Supplementary Material for Manuscript ID IEEE TCNS 23-0652

Structural Controllability of Multiplex Networks with the Minimum Number of Driver Nodes

Xiang Li, Guoqi Li, Leitao Gao, Lock Yue Chew and Gaoxi Xiao

January 17, 2024

The Supplementary Material is organized to align with the sequence of topics as they appear in the main paper. In Section I, we show that the proposed "Minimum-cost Flow based Driver-node Identification" (MFDI) method is extendable to two-layer networks with different node counts per layer and multiplex networks with more than two layers. In Section II, the MFDI method is compared with the existing methods for finding the minimum number of controllers required to ensure the structural controllability of two-layer multiplex networks. In Section III, we demonstrate that using the method designed for single-layer networks on two-layer networks can result in the allocation of driver nodes that violate the constraint. In Section IV, we include the actual controllability simulation results.

1 Two Extensions of MFDI Algorithm

In this section, we discuss two main extensions of the MFDI algorithm. Firstly, we adapted the MFDI algorithm for two-layer networks with different node counts across layers, focusing on finding the minimum number of driver nodes (restricted to the first layer) for structural controllability. Building on this, secondly, we extended the MFDI algorithm to multiplex networks with more than two layers, as layers beyond the first can be treated as a single 'super second layer'.

1.1 Extension of MFDI Algorithm for Two-layer Networks with Different Node Counts Across Layers

While Theorems 1 - 4 are initially formulated for multiplex networks with the same number of nodes in each layer and one-to-one directed interlayer connections, their main principles largely remain applicable when being extended to two-layer networks with varying numbers of nodes across layers, connected by directed edges from layer 1 to layer 2. However, for mathematical rigor, certain modifications in notation are required. For instance, if layer

1 contains n_1 nodes and layer 2 contains n_2 nodes, then the node set of the network becomes $V_A = V_1 \cup V_2$ with $V_1 = \{v_1, \cdots, v_{n_1}\}$ and $V_2 = \{v_{n_1+1}, \cdots, v_{n_1+n_2}\}$; the edge set is $E_A = E_1 \cup E_2 \cup E_I$ where $E_1 \subseteq V_1 \times V_1$, $E_2 \subseteq V_2 \times V_2$ and $E_I \subseteq V_1 \times V_2$. When a feasible solution exists for the extended problem, by modifying these symbols in Theorems 1 - 4 and slightly expanding the MFDI to an extended MFDI algorithm (as shown in Algorithm S1), one can identify the minimum number of driver nodes located only in the first layer to ensure the system's structural controllability. Our subsequent discussions may reference Theorems 1 - 4 and Lemma 4 from the main paper, so we have included them here for your convenience and easy reference.

Theorem 1: In $D(V_A, E_A)$, any set of vertices from which all network vertices are reachable contains at least one vertex of each source SCC.

Theorem 2: In $D(V', E', b, d)$, for any node v_i^{in} with an incoming edge $(q_s, v_i^{in}) \in E_s$ satisfying $f(q_s, v_i^{in}) = 1$, its corresponding vertex $v_i \in V_A$ is an initial node of a path in $\mathscr{P}$ in $D(V_A, E_A)$ where V_A can be covered by $\mathscr{P} \cup \mathscr{C}$.

Theorem 3: The cardinality of the minimum subtraction node set, $|(V_c \setminus V_r)^*|$, equals the total cost of the minimum-cost flow in $D(V'', E'', c, b, d)$.

Theorem 4: In a network $D(V_A, E_A)$ where $V_A = V_1 \cup V_2$ and $E_A = E_1 \cup E_2 \cup \{(v_i, v_{n+i})\}_{i=1}^{n}$, the cardinality of the minimum set of driver nodes for ensuring its structural controllability $|V_d^*|$ satisfies

$$|V_d^*| = |V_r| + |(V_c \setminus V_r)^*|,$$

where $|V_r| = k_0$ is the number of source SCCs in $D(V_1, E_1)$, and

$$|(V_c \setminus V_r)^*| = \sum_{(v_i, v_j) \in E''} f^*(v_i, v_j) \cdot c(v_i, v_j)$$

equals the total cost of the minimum-cost flow in $D(V'', E'', b, c, d)$.

The remaining gap to address is identifying the cases where no feasible solution exists, for which we introduce Lemma 4.

Lemma 4: In a two-layer network with different numbers of nodes per layer, a set of driver nodes on layer 1 that ensures the structural controllability of the system exists unless:

a) The two-layer network contains a source Strongly Connected Component (SCC) that does not include any nodes from layer 1; or

b) There is no feasible flow in the network $D(V'', E'', b, c, d)$ constructed in Step 1 of the MFDI algorithm.

Proof. ($\Longrightarrow$) The sufficiency is relatively straightforward. When neither of the aforementioned conditions is satisfied, the extended MFDI algorithm, as listed below, can be applied to find a feasible solution that ensures the structural controllability of the system, as per Theorems 1 - 4.

($\Longleftarrow$) The necessity proof has two parts. Condition a) follows from Lemma 3 and Theorem 1, which requires that driver nodes for structural controllability include at least one node from each source SCC. The presence of a driver node in layer 2 contradicts the requirement for all driver nodes to be on layer 1. For condition b), in the absence of a feasible flow in $D(V'', E'', b, c, d)$, it is impossible to find a set of vertex-disjoint paths and cycles, i.e., $\mathscr{P} \cup \mathscr{C}$, that covers all nodes in the multilayer network with each path's start node in layer 1, thus contradicting the necessity condition b) of Lemma 3 and rendering the problem infeasible.

■

In the extended MFDI algorithm, we have incorporated lines 4 - 5 and 17 - 18, to assess feasibility in line with the conditions a) and b) outlined in Lemma 4. Specifically, lines 4 - 5 are designed to check for the existence of any source SCC in layers other than the first layer. Lines 17-18 assess the existence of a feasible flow in $D(V'', E'', b, c, d)$. The remaining code segments are consistent with those for the original MFDI.

Algorithm S1: Extended Minimum-cost Flow based Driver-node Identification Algorithm (Part 1)

Input: Network $D(V_A, E_A)$ where $V_A = V_1 \cup V_2$, $E_A = E_1 \cup E_2 \cup E_I$.

Output: Minimum driver node set V_d^*.

Step 0: Feasibility Check:

if There is a source SCC found in layers other than the first **then**

  Return False

else

  Step 1: Associate graph construction:

  Find all k_0 source SCCs in $D(V_1, E_1)$ as $S_1, S_2, ..., S_{k_0}$, respectively.

  Create a fictitious vertex set $V_{\mathscr{R}} \leftarrow \{r_i\}_{i=1}^{k_0}$ consisting of k_0 vertices.

  Define a vertex set $V'' \leftarrow V^{in} \cup V^{out} \cup \{q_s, q_t\} \cup V_{\mathscr{R}}$, where $V^{in} \leftarrow \{v_i^{in} : v_i \in V_A\}$, and $V^{out} \leftarrow \{v_i^{out} : v_i \in V_A\}$.

  Define an edge set $E'' \leftarrow E'_A \cup E_s \cup E_t \cup \{(q_s, q_t)\} \cup \{(q_t, r_i)\}_{i=1}^{k_0} \cup \{(r_i, v_j^{in}) : v_j \in S_i\}_{i=1}^{k_0}$, where $E'_A \leftarrow \{(v_i^{out}, v_j^{in}) : (v_i, v_j) \in E_A\}$, $E_s \leftarrow \{(q_s, v^{in}) : v \in V_1\}$, $E_t \leftarrow \{(v^{out}, q_t) : v \in V_A\}$.

  Set $b(e) \leftarrow 1$, for every $e \in E'' \setminus \{(q_t, q_s)\}$, and $b(q_t, q_s) \leftarrow +\infty$.

  Set $c(e) \leftarrow 1$, for every $e \in E_s$; Set $c(q_t, r_i) \leftarrow -1$, for every $(q_t, r_i) \in \{(q_t, r_i)\}_{i=1}^{k_0}$, and $c(r_i, v_j^{in}) \leftarrow 1$, for every $(r_i, v_j^{in}) \in \{(r_i, v_j^{in}) : v_j \in S_i\}_{i=1}^{k_0}$; and set $c(e) \leftarrow 0$, for every $e \in E'_A \cup E_t \cup \{(q_s, q_t)\}$.

  Set $d(v_i^{in}) \leftarrow 1$, for every $v_i^{in} \in V^{in}$, and $d(v_j^{out}) \leftarrow -1$, for every $v_j^{out} \in V^{out}$; and set $d(v) \leftarrow 0$, for every $v \in V'' \setminus (V^{in} \cup V^{out})$.

  Obtain the associate graph $D(V'', E'', b, c, d)$.

Step 2: Driver nodes identification:

  Obtain the minimum-cost flow f^* in $D(V'', E'', b, c, d)$.

  if There is no feasible f^* **then**

    Return False

  else

Algorithm S1: Extended Minimum-cost Flow based Driver-node Identification Algorithm (Part 2)

Obtain edge set $E_{p1} \leftarrow \{(q_s, v_i^{in}) : f^*(q_s, v_i^{in}) > 0, v_i^{in} \in V^{in}\}$, and $E_{p2} \leftarrow \{(r_i, v_j^{in}) : f^*(r_i, v_j^{in}) > 0, v_j^{in} \in V^{in}, r_i \in V_{\mathscr{R}}\}$.

Initialize the driver node set $V_d^* \leftarrow \emptyset$.

for $(q_s, v_i^{in}) \in E_{p1}$ **do**

  $V_d^* \leftarrow V_d^* \cup \{v_i\}$

end for

for $(r_i, v_j^{in}) \in E_{p2}$ **do**

  $V_d^* \leftarrow V_d^* \cup \{v_j\}$

  $V_{\mathscr{R}} \leftarrow V_{\mathscr{R}} \setminus \{r_i\}$

end for

if $V_{\mathscr{R}} \neq \emptyset$ **then**

  for $r_i \in V_{\mathscr{R}}$ **do**

    $V_d^* \leftarrow V_d^* \cup \{\text{a node from } S_i\}$

  end for

end if

Return V_d^*

end if

end if

1.2 Extension of MFDI Algorithm for Multiplex Networks with More Than Two Layers

Building naturally on the extension of MFDI algorithm for multilayer networks with different numbers of nodes per layer, the second extension applies to multiplex networks with more than two layers. The main idea for extending the MFDI algorithm to k-layer

$(k = 2, 3, \cdots)$ multiplex networks is to treat all layers beyond the first as a unified 'super second layer', as visually depicted in Fig. S1. This is based on the criteria outlined in Lemmas 3 and 4 in the main manuscript, which stipulates that a system $(\bar{A}, \bar{B})$ is structurally controllable if all network nodes are accessible and can be covered by a set of node-disjoint stems and cycles. In the extended MFDI algorithm shown in Algorithm S1, the concepts of V_2 and E_2 can be adapted to represent the 'super second layer' in a k-layer multiplex network. Here, V_2 encompasses the collective set of nodes from the original network's second layer up to the k^{th} layer. Similarly, E_2 constitutes the combined set of both interlayer and intralayer edges spanning from the second to the k^{th} layer of the original network.

In the context of extending the MFDI algorithm to handle k-layer multiplex networks treated as 'two-layer networks with a super second layer', the graph construction process remains fundamentally the same as that in the original MFDI algorithm. Specifically, the model transformation adheres to the proofs and methodologies outlined in Theorems 1 - 4, where we can consider a network with n_1 nodes in the first layer and n_2 nodes in the 'super second layer'. Consequently, the node set of the multilayer network becomes $V_A = V_1 \cup V_2$, where $V_1 = v_1, \cdots, v_{n_1}$ and $V_2 = v_{n_1+1}, \cdots, v_{n_1+n_2}$. The edge set is defined as $E_A = E_1 \cup E_2 \cup E_I$, where $E_1 \subseteq V_1 \times V_1$, $E_2 \subseteq V_2 \times V_2$, and $E_I \subseteq V_1 \times V_2$. For consistency in notation and clarity in the extended MFDI algorithm, it is necessary to revise the description in Theorem 4 from "$E_A = E_1 \cup E_2 \cup \{(v_i, v_{n+i})\}_{i=1}^{n}$" to "$E_A = E_1 \cup E_2 \cup E_I$".

The consistency in the graph construction across the original and extended MFDI algorithms allows the minimum cost flow computation to be largely the same for the two cases. A crucial aspect to consider in the extended MFDI algorithm is the scenario where a feasible flow does not exist. To address this, an additional check is implemented in line 17 of the extended MFDI algorithm. This condition evaluates whether a feasible flow is present; if not, it concludes that the problem has no feasible solution. When a feasible flow does exist, the process of the minimum cost flow computation in the extended MFDI algorithm is identical to that in the original MFDI algorithm. This consistency is assured by the foundational proofs and methodologies delineated in Theorems 1 - 4.

The identification of driver nodes in the extended MFDI algorithm is meaningful only when the problem has a feasible solution. In cases where a solution exists, the process of identifying driver nodes is based on the minimum-cost flow computed on the constructed network. In the extended MFDI algorithm, the steps for driver node identification (lines 20 to 33 in Algorithm S1) are identical to those in the original MFDI algorithm (lines 10 to 23 in Algorithm 1 in the main paper). This similarity directly results from the scalability of the proofs and methodologies in Theorems 1 - 4 to multi-layer networks. For detailed descriptions of these steps, we have thoroughly explained them in the fourth paragraph on page 7 of the main paper.

The extended MFDI algorithm, when applied to k-layer multiplex networks, certainly results in an increased computational complexity compared to that of the original MFDI algorithm. As established in Theorem 5, the MFDI algorithm's time complexity is $\mathcal{O}\left(|V|^2|E|\log(|V|)\right)$, where $|V|$ represents the total number of nodes and $|E|$ signifies the total number of edges in the multiplex network. In the context of a k-layer multiplex network with each layer comprising n nodes, the total number of nodes equals $|V| = kn$. Considering the edge component of the time complexity calculation, and assuming an average degree μ in the network, the total number of edges in a k-layer multiplex network can be estimated as $|E| = k\mu n$. Under this assumption, the time complexity of the extended MFDI algorithm for a k-layer network is therefore $\mathcal{O}\left(k^3n^3\mu\log(kn)\right)$.

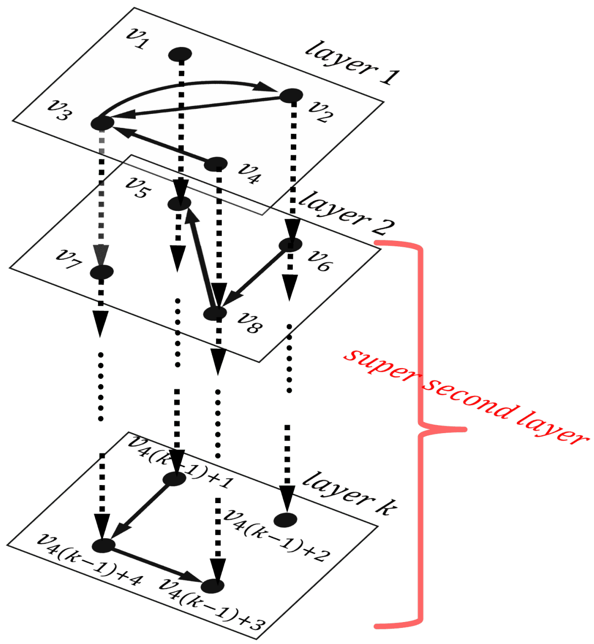

Fig. S1: An illustrative example treating all subsequent layers, beyond the first layer, as a single super second layer for simplification of the multiplex network analysis.

2 Comparative Analysis of MFDI with Existing Techniques

To the best of our knowledge, our work is the first to study the problem of finding the minimum number of driver nodes to ensure the structural controllability of multiplex networks. Our work shares a common interest in the structural controllability of multiplex networks with several other significant studies but features some distinct characteristics [14, 18, 19]. In [14], the authors examined finding the minimum number of controllers on multiplex multi-time-scale networks within discrete dynamics, providing a solution using the max flow method. However, this work may not be directly comparable to our work and existing works on continuous dynamics [18, 19] due to the significant difference in the dynamics framework. In the following, we shall conduct the comparisons between the

MFDI algorithm and the algorithms proposed in [18, 19] from problem formulation and numerical results perspectives.

In [18,19], the authors delved into the problem of finding the minimum number of controllers required to ensure target controllability on two-layer multiplex networks. These two studies respectively employed a maximum flow algorithm and a minimum-cost maximum flow algorithm, respectively. When their methods are applied to the problem of structural controllability, the target set is considered as the whole network's node sets. Specifically, their formulation of the problem is as follows. In a two-layer multiplex network with n nodes in each layer, of which the wiring diagram is $\bar{A} = \begin{bmatrix} \bar{A}_{11} & \mathbf{0} \\ \bar{A}_{21} & \bar{A}_{22} \end{bmatrix} \in \{0, \star\}^{2n \times 2n}$, the problem can be formulated as

$$\begin{aligned} \min_{\bar{B}_1 \in \{0,\star\}^{n\times m}} \; & m, \\ \text{s.t.} \quad & \bar{B} = \begin{bmatrix} \bar{B}_1 \\ \mathbf{0} \end{bmatrix} \in \{0, \star\}^{2n \times m}, \\ & \dot{\mathbf{x}}(t) = \bar{A}\mathbf{x}(t) + \bar{B}\mathbf{u}(t), \\ & (\bar{A}, \bar{B}) \text{ is structurally controllable.} \end{aligned} \tag{S1}$$

where m represents the number of external controllers.

In our paper, we study the problem of finding the minimum number of driver nodes necessary to ensure the structural controllability of multiplex networks. On a two-layer multiplex network, of which the wiring diagram is denoted as $\bar{A} = \begin{bmatrix} \bar{A}_{11} & \mathbf{0} \\ \bar{A}_{21} & \bar{A}_{22} \end{bmatrix} \in$

$\{0, \star\}^{2n \times 2n}$, we tackle the optimization problem

$$\begin{aligned}
&\min_{\bar{B}_1 \in \{0,\star\}^{n \times n}} \|\bar{B}\|_0, \\
&\text{s. t.} \quad \bar{B} = \begin{bmatrix} \bar{B}_1 \\ \mathbf{0} \end{bmatrix} \in \{0, \star\}^{2n \times n}, \\
&\qquad \dot{\mathbf{x}}(t) = \bar{A}\mathbf{x}(t) + \bar{B}\mathbf{u}(t), \\
&\qquad (\bar{A}, \bar{B}) \text{ is structurally controllable.}
\end{aligned} \tag{S2}$$

The main distinction between formulations (S1) and (S2) is that while (S1) focuses on finding the minimum number of controllers, (S2) is concerned with identifying the minimum number of driver nodes required to ensure the structural controllability of multiplex networks. Both equations (S1) and (S2) impose a constraint that controllers can only be connected to nodes in the first layer.

The proposed MFDI method is compared with the methods described in [18, 19] in terms of the driver nodes required to ensure the structural controllability of two-layer multiplex networks. The results of comparisons on synthetic multiplex networks are presented in Table S1, where each layer consists of n nodes and μ_i denotes the average nodal degree in the i^{th} layer. In Table S1, each value is represented as the mean $\pm$ standard deviation calculated from 100 independent simulations. Moreover, Table S2 showcases the comparisons made on real-life multiplex networks where each layer consists of n nodes and $|E_i|$ denotes the number of edges in the i^{th} layer. The real-life network datasets are accessible on the website of CoMuNe Laboratory[1], the Research Unit of Bruno Kessler Foundations Center for Information and Communication Technologies. For those networks with more than two layers, we only use the data of the first two layers in the simulations. In Table S2, each real-life multiplex network is represented where n marks the number of nodes in each layer, and $|E_i|$ denotes the number of edges in the i^{th} layer.

[1] https://manliodedomenico.com/data.php

Table S1: Comparisons between the number of driver nodes obtained by different methods on two-layer synthetic multiplex networks ($n = 1000$)

	maximum flow algorithm [18]	minimum-cost maximum flow algorithm [19]	proposed MFDI algorithm
ER-ER, $\mu_1 = 2$, $\mu_2 = 2$	330.99 ± 10.48	330.99 ± 10.48	**330.62 ± 10.11**
ER-ER, $\mu_1 = 3$, $\mu_2 = 3$	131.44 ± 9.10	131.44 ± 9.10	**127.95 ± 8.87**
ER-ER, $\mu_1 = 4$, $\mu_2 = 4$	44.04 ± 5.52	44.04 ± 5.52	**41.88 ± 5.40**
ER-ER, $\mu_1 = 5$, $\mu_2 = 5$	18.10 ± 4.33	18.10 ± 4.33	**14.28 ± 3.55**
ER-ER, $\mu_1 = 8$, $\mu_2 = 8$	6.64 ± 2.17	6.64 ± 2.17	**1.36 ± 0.66**
SF-SF, $\mu_1 = 2$, $\mu_2 = 2$	776.58 ± 7.89	776.58 ± 7.89	**775.16 ± 7.37**
SF-SF, $\mu_1 = 3$, $\mu_2 = 3$	693.82 ± 8.92	693.82 ± 8.92	**692.28 ± 8.21**
SF-SF, $\mu_1 = 10$, $\mu_2 = 10$	301.46 ± 9.85	301.46 ± 9.85	**300.66 ± 9.88**
SF-SF, $\mu_1 = 15$, $\mu_2 = 15$	129.81 ± 8.07	129.81 ± 8.07	**128.64 ± 8.16**
SF-SF, $\mu_1 = 20$, $\mu_2 = 20$	26.41 ± 7.07	26.41 ± 7.07	**23.22 ± 6.93**
SF-SF, $\mu_1 = 30$, $\mu_2 = 30$	6.28 ± 1.71	6.28 ± 1.71	**1.00 ± 0.00**

Table S2: Comparisons between the Number of Driver Nodes Obtained by Different Methods on Two-layer Real-life Multiplex Networks

	maximum flow algorithm [18]	minimum-cost maximum flow algorithm [19]	proposed MFDI algorithm
CKM physicians' innovation [43] $n = 246$, $\lvert E\rvert_1 = 480$, $\lvert E\rvert_2 = 565$	133	133	**127**
Lazega law firm [44] $n = 71$, $\lvert E\rvert_1 = 892$, $\lvert E\rvert_2 = 575$	9	9	**6**
C. Elegans connectome [47] $n = 279$, $\lvert E\rvert_1 = 1031$, $\lvert E\rvert_2 = 1639$	135	135	**60**
C. Elegans GPI [48] $n = 3879$, $\lvert E\rvert_1 = 5557$, $\lvert E\rvert_2 = 313$	3817	3817	**3810**
Arabidopsis GPI [48] $n = 6980$, $\lvert E\rvert_1 = 13857$, $\lvert E\rvert_2 = 4411$	6597	6597	**6564**

From Tables S1 and S2, it can be observed that the proposed MFDI consistently finds the smallest number of driver nodes on either synthetic or real-life networks. This is the comparison between the number of driver nodes when minimizing the number of controllers versus the actual minimum number of driver nodes. However, it's important to note that we should not claim our algorithm "outperforms" the algorithms in [18, 19] because their optimization targets are different from ours. Table S2 also outlines the number of driver nodes needed when minimizing the number of controllers (using the maximum flow algorithm [18] and the minimum-cost maximum flow algorithm [19]) versus the actual minimum number of driver nodes (using the MFDI algorithm) in four real-life networks. The comparisons demonstrate the different results under different control objectives. It may be worth mentioning that the differences tend to be smaller in highly sparse networks, as can be observed in the last two rows of Table R1. Further future studies are needed to verify whether such is always the case in networks with different clustering and community structures.

3 Applying Single-Layer Methods to Two-Layer Networks

The fundamental distinction between identifying the minimum number of driver nodes required to ensure structural controllability in single-layer and two-layer multiplex networks is that, in the latter, the driver nodes must be situated in the first layer. Casting a two-layer network into a single-layer network and using the method in [22] on two-layer networks may result in the allocation of driver nodes that violates this constraint. We further elucidate this with mathematical formulations and supportive empirical data.

Specifically, in [22], the authors studied the problem of finding the minimum number of driver nodes on single-layer networks, which could be formulated as

$$\begin{aligned} &\min_{\bar{B}\in\{0,\star\}^{n\times n}} \|\bar{B}\|_0, \\ &\text{s.t.} \quad \dot{\mathbf{x}}(t) = \bar{A}\mathbf{x}(t) + \bar{B}\mathbf{u}(t), \\ &\qquad\quad (\bar{A},\bar{B}) \text{ is structurally controllable,} \end{aligned} \tag{S3}$$

where $\bar{A} \in \{0,\star\}^{n\times n}$ denotes the wiring diagram of the network, and $\|\bar{B}\|_0$ denotes the number of non-zero entries in B, i.e., the number of driver nodes. This problem was solved by a minimum-weighted maximum matching method.

In our paper, we study the problem of finding the minimum number of driver nodes needed to ensure the structural controllability of multiplex networks, where the driver nodes can only be situated on the first layer of the network. On a two-layer multiplex network, of which the wiring diagram is denoted as $\bar{A} = \begin{bmatrix} \bar{A}_{11} & \mathbf{0} \\ \bar{A}_{21} & \bar{A}_{22} \end{bmatrix} \in \{0,\star\}^{2n\times 2n}$, the problem is formulated as

$$
\begin{aligned}
&\min_{\bar{B}_1 \in \{0,\star\}^{n\times n}} \|\bar{B}\|_0, \\
&\text{s.t.} \quad \bar{B} = \begin{bmatrix} \bar{B}_1 \\ \mathbf{0} \end{bmatrix} \in \{0,\star\}^{2n\times n}, \\
&\qquad \dot{\mathbf{x}}(t) = \bar{A}\mathbf{x}(t) + \bar{B}\mathbf{u}(t), \\
&\qquad (\bar{A}, \bar{B}) \text{ is structurally controllable.}
\end{aligned} \tag{S4}
$$

We shall then illustrate that the minimum-weighted maximum matching method in [22] may violate the constraints in our problem (S4). In order to keep the dimensions consistent, we rewrite the formulation (S3) as

$$
\begin{aligned}
&\min_{\bar{B} \in \{0,\star\}^{2n\times n}} \|\bar{B}\|_0, \\
&\text{s.t.} \quad \dot{\mathbf{x}}(t) = \bar{A}\mathbf{x}(t) + \bar{B}\mathbf{u}(t), \\
&\qquad (\bar{A}, \bar{B}) \text{ is structurally controllable,}
\end{aligned} \tag{S5}
$$

where $\bar{A} \in \{0,\star\}^{2n\times 2n}$. The distinction between (S4) and (S5) is that (S5) does not confine the locations of driver nodes. As such, the lower half of $\bar{B} \in \{0,\star\}^{2n\times n}$ obtained by using minimum-weighted maximum matching method [22] may contain non-zero entries, violating the restriction set in (S4). Table S3 records the proportion of instances where the method proposed in [22] fails to meet the constraints of our problem (S4) on synthetic networks. This implies the scenarios where the lower half of the obtained $\bar{B} \in \{0,\star\}^{2n\times n}$ has non-zero entries. In Table S3, μ_i represents the mean degree of the i^{th} layer. For each network configuration, the result is generated from simulations run on 100 independent networks. Table S4 illustrates cases from real-life multiplex networks, assessing whether the calculated $\bar{B} \in \{0,\star\}^{2n\times n}$ obtained using the minimum-weighted maximum matching method [22], violates the constraints outlined in (S4). The real-life network datasets are accessible on the website of CoMuNe Laboratory[2], the Research Unit of Bruno Kessler

[2]https://manliodedomenico.com/data.php

Foundations Center for Information and Communication Technologies. For those networks with more than two layers, we only use the data of the first two layers in the simulations. In Table S4, for each real-life multiplex network, n denotes the number of nodes in each layer and $|E_i|$ denotes the number of edges in the i^{th} layer. The symbol ✓ denotes that $\bar{B}$, calculated using the minimum-weighted maximum matching method, complies with the constraints specified in (S4). Conversely, ✗ indicates that the calculated $\bar{B}$ violates the constraints.

Table S3: The proportion of instances where the calculated B using the minimum-weighted maximum matching method [22] on synthetic networks fails to comply with the constraints specified in problem (S4)

two-layer ER-ER multiplex networks, n = 1000	$\mu_1 = 2$	$\mu_1 = 3$	$\mu_1 = 4$	$\mu_1 = 5$
$\mu_2 = 2$	100%	100%	100%	100%
$\mu_2 = 3$	100%	100%	100%	100%
$\mu_2 = 4$	84%	83%	94%	100%
$\mu_2 = 5$	46%	42%	43%	67%
two-layer SF-SF multiplex networks, n = 1000	$\mu_1 = 2$	$\mu_1 = 3$	$\mu_1 = 4$	$\mu_1 = 5$
$\mu_2 = 2$	100%	100%	100%	100%
$\mu_2 = 3$	100%	100%	100%	100%
$\mu_2 = 4$	92%	100%	100%	100%
$\mu_2 = 5$	86%	95%	100%	100%

Table S4: Assessment of constraint violations in (S4) by the minimum-weighted maximum matching method [22] on real-life multiplex networks

CKM physicians' innovation [43], $n = 246$, $\lvert E\rvert_1 = 480$, $\lvert E\rvert_2 = 565$	✗
Lazega law firm [44], $n = 71$, $\lvert E\rvert_1 = 892$, $\lvert E\rvert_2 = 575$	✓
C. Elegans connectome [47], $n = 279$, $\lvert E\rvert_1 = 1031$, $\lvert E\rvert_2 = 1639$	✗
C. Elegans GPI [48], $n = 3879$, $\lvert E\rvert_1 = 5557$, $\lvert E\rvert_2 = 313$	✗
Arabidopsis GPI [48], $n = 6980$, $\lvert E\rvert_1 = 13857$, $\lvert E\rvert_2 = 4411$	✗

From Tables S3 and S4, it can be seen that on either synthetic or real-life networks, results generated by the minimum-weighted maximum matching method [22] may not fulfill the constraints in (S4).

4 Actual Controllability Simulation Results

In this section, we include the actual controllability simulation results. This provides interested readers with comprehensive access to in-depth results. We may discuss the actual controllability by making use of the example shown in Fig. 1(c). To delve into actual controllability, we begin by replacing the structural patterns of the system matrix $\bar{A} \in \{0, \star\}^{n\times n}$ and input matrix $\bar{B} \in \{0, \star\}^{n\times m}$ with the corresponding weights. For simplicity and without loss of generality, we replace the parameter $\star$ with 1. In this case, we have

$$
A=\begin{bmatrix}
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0\\
0&1&0&0&0&0&0&0&1&0&1&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&1&0&0&0&0&0&0\\
0&0&0&1&0&0&0&0&0&0&0&0&0&0&1&0\\
0&0&0&0&1&0&0&0&0&0&0&0&0&1&0&0\\
0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&1\\
0&0&0&0&0&0&1&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0
\end{bmatrix}
\text{ and } B=\begin{bmatrix}
1&0&0&0&0\\
0&1&0&0&0\\
0&0&0&0&0\\
0&0&0&0&0\\
0&0&1&0&0\\
0&0&0&1&0\\
0&0&0&0&0\\
0&0&0&0&1\\
0&0&0&0&0\\
0&0&0&0&0\\
0&0&0&0&0\\
0&0&0&0&0\\
0&0&0&0&0\\
0&0&0&0&0\\
0&0&0&0&0\\
0&0&0&0&0
\end{bmatrix}.
$$

In control theory, a networked system is controllable if it can be driven to any desired state with appropriate inputs. Without loss of generality, we set the system's desired state as $\mathbf{0}$, while the initial state is denoted as $\mathbf{x}_0 \in \mathbb{R}^{n\times 1}$. We denote the required control time as t_f. Given a controllable system (A, B), when the input signal

$$
\mathbf{u}(t) = -B^T e^{A^T\left(t_f - t\right)} \left[W_B\right]^{-1} e^{At_f}\mathbf{x}_0 \tag{S6}
$$

is applied, where $W_B = \int_0^{t_f} e^{A\left(t_f-t\right)} BB^T e^{A^T\left(t_f-t\right)} dt$, the states could reach the origin at time $t = t_f$, i.e., $\mathbf{x}(t_f) = e^{At_f}\mathbf{x}_0 - W_B\left(W_B\right)^{-1} e^{At_f}\mathbf{x}_0 = \mathbf{0}$ [18–21, 31]. We generate the initial state $\mathbf{x}_0$ from the normal distribution with a mean of 0 and a variance of 1 and we set $t_f = 2$. Upon applying the input signal as defined by (S6), the states of all network nodes are simulated and the results are presented in Fig. S2. All node states converge to zero at $t = t_f$.

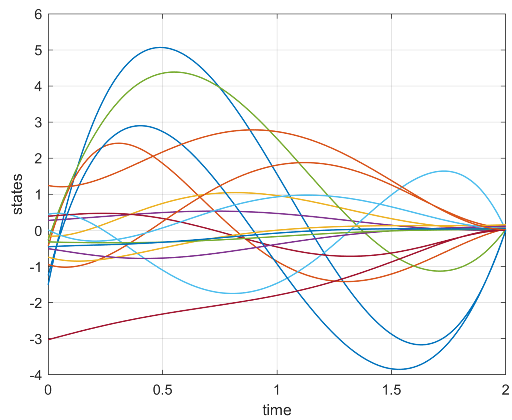

Fig. S2: Actual controllability of the networked system (A, B).

References (in sequence as they appear in the Supplementary Material)

[14] M. Pósfai, J. Gao, S. P. Cornelius, A.-L. Barabási, and R. M. D'Souza, "Controllability of multiplex, multi-time-scale networks," *Phys. Rev. E*, vol. 94, p. 032316, Sep 2016.

[18] K. Song *et al.*, "Target controllability of two-layer multiplex networks based on network flow theory," *IEEE Trans. Cybern.*, vol. 51, no. 5, pp. 2699–2711, 2021.

[19] J. Ding *et al.*, "Target controllability in multilayer networks via minimum-cost maximum-flow method," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 5, pp. 1949–1962, 2021.

[43] J. Coleman, E. Katz, and H. Menzel, "The diffusion of an innovation among physicians," *Sociometry*, vol. 20, no. 4, pp. 253–270, 1957.

[44] T. A. Snijders, P. E. Pattison, G. L. Robins, and M. S. Handcock, "New specifications for exponential random graph models," *Sociol. Methodol.*, vol. 36, no. 1, pp. 99–153, 2006.

[47] B. L. Chen, D. H. Hall, and D. B. Chklovskii, "Wiring optimization can relate neuronal structure and function," *Proc. Natl. Acad. Sci.*, vol. 103, no. 12, pp. 4723–4728, 2006.

[48] C. Stark, B. J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, and M. Tyers, "Biogrid: a general repository for interaction datasets," *Nucleic Acids Res.*, vol. 34, no. suppl_1, pp. D535–D539, 2006.

[22] S. Pequito, S. Kar, and A. P. Aguiar, “A framework for structural input/output and control configuration selection in large-scale systems,” *IEEE Trans. Autom. Control*, vol. 61, no. 2, pp. 303–318, 2016.

[20] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, “Controllability of complex networks,” *Nature*, vol. 473, no. 7346, pp. 167–173, 2011.

[31] G. Li *et al.*, “Enabling controlling complex networks with local topological information,” *Sci. Rep.*, vol. 8, no. 1, pp. 1–10, 2018.

[21] G. Li, X. Chen, P. Tang, G. Xiao, C. Wen, and L. Shi, “Target control of directed networks based on network flow problems,” *IEEE Trans. Control. Netw. Syst.*, vol. 7, no. 2, pp. 673–685, 2020.

[42] M. Pósfai, Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, “Effect of correlations on network controllability,” *Sci. Rep.*, vol. 3, no. 1, pp. 1–7, 2013.

[13] A. Chapman, M. Nabi-Abdolyousefi, and M. Mesbahi, “Controllability and observability of network-of-networks via cartesian products,” *IEEE Trans. Autom. Control*, vol. 59, no. 10, pp. 2668–2679, 2014.

[15] G. Menichetti, L. DallAsta, and G. Bianconi, “Control of multilayer networks,” *Sci. Rep.*, vol. 6, no. 1, pp. 1–8, 2016.

[16] D. Wang and X. Zou, “Control energy and controllability of multilayer networks,” *Adv Complex Syst*, vol. 20, no. 04n05, p. 1750008, 2017.

[17] Z. Yuan, C. Zhao, W.-X. Wang, Z. Di, and Y.-C. Lai, “Exact controllability of multiplex networks,” *New J. Phys.*, vol. 16, no. 10, p. 103036, 2014.