scenarios, it is sufficient to screen out an overwhelming majority of active reverse edges, as is done in the present study.

Algorithm 1. Active reverse edge searching algorithm.

Require: Graph Laplacian $\bar{\mathbf{L}}_o = [\bar{l}_{ij}] \in \mathbb{R}^{N \times N}$
Initialize $\mathcal{H}(v)$ that maps node v to the longest distance from root to it
for $(\ell, h) \in \{1, \dots, N\} \times \{1, \dots, N\}$, $h \neq \text{root}$ that satisfies $\mathcal{H}(\ell) > \mathcal{H}(h)$ **do**
 Search paths from node h to node ℓ and construct the path set $\Theta_{\ell,h}$ as $\Theta_{\ell,h} = \{I_i = \{h, \dots, \ell\} | i = 1, \dots, c\}$
 $\Omega = \cup_{i=1}^c I_i = \{v_1, v_2, \dots, v_n\}$, initialize $\mathbf{L} = [l_{ij}] \in \mathbb{R}^{n \times n}$ as zero matrix
 for $(i, j) \in \{1, \dots, n\} \times \{1, \dots, n\}$ **do**
 $l_{ij} = \bar{l}_{ij}$
 end for
 $l_{1,n} = -1, l_{1,1} \leftarrow l_{1,1} + 1, \forall i \in \{1, \dots, n\}, z_i = l_{i,i} - \sum_{j=1, j \neq i}^n |l_{ij}|$
 if $\Theta_{\ell,h} \neq \phi$ and (Condition 1, Condition 2, or Condition 3 holds) **then**
 The reverse edge $\ell \rightarrow h$ is active
 else The reverse edge $\ell \rightarrow h$ is silent
 end if
end for

Here, $\Theta_{\ell,h}$: path set in the subnetwork \mathbb{G}_1 . Ω : node set in the subnetwork \mathbb{G}_1 . z_i : exterior information flow of the i th node. ϕ : the empty set; the three conditions are given below

Condition 1 (strict waist bunching): $\exists i \in \{1, \dots, n\}$, such that $l_{i,i} = 1$.

Condition 2 (generalized waist bunching):

$$\forall i \in \{1, \dots, n\}, z_i \leq 1 \text{ and } \exists p \in \{1, \dots, n\}, \text{ such that } z_p = 0.$$

Condition 3 (funnel bunching): $\exists i \in \{1, \dots, n\}, z_i \geq 2$ and $\forall p \in \{1, \dots, i-1\}, z_p \leq 1$ and $\forall q \in \{i+1, \dots, n\}$, such that $z_q = 0$.

3. Results

3.1. Only a small portion of reverse edges in large-scale hierarchical networks are active

Hierarchical networks widely exist in natural biological groups, gene networks, and engineering systems. It is natural to consider that adding a reverse edge inserts cycles in the original network in such a way that it impedes the synchronization process. Counterflow scenarios sometimes bring valuable applications or serious

and even disasters, such as liquid counterflow in a casing heat exchanger or pedestrian counterflow inducing a transition from a free to congested state in traffic. Thus, it is crucial to investigate the counterflow effect from a network viewpoint. Intuitively, it might be common sense that most of the reverse edges from a lower level to a higher one influence the synchronization of the hierarchical network. Surprisingly, however, we found out via exhausting all possible reverse edges that just a few of them are actually capable of changing the synchronization rhythm for a large volume of hierarchical networks.

First, for arbitrary hierarchical networks with no loops, reverse edges are defined as connections from lower levels to higher ones. Here, the level sequence number of each node is defined as the furthest distance to the root. Without loss of generality, we take a representative hierarchical graph—that is, an inter-communication network of a horse herd, given in Fig. 1—as an example. Evidently, it is imperative to investigate the quantitative role of reverse communication edges on real horse herds. For this study, we used 17 mini-USVs to mimic the synchronization of a horse herd, where each vessel imitates a horse cluster to align to its front cluster(s) (i.e., the immediate leader(s)).

As shown in Fig. 2(a), during each experiment, the USVs are initially randomly distributed on a pool. With the inter-USV communication topology in Fig. 1, the final moving directions of the USVs are synchronized, as shown in Fig. 2(b). Significantly, we record different time intervals taken to reach synchronization with different reverse edges. In this way, the role of different reverse edges becomes quantitatively comparable. For more details of the experiments, refer to Section S3 in Appendix A.

Counterintuitively, it can be observed in Fig. 2 that, for such a small horse-herd topology, only 63% of all the possible reverse edges were able to affect the synchronization process (18.0 s in Fig. 2(e) for a reverse edge 17→12 vs 7.5 s in Fig. 2(c)), so we denoted them as active reverse edges. The remaining 37% of the reverse edges are silent, as they did not influence the synchronization speed at all (7.5 s, like a silence reverse edge 17→11 vs 7.5 s; see Figs. 2(c) and (d)).

A possible concern is that the active reverse edges seem to build the majority of all possible reverse edges. According to extensive investigations on real hierarchical networks, however, it appears that the proportion of active edges decreases sharply with increasing network size. Take a typical huge hierarchical network—that is, a GO consisting of 29 323 nodes and 56 377 edges—for discussion [26,27]. According to a detailed analysis of the synchronization processes with all possible reverse edges (over 380 million), only less than 0.08% are active. In order to obtain insight into this phenomenon, a thorough understanding of the types of reverse edges that can tune the corresponding network dynamics and the synchronization rhythms is necessary. At this point, we conclude that three essential concerns need to be addressed in this section: ① How many active reverse edges are there in each hierarchical network? ② How can one accurately locate them with high efficiency? ③ To what extent is an active reverse edge able to slow down the synchronization process?

However, due to the water environment, it is difficult to ensure that all the initial states are identical for the 17 USVs in each independent run of experiments. Once the direction and position of any vehicle are adjusted, the emanating wave will disturb the others. Therefore, we only try to set the initial states of the USVs to be as arbitrary as possible in each independent run. In addition, a simulation is presented in Fig. 3 to depict the slowing down effect of an active reverse edge in an ideal environment. Consider the dynamics of a first-order integrator $\dot{\mathbf{x}} = -\mathbf{L}_o \mathbf{x}$, where \mathbf{x} denotes the state of system. The synchronization process is apparently different with silent and active reverse edges for identical initial states.

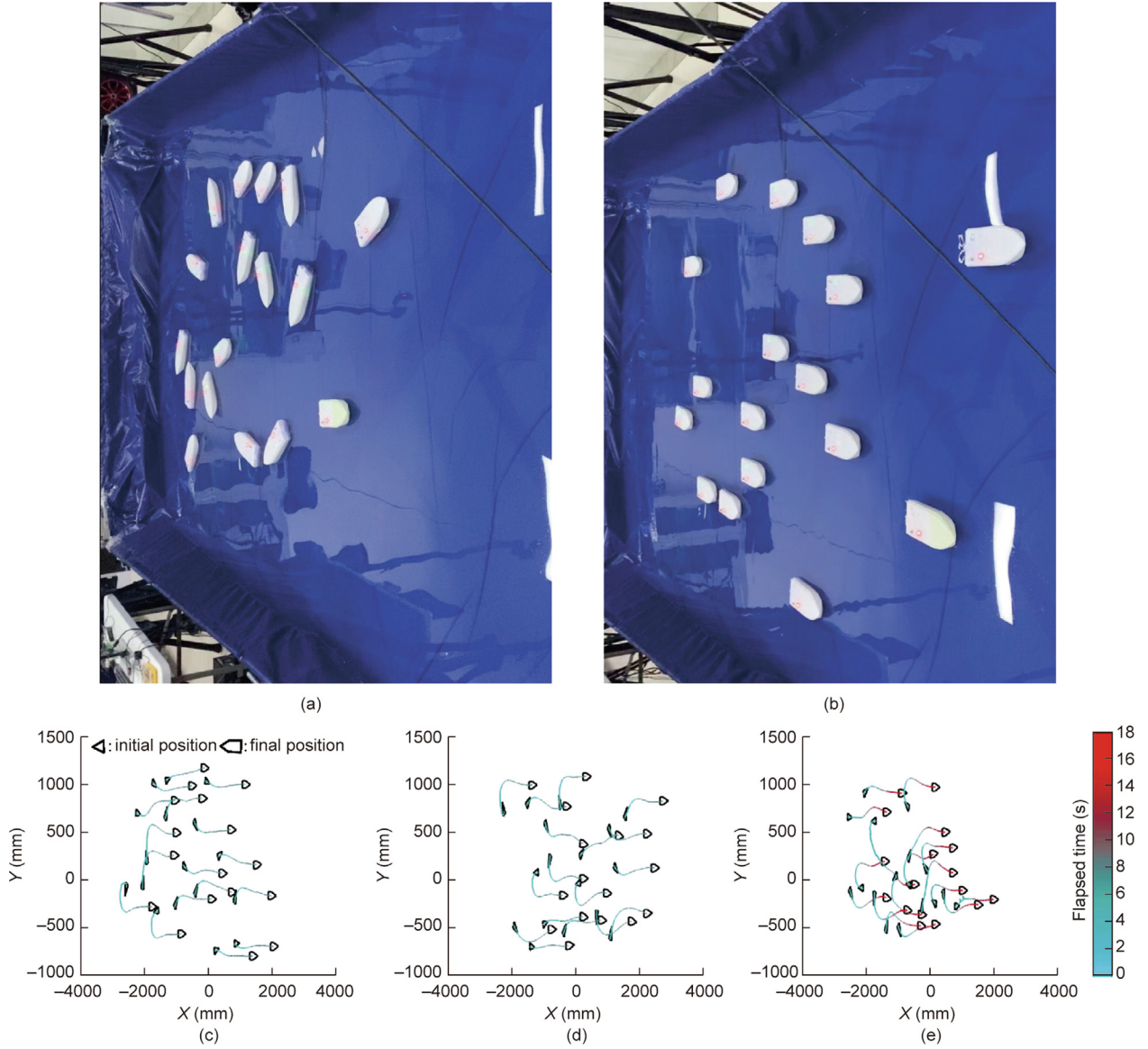


Fig. 2. Snapshots and moving trajectories, as well as the corresponding synchronization periods of the 17-USV system. The horizontal and vertical axes form the Cartesian coordinate system of the water surface plane. The color of the points on the trajectory represents the elapsed time from the beginning of the experiment, referring to the legend on the right. (a, b) Snapshots of the initial direction stage and synchronized direction stage. Adding certain active reverse edges to the network could extend the period of the synchronization process. For the full experimental record, see [Videos S1](#) and [S2](#) in [Appendix A](#). (c) Trajectories with no reverse edge case; synchronization period is 7.5 s. (d) Trajectories with a silent reverse edge (17→11); synchronization period is 7.5 s. (e) Trajectories with an active reverse edge (17→12); synchronization period is 18 s. Only 63% of the reverse edges slowed down the synchronization process; the others were silent. Here, the moving direction synchronization threshold is 0.05 rad.

3.2. Where are the rare active reverse edges?

To exhaustively and precisely locate the active reverse edges for a huge variety of hierarchical networks, an index is needed to quantify the synchronization speed. It is well accepted in the literature that the Fiedler value $\lambda_2(\mathbf{L}_0)$ (abbreviated as λ_2 ; i.e., the second smallest eigenvalue of the graph's Laplacian \mathbf{L}_0) is an option, as it quantifies the connection strength of graphs. Importantly, it can be observed in [Fig. 4](#) that the correlation coefficient $R = 0.8368$ and the significant p -value $p < 10^{-3}$ in the Pearson correlation test [49], which implies that the synchronization speed is strongly correlated with the algebraic connectivity $\lambda_2(\mathbf{L}_0)$ of the real 17-USV platform with the horse-herd topology backbone. Accordingly, we adopt λ_2 as the index to quantify the graph synchronization speed, in the sense that the smaller the index λ_2 is, the more slowly the network is able to attain synchronization.

This situation raises a point regarding what kind of additional reverse edges can possibly change λ_2 . To address this question, technical details were taken into account, and it was observed that, if no new loop was formed by the additional edge, this edge would be silent. As shown in [Fig. 1](#), the green additional edge 16→13 (the green dashed arrow) was unable to form any new loop. Furthermore, the Laplacian spectrum λ remained unchanged in most cases, as the original Laplacian of the entire network can still be rewritten into a lower triangular one simply by changing the sequence of nodes. In other words, forming a new loop becomes a necessary condition (but not a sufficient one) for active reverse edges. To illustrate this issue, take the two reverse edges 14→13 (the red dashed arrow) and 13→10 (the blue dashed arrow) in [Fig. 1](#) as an example. Both have formed loops, but the former is active whereas the latter is not. Therefore, the formation of new loops was not an assurance that the reverse

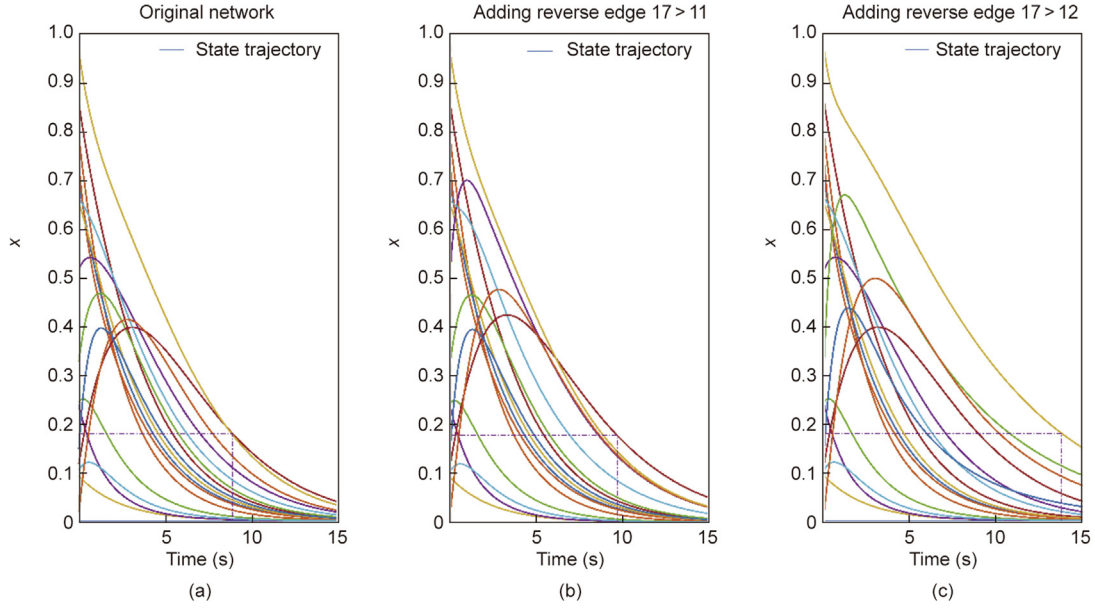


Fig. 3. Simulated state trajectories of the horse-herd network synchronizing with identical initial states. Under the dynamics of a first-order integrator, the states of all nodes in the system converge to the state of the root node. With identical initial states, the synchronization process is merely affected by the addition of a silent reverse edge but is clearly slowed down by the addition of an active one. The dashed line represents the time required for the state to converge to the same level. (a) State trajectory without additional edge; (b) state trajectory with a silent reverse edge (17→11); (c) state trajectory with an active reverse edge (17→12).

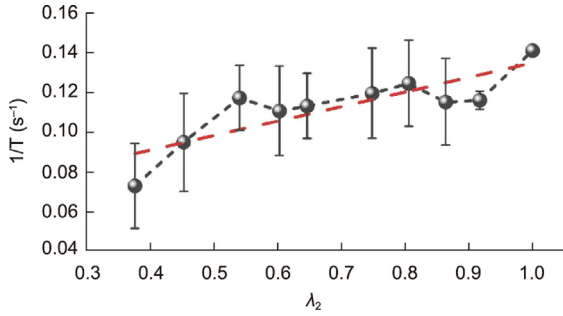


Fig. 4. Experimental validation of the correlation between the Fiedler value $\lambda_2(L_0)$ and the convergence speed $1/T$ of the real 17-USV platform with the horse-herd topology backbone shown in Fig. 1. $\lambda_2(L_0)$ is the minimal real part of the nonzero eigenvalues of Laplacian matrix L_0 . T denotes the convergence time of the network synchronization process. It is observed that the correlation coefficient is $R = 0.8368$ and the significant p -value is $p < 10^{-3}$ in the Pearson correlation test, which verifies that λ_2 is qualified as a quantitative index of network synchronizability.

edges were active, and it remains a challenging issue to precisely locate active reverse edges.

To this end, consider an N -node single-rooted hierarchical network \mathbb{G} in topological ordering, with a lower triangular Laplacian L_0 . Assume that there exists at least one path from node h to node ℓ , which forms a subnetwork \mathbb{G}_1 consisting of $r + 1$ nodes. Our investigation revealed that the in-degrees of the nodes in subnetwork \mathbb{G}_1 substantially influence the effectiveness of an additional reverse edge (see Section 2 for details). In particular, in \mathbb{G}_1 , the node in-degree could be divided into interior and exterior in-degrees according to the starting nodes in subnetwork \mathbb{G}_1 and the complementary set of \mathbb{G}_1 , respectively. Thus, if all the nodes in \mathbb{G}_1 have at least one exterior in-degree, the reverse edge is surely not able to influence the synchronization process of the network, as the algebraic connectivity λ_2 is equal to or greater than 1 by the Gershgorin circle theorem. It is therefore necessary to identify an active reverse edge on a prior basis if there exists at least one node in subnetwork \mathbb{G}_1 without an exterior in-degree.

To explain the effect of an active reverse edge, the in-degree of a node is denoted as the width of the information channel at this point, as the width varies along the information flow. Such a phenomenon is reminiscent of a bunching effect, as shown in Fig. 5(a), which means that there exist some nodes that tighten up the information channel and thereby activate the counterflow effect of the reverse edge. Our investigation reveals that there are generally two kinds of bunching effects: more precisely, waist bunching and funnel bunching. As shown in Fig. 5(b), waist bunching denotes the scenario where the information channel is tightened up by at least one node with an interior-only in-degree equal to 1 (namely, a streaming node). Therefore, the counterflow influence of the reverse edge has not been diluted, due to the existence of the narrowest channel width of 1. However, further analysis of waist bunching shows that there exists at least one node with an interior-only in-degree and all the other nodes in \mathbb{G}_1 with no more than 1 exterior in-degree. Despite the existence of exterior in-degree(s), the counterflow influence of the reverse edge is not completely neutralized by the limited exterior nodes.

As shown in Fig. 5(c), funnel bunching denotes the case where the information channel is first stretched out and then suddenly narrowed down in the subnetwork \mathbb{G}_1 . Therein, the upper nodes bear nonnegligible exterior in-degrees and the bottom nodes bear interior-only in-degrees. In such a situation, the counterflow influence of the reverse edge would not be completely neutralized either. For more details on an active reverse edge detection algorithm via the aforementioned bunching effect analysis, we refer to Algorithm 1.

To verify the proposed Algorithm 1, we examined the mentioned huge GO, which is separated into 17 hierarchical levels according to their distances from the single root. It can be observed that more than 99.5% of the effective reverse edges among the 379 260 333 possible ones can be promptly screened out without any exhaustive process. The remaining 0.5% of the reverse edges can be evaluated directly by calculating their Fiedler value—that is, the smallest real part of the corresponding discriminant matrices' eigenvalues (see Section S1 for the definition of a discriminant matrix).

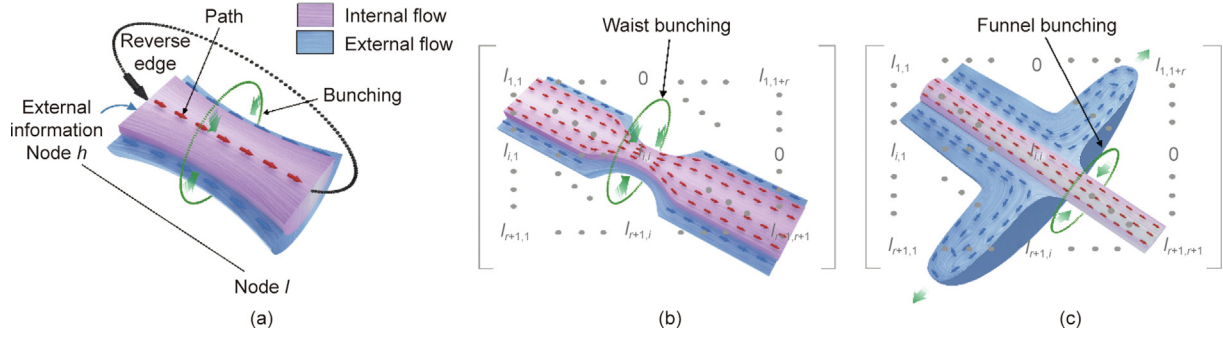


Fig. 5. (a) Illustration of the effect of information channel bunching on hierarchical networks. Here, a reverse edge is added from node l to node h . The nodes in the red region only transfer internal information, as they contain interior-only in-degrees. In contrast, the nodes in the blue region contain exterior in-degrees and act as external information channels. The role of bunching tightens the information channel so as to activate the involved reverse edges. (b, c) There are two typical bunching nodes: (b) waist bunching, where the information channel is bunched as a waist at the subnetwork \mathbb{G}_1 ; and (c) funnel bunching, where the information channel is stretched out and then suddenly narrowed down at the subnetwork \mathbb{G}_1 . Here, l_{ij} refers to the elements in the i th row and j th column of the corresponding block matrix \mathbf{L} of the subnetwork \mathbb{G}_1 .

3.3. The level where the synchronization process can be slowed down by an active reverse edge

Knowing the locations of the active reverse edges, it could be expected that it would be possible to finely tune the network synchronizability by delicately adjusting the tiny proportion of active reverse edges; or, conversely, by deliberately defending the few active reverse edges from external attacks. However, the degree to which the synchronization process can be slowed by active reverse edges is not known. For simplicity, let us start from a chain network. Although an active reverse edge surpassing only one node can reduce the synchronization speed by a surprisingly high rate of 61.8%, this is obviously not the maximal slowing-down effect. Surpassing more chain nodes contributes to further slowing down the synchronization speed.

Next, we further quantitatively investigate the impact of the active reverse edge for 16 representative kinds of real hierarchical networks ranging from tens of nodes to tens of thousands of nodes. The quantitative effects of active reverse edges on the synchronization process of these networks are shown in Fig. 6 and summarized in Table 1. We use percentage of reduction (PR) and average PR (APR) to measure the extent to which the convergence speed is affected by a reverse edge; that is, $PR = (\lambda_2 - \tilde{\lambda}_2) / \lambda_2$, where λ_2 and $\tilde{\lambda}_2$ denote the Fiedler value of a network before and after this reverse edge is added, respectively. APR is the average of the PR of reverse edges that have the same average in-degree \bar{d}_{in} and surpassing range r .

It can be observed that the ratio of effective reverse links (ERLs) in all the networks is not larger than 36%, except for the Horse network Fig. 6(b). Interestingly, the effect of active reverse edges on the synchronization process has nothing to do with the network size or its specific position. Instead, the effect is merely determined by the subnetwork containing additional active reverse edges. In other words, the transmitting capability or synchronization process of any other edges or nodes outside the subnetwork \mathbb{G}_1 is not influenced at all. This is due to the fact that the Laplacian \mathbf{L}_o of a hierarchical network \mathbb{G} is a lower triangular matrix; hence, the active reverse edge only affects the involved sub-triangular entries of \mathbf{L}_o (Section S1).

It should be noted that, as shown in Fig. 6, only a single active reverse edge can abruptly slow down the convergence speed by more than 60%, thus negatively affecting the synchronization process of the network regardless of how large the network is (e.g., the huge GO). Moreover, the slowing-down effect is merely influenced by the average in-degree in the subnetwork \mathbb{G}_1 , in the sense that the smaller the average in-degree is, the more slowly the synchro-

nization is achieved. This phenomenon can be explained by the fact that the impact of the active reverse edge is diluted by other branches merging into an identical subgraph. It is similar to a scenario in which, when several streams merge in confluence with the mainstream, the influence of each confluent is diluted. This observation is useful for finetuning the synchronization process of a network by modulating the in-degree of the nodes.

Accordingly, using Algorithm 1, it is possible to precisely screen out the few active reverse edges and quantify the intensity of the reverse edge impact by focusing on a much smaller subnetwork. These findings refine our knowledge of hierarchical networks, which can be mathematically analyzed for some regular networks.

Here, an appealing fact is noticeable: namely, that the golden ratio of the synchronization reduction rate $\lambda^* = 1 - (\sqrt{5} - 1)/2 \approx 0.382$ is the minimal $\tilde{\lambda}_2$ with respect to d_{in} if the in-degree of the landing node of the active reverse edge is 1 and the maximal $\tilde{\lambda}_2$ with respect to r if the surpassing range of the active reverse edge is also 1. This is due to the fact that the changed $r + 1$ eigenvalues are determined by an explicit eigenpolynomial $\lambda^2 - \lambda - 1 = 0$ with a golden ratio solution. From the perspective of applications, the revealed golden ratio for the algebraic connectivity λ_2 is a kind of optimal spectrum value of the networks under investigation. It may give a balanced point between the energy cost and the optimal moving trajectory for networked systems [50] as well.

4. Discussion

Reverse edges representing feedbacks from lower-level nodes to higher-level ones usually play crucial roles in natural, social, power, and engineering hierarchical networked systems. The collective entities in a network cooperatively interact with each other, driving the system to a stationary state [6]. Such reverse edges may be messages about food, predators, or habitats sent by followers in animal groups; opinion feedback in social networks; power flows in distributed power-generation systems; artificially added tuning connections in microcosmic cell tissues; or macrocosmic cooperative robotic systems. The significant influence of a single reverse edge has been identified in such universal natural, social, and engineering hierarchical networked systems. More precisely, through both extensive statistical investigation and strict mathematical analysis, it can be concluded that the role of reverse edges depends solely on the subnetwork containing them. Only the reverse edges located at positions with in-degrees satisfying some specific “bunching” conditions concerning the involved subnetworks actu-

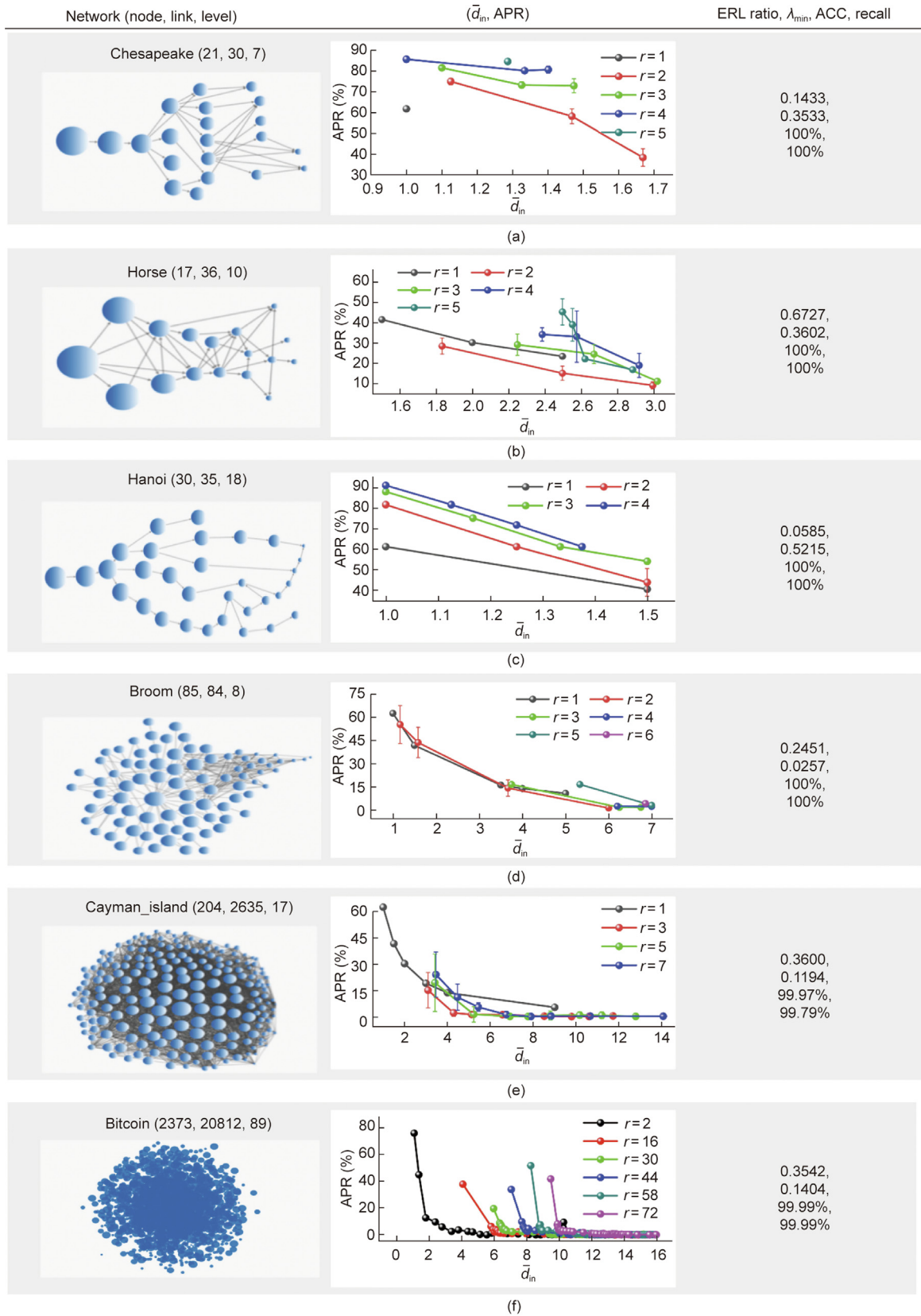


Fig. 6. The effect of reverse edges on six typical hierarchical networks with different scales. The proximity of a node's level to the root node is indicated by its size. \bar{d}_{in} represents the average in-degree of the subnetwork G_r , and r represents the surpassing range—that is, the difference in the hierarchical levels of nodes h and l . APR: the average percentage of reduction. The percentage of reduction (PR) of a reverse edge refers to the ratio of the decrease in convergence speed to its original value. The label on the vertical axis for one point represents the average PR for all reverse edges satisfying the given \bar{d}_{in} and r . The effective reverse link (ERL) ratio denotes the ratio of effective reverse links, and λ_{min} denotes the minimum value of the $\lambda_2(L_0)$ of the entire network with different additional reverse edges. The second column depicts the correlativity of average in-degrees corresponding to an active reverse edge and its impact on the synchronization process. To measure the precision of the proposed Algorithm 1, ACC represents the accuracy ratio of correctly identified reverse edges to all reverse edges. Recall represents the ratio of correctly identified active reverse edges to all active reverse edges.

Table 1
Effect of reverse edges on an additional seven typical hierarchical networks with different scales.

Network (node, link, level)	ERL ratio	λ_{\min}	ACC (%)	Recall (%)
Aspergillus (38, 37, 3)	0.0345	0.3820	100.00	100.00
Candida (39, 38, 4)	0.0381	0.2451	100.00	100.00
Generic-GO-slim (72, 73, 4)	0.0189	0.2451	100.00	100.00
Meta-Gene (43, 46, 5)	0.0521	0.2451	100.00	100.00
Plant-slim (45, 51, 4)	0.0454	0.2451	100.00	100.00
Yeast (101, 101, 3)	0.0095	0.3820	100.00	100.00
GO (29, 323, 56, 377, 17)	0.0008	0.0643	99.99	99.55

ally influence the network synchronizability. Such a small subnetwork can finely tune the Fiedler value $\lambda_2(L_0)$ in order to influence the synchronization rhythms of entire large-scaled networks simply by tweaking the parameters of the involved subnetwork, such as its average in-degree. Furthermore, it has been revealed that specific parameter combinations of reverse edges—such as surpassing ranges and the in-degree of the landing node—can produce the golden ratio for the algebraic connectivity of a hierarchical network. This optimal spectrum of network synchronization and coordination is beneficial for optimizing network performance and balancing the energy cost and optimal orbit of networked systems, such as atom nuclei [50].

We have demonstrated that two nesting reverse edges with joint surpassing nodes bring in a stronger slowdown effect than a single reverse edge if they pass more waist bunching. Moreover, the reduction of the synchronization speed is strongly consistent with the summation of the surpassing range of the added reverse edges in a chain network. Such a case can be extended to the stem-based [51] nature of tuning the synchronizability in any hierarchical network. It is still an open challenge to quantitatively understand the effect of adding two or more reverse edges to a general network. If the added reverse edges have no joint nodes, the one with the most significant effect plays the key role. However, research is still underway on situations in which the reverse edges have joint bunching effects, since the surpassing range, in-degree, and position have a complicated coupling influence on the effect of the reverse edge; hence, no niche analysis tools in this regard exist. For nonlinear dynamical networked systems, we suggest that a linearization design is a viable approach to make the system compatible with the present analytical method. However, the influence mechanisms of the network structure, initial states, and nonlinearity formats on the synchronizability are far from clear at present.

From a scientific point of view, this study refines and enhances the existing knowledge about the role of reverse edges. For example, our investigation reveals that most of the reverse edges of abundant large hierarchical networks are inactive, as they do not influence the synchronization process and cooperation capability of the network at all. Moreover, our approach contributes by locating rare active reverse edges and quantitatively revealing their effect on the cooperation performance. From the aspect of engineering applications, the present investigation sheds some light on attacking or defending large-scaled hierarchical networks by identifying their precious subnetworks with bunching effects (e.g., stems).

5. Conclusions

In this work, we focused on the effect of the addition of a reverse edge on the synchronization process of a hierarchical network. It was revealed that, regardless of network scale, there exist few active reverse edges that can influence the synchronization process of the network due to their bunching effect on the involved information flow. These active reverse edges can individually slow down the synchronization of a huge hierarchical network by over

60%, and the magnitude of this slow-down effect was found to be highly dependent on the average in-degree of the surpassing subnetwork. According to the theoretical model of the bunching effect, a searching algorithm of precious reverse edges was also proposed and validated with high accuracy and efficiency. Moreover, a USV-group direction alignment experiment and extensive simulations on real hierarchical networks were conducted to support these findings and analyses. The present work may provide some insights into protecting, attacking, or regulating the synchronization process of entire hierarchical networks, such as multi-robot systems, power grids, or wireless sensor networks, merely by tweaking the reverse edges surpassing a few upwards nodes.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (62225306, U2141235, 52188102, and 62003145), National Key Research and Develop Program of China (2022ZD0119601), Guangdong Basic and Applied Research Foundation (2022B1515120069), and the Science and Technology Project of State Grid Corporation of China (5100-202199557A-0-5-ZN).

Compliance with ethics guidelines

Haosen Cao, Bin-Bin Hu, Xiaoyu Mo, Duxin Chen, Jianxi Gao, Ye Yuan, Guanrong Chen, Tamás Vicsek, Xiaohong Guan, and Hai-Tao Zhang declare that they have no conflict of interest or financial conflicts to disclose.

Appendix A. Supplementary data

Supplementary data to this article can be found online at <https://doi.org/10.1016/j.eng.2023.06.011>.

References

- [1] Strogatz SH. Exploring complex networks. *Nature* 2001;410(6825):268–76.
- [2] Schöll E. Chaos control sets the pace. *Nat Phys* 2010;6(3):161–2.
- [3] Albert R, Barabási AL. Statistical mechanics of complex networks. *Rev Mod Phys* 2002;74(1):47–97.
- [4] Newman MEJ. The structure and function of complex networks. *SIAM Rev* 2003;45(2):167–256.
- [5] Parrish JK, Edelstein-Keshet L. Complexity, pattern, and evolutionary trade-offs in animal aggregation. *Science* 1999;284(5411):99–101.
- [6] Vicsek T, Zafeiris A. Collective motion. *Phys Rep* 2012;517(3–4):71–140.
- [7] Ioannou CC, Guttal V, Couzin ID. Predatory fish select for coordinated collective motion in virtual prey. *Science* 2012;337(6099):1212–5.
- [8] Weimerskirch H, Martin J, Clerquin Y, Alexandre P, Jiraskova S. Energy saving in flight formation. *Nature* 2001;413(6857):697–8.
- [9] Sawicki J, Omelchenko I, Zakharova A, Schöll E. Chimera states in complex networks: interplay of fractal topology and delay. *Eur Phys J Spec Top* 2017;226:1883–92.
- [10] Nishikawa T, Motter AE. Synchronization is optimal in nondiagonalizable networks. *Phys Rev E* 2006;73(6):065106.
- [11] Nishikawa T, Motter AE. Maximum performance at minimum cost in network synchronization. *Physica D* 2006;224(1–2):77–89.
- [12] Sorrentino F. Effects of the network structural properties on its controllability. *Chaos* 2007;17(3):033101.
- [13] Li L, Jia QS, Wang H, Yuan R, Guan X. A systematic method for network topology reconfiguration with limited link additions. *J Netw Comput Appl* 2012;35(6):1979–89.
- [14] Pokhilko A, Fernández AP, Edwards KD, Southern MM, Halliday KJ, Millar AJ. The clock gene circuit in *Arabidopsis* includes a repressilator with additional feedback loops. *Mol Syst Biol* 2012;8(1):574.
- [15] Verma R. Analysis of 1996 western American electric blackouts. In: Proceedings of Bulk Power System Dynamics and Control-VI; 2004 Aug 22–27; Cortina d'Ampezzo, Italy; 2004.
- [16] Berizzi A. The Italian 2003 blackout. In: Proceedings of IEEE Power Engineering Society General Meeting; 2004 Jun 6–10; Denver, CO, USA; 2004.
- [17] Andersson G, Donalek P, Farmer R, Hatziaziyriou N, Kamwa I, Kundur P, et al. Causes of the 2003 major grid blackouts in North America and Europe, and recommended means to improve system dynamic performance. *IEEE Trans Power Syst* 2005;20(4):1922–8.

- [18] Liu YY, Slotine JJ, Barabási AL. Control centrality and hierarchical structure in complex networks. *PLoS ONE* 2012;7(9):e44459.
- [19] Iudice FL, Garofalo F, De Lellis P. Bounded partial pinning control of network dynamical systems. *IEEE Trans Control Netw Syst* 2023;10(1):238–48.
- [20] Das AK, Fierro R, Kumar V, Ostrowski JP, Spletzer J, Taylor CJ. A vision-based formation control framework. *IEEE Trans Robot Autom* 2002;18(5):813–25.
- [21] Tanner HG, Pappas GJ, Kumar V. Leader-to-formation stability. *IEEE Trans Robot Autom* 2004;20(3):443–55.
- [22] Nagy M, Ákos Z, Biro D, Vicsek T. Hierarchical group dynamics in pigeon flocks. *Nature* 2010;464(7290):890–3.
- [23] Ding W, Yan G, Lin Z. Collective motions and formations under pursuit strategies on directed acyclic graphs. *Automatica* 2010;46(1):174–81.
- [24] Torres P, van Wingerden JW, Verhaegen M. PO-MOESP subspace identification of directed acyclic graphs with unknown topology. *Automatica* 2015;53:60–71.
- [25] King AJ, Sueur C. Where next? Group coordination and collective decision making by primates. *Int J Primatol* 2011;32:1245–67.
- [26] Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, et al. Gene Ontology: tool for the unification of biology. *Nat Genet* 2000;25(1):25–9.
- [27] The Gene Ontology Consortium. Expansion of the Gene Ontology knowledgebase and resources. *Nucleic Acids Res* 2017;45(D1):D331–8.
- [28] Cantoni M, Weyer E, Li Y, Ooi SK, Mareels I, Ryan M. Control of large-scale irrigation networks. *Proc IEEE* 2007;95(1):75–91.
- [29] Gebraad PMO, van Dam FC, van Wingerden JW. A model-free distributed approach for wind plant control. In: *Proceedings of American Control Conference*; 2013 Jun 17–19; Washington, DC, USA; 2013.
- [30] Yang F, Kang P, Guan X. Distributed economic dispatch for cyber attacked smart grid based on resilient event-triggered consensus. In: *Proceedings of 59th IEEE Conference on Decision and Control (CDC)*; 2020 Dec 14–18; Jeju, Republic of Korea; 2020.
- [31] Smotherman M, Krishnamurthy S, Aravind PS, Hunnicutt D. Efficient DAG construction and heuristic calculation for instruction scheduling. In: *Proceedings of the 24th Annual International Symposium on Microarchitecture*; 1991 Nov 18–20; Albuquerque, NM, USA; 1991.
- [32] Horowitz R, Varaiya P. Control design of an automated highway system. *Proc IEEE* 2000;88(7):913–25.
- [33] Montague PR, Dayan P, Person C, Sejnowski TJ. Bee foraging in uncertain environments using predictive hebbian learning. *Nature* 1995;377(6551):725–8.
- [34] Huelsbergen L. Dynamic resolution: a runtime technique for the parallelization of modifications to directed acyclic graphs. *Int J Parallel Program* 1997;25:385–417.
- [35] Johnston WM, Hanna JRP, Millar RJ. Advances in dataflow programming languages. *ACM Comput Surv* 2004;36(1):1–34.
- [36] Bang-Jensen J, Gutin GZ. *Digraphs: theory, algorithms and applications*. London: Springer; 2009.
- [37] Friedman N. Inferring cellular networks using probabilistic graphical models. *Science* 2004;303(5659):799–805.
- [38] Husmeier D, Dybowski R, Roberts S. *Probabilistic modeling in bioinformatics and medical informatics*. London: Springer; 2005.
- [39] Moret BME, Nakhleh L, Warnow T, Linder CR, Tholse A, Paoletti A, et al. Phylogenetic networks: modeling, reconstructibility, and accuracy. *IEEE/ACM Trans Comput Biol Bioinformatics* 2004;1(1):13–23.
- [40] Quinn CJ, Coleman TP, Kiyavash N, Hatsopoulos NG. Estimating the directed information to infer causal relationships in ensemble neural spike train recordings. *J Comput Neurosci* 2011;30:17–44.
- [41] Eisen MB, Spellman PT, Brown PO, Botstein D. Cluster analysis and display of genome-wide expression patterns. *Proc Natl Acad Sci USA* 1998;95(25):14863–8.
- [42] Granger CW. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica* 1969;37(3):424–38.
- [43] Witthaut D, Timme M. Braess's paradox in oscillator networks, desynchronization and power outage. *New J Phys* 2012;14(8):083036.
- [44] Cohen JE, Horowitz P. Paradoxical behaviour of mechanical and electrical networks. *Nature* 1991;352(6337):699–701.
- [45] Ozogány K, Vicsek T. Modeling the emergence of modular leadership hierarchy during the collective motion of herds made of harems. *J Stat Phys* 2015;158(3):628–46.
- [46] Seres IA, Gulyás L, Nagy DA, Burcsi P. Topological analysis of Bitcoin's lightning network. In: *Pardalos P, Kotsireas I, Guo Y, Knottenbelt W, editors. Mathematical research for blockchain economy*. Cham: Springer; 2020.
- [47] Liu B, Chen Z, Zhang HT, Wang X, Geng T, Su H, et al. Collective dynamics and control for multiple unmanned surface vessels. *IEEE Trans Control Syst Technol* 2020;28(6):2540–7.
- [48] Demmel JW. *Applied numerical linear algebra*. Philadelphia: SIAM; 1997.
- [49] Gibbons JD, Chakraborti S. Nonparametric statistical inference. In: *Lovric M, editor. International encyclopedia of statistical science*. Berlin: Springer; 2014.
- [50] Calleman CJ. *The purposeful universe: how quantum theory and Mayan cosmology explain the origin and evolution of life*. New York City: Simon and Schuster; 2009.
- [51] Mo X, Chen Z, Zhang HT. Effects of adding a reverse edge across a stem in a directed acyclic graph. *Automatica* 2019;103:254–60.