Controllability of multiplex, multi-time-scale networks

Márton Pósfai[*]

Complexity Science Center, University of California, Davis, California 95616, USA;
Department of Computer Science, University of California, Davis, California 95616, USA;
and Department of Physics of Complex Systems, Eötvös University, Budapest H-1117, Hungary

Jianxi Gao and Sean P. Cornelius

Center for Complex Network Research, Department of Physics, Northeastern University, Boston, Massachusetts 02115, USA

Albert-László Barabási

Center for Complex Network Research, Department of Physics, Northeastern University, Boston, Massachusetts 02115, USA;
Center for Cancer Systems Biology, Dana-Farber Cancer Institute, Harvard University, Boston, Massachusetts 02215, USA;
Department of Medicine, Brigham and Women's Hospital, Harvard Medical School, Boston, Massachusetts 02115, USA;
and Center for Network Science, Central European University, Budapest H-1051, Hungary

Raissa M. D'Souza

Complexity Science Center, University of California, Davis, California 95616, USA;
Department of Computer Science, University of California, Davis, California 95616, USA;
Department of Mechanical and Aerospace Engineering, University of California, Davis, California 95616, USA;
and Santa Fe Institute, 1399 Hyde Park Road, Santa Fe, New Mexico 87501, USA

The paradigm of layered networks is used to describe many real-world systems, from biological networks to social organizations and transportation systems. While recently there has been much progress in understanding the general properties of multilayer networks, our understanding of how to control such systems remains limited. One fundamental aspect that makes this endeavor challenging is that each layer can operate at a different time scale; thus, we cannot directly apply standard ideas from structural control theory of individual networks. Here we address the problem of controlling multilayer and multi-time-scale networks focusing on two-layer multiplex networks with one-to-one interlayer coupling. We investigate the practically relevant case when the control signal is applied to the nodes of one layer. We develop a theory based on disjoint path covers to determine the minimum number of inputs (N_i) necessary for full control. We show that if both layers operate on the same time scale, then the network structure of both layers equally affect controllability. In the presence of time-scale separation, controllability is enhanced if the controller interacts with the faster layer: N_i decreases as the time-scale difference increases up to a critical time-scale difference, above which N_i remains constant and is completely determined by the faster layer. We show that the critical time-scale difference is large if layer I is easy and layer II is hard to control in isolation. In contrast, control becomes increasingly difficult if the controller interacts with the layer operating on the slower time scale and increasing time-scale separation leads to increased N_i, again up to a critical value, above which N_i still depends on the structure of both layers. This critical value is largely determined by the longest path in the faster layer that does not involve cycles. By identifying the underlying mechanisms that connect time-scale difference and controllability for a simplified model, we provide crucial insight into disentangling how our ability to control real interacting complex systems is affected by a variety of sources of complexity.

I. INTRODUCTION

Over the past two decades, the theory of networks proved to be a powerful tool for understanding individual complex systems [1,2]. However, it is now increasingly appreciated that complex systems do not exist in isolation, but interact with each other [3,4]. Indeed, an array of phenomena—from cascading failures [5,6] to diffusion [7]—can be fully understood only if these interactions are taken into account. Traditional network theory is not sufficient to describe the structure of such systems, so in response to this challenge, the paradigm of multilayer networks is being actively developed. Here we study

a fundamental, yet overlooked, aspect of multilayer networks: Each individual layer can operate at a different time scale. Particularly, we address the problem of controlling multilayer, multi-time-scale systems focusing on two-layer multiplex networks. Recently, significant efforts have been made to uncover how the underlying network structure of a system affects our ability to influence its behavior [8–16]. However, despite the appearance of coupled systems from infrastructure to biology, the existing literature—with a few notable exceptions [17–20]—has focused on control of networks in isolation, and the role of time scales remains unexplored.

Control of multilayer networks is important for many applications. For example, consider a CEO aiming to lead a company consisting of employees and management. Studying the network of managers or the network of employees in

————
[*]posfai@ucdavis.edu

isolation does not take into account important interactions between the different levels of hierarchy of the company. On the other hand, treating the system as one large network ignores important differences between the dynamics of the different levels, e.g., management may meet weekly, while employees are in daily interaction. In general, the interaction of time scales plays an important role in organization theory [21]. Alternatively, consider gene regulation in a living cell. External stimuli activate signaling pathways which, through a web of protein-protein interactions, affect transcription factors responsible for gene expression. The activation of a signaling pathway happens on the time scale of seconds, while gene expression typically takes hours [22]. As a third example, consider an operator of an online social network who wants to enhance the spread of certain information by interacting with its users. However, a user may subscribe to multiple social networking services and may opt to share news encountered in one network through a different one, out of reach of the operator. The dynamics of user interaction on different web sites can be very different depending on user habits and the services offered [23–25]. For example the URL shortening service Bitly reports that the half-life of shared links depends on the social networking platform used: Half the clicks on a link happened within 2.8 h after posting on Twitter, within 3.2 h on Facebook, and within 7.4 h on YouTube [26].

Common features of these examples are that (i) each interacting subsystem is described by a separate complex network; (ii) the dynamics of each subsystem operate on a different, but often comparable time scale; and (iii) the external controller directly interacts with only one of the subsystems. Here we study the control properties of a model that incorporates these common features, yet remains tractable. More specifically, we study discrete-time linear dynamics on two-layer multiplex networks, meaning that we assume one-to-one coupling between the nodes of the two layers. This choice ensures both analytical tractability and the isolation of the role of time scales from the effect of more complex multilayer network structure. Identifying the underlying mechanisms that govern the controllability of this simple model provides crucial insight into disentangling how our ability to control real interacting complex systems is affected by a variety of sources of complexity.

So far only limited work investigated controllability of multilayer networks. Menichetti *et al.* investigated the controllability of two-layer multiplex networks governed by linear dynamics such that the dynamics of the two layers are not coupled, but the input signals in the two layers are applied to the same set of nodes [18]. Yuan *et al.* identified the minimum number of inputs necessary for full control of diffusion dynamics, allowing the controller to interact with any layer [19]. Zhang *et al.* investigated the controllable subspace of multilayer networks with linear dynamics without time-scale separation if the controller is limited to interact with only one layer; showing that it is more efficient to directly control peripheral nodes than central ones [20]. Here we also limit the controller to one layer, yet by exploring the minimum input problem, we offer a direct metric which allows us to compare our findings to previous results for single-layer networks [10]. More so, the key innovation of our work is that

we take into account the time scale of the dynamics of each layer, a mostly overlooked aspect of multilayer networks.

It is worth mentioning the recent work investigating the related, but distinct problem of controllability of networks with time-delayed linear dynamics [27]. The key difference between time-delay and time scale difference is that for time-delayed dynamics the state of a node will depend on some previous state of its neighbors; however, the typical time to change the state of a node remains the same throughout the system. While in case of time-scale difference, the typical time needed for changes to happen can be different in different parts of the system.

In the next section, we introduce a simple model that captures some common properties of multilayer networks and we describe the problem setup. In Sec. III, we develop a theory to determine the minimum number of inputs required for controlling multiplex, multi-time-scale networks with discrete-time linear dynamics relying on graph combinatorial methods. In Sec. IV, we use networks with tunable degree distribution to systematically uncover the role of network structure and time-scale separation. We study three scenarios: no time-scale separation, layer I operates faster, and layer II operates faster. Finally, in Sec. V we provide a discussion of our results and we outline open questions.

II. MODEL DEFINITION

We aim to study the controllability of coupled complex dynamical systems with the following properties: (i) each subsystem (layer) is described by a complex network; (ii) the operation of each layer is characterized by a different time scale; and (iii) the controller only interacts directly with one of the layers. We propose a model that satisfies these requirements and yet is simple enough to remain tractable. We focus on two-layer multiplex systems, meaning that there is a one-to-one correspondence between the nodes of the two layers.

The model is defined by a weighted directed two-layer multiplex network $\mathcal{M}$ which consists of two networks $\mathcal{L}_\text{I}$ and $\mathcal{L}_\text{II}$ called layers and a set of links $E_\text{I,II}$ connecting the nodes of the different layers. Each layer $\mathcal{L}_\alpha$ (where $\alpha \in \{\text{I,II}\}$) consists of a set of nodes $V_\alpha = \{v_1^\alpha, v_2^\alpha, \ldots, v_N^\alpha\}$ and a set of links E_α, where a directed link $(v_i^\alpha, v_j^\alpha, w_{ij}^\alpha) \in E_\alpha$ is an ordered node pair and a weight representing that node v_i^α influences node v_j^α with strength w_{ij}^α. The two layers are connected by link set $E_\text{I,II} = \{(v_i^\text{I}, v_i^\text{II}, w_i^\text{I,II}) | i = 1,2\ldots,N\}$; in other words, there is directed one-to-one coupling from layer I to layer II [Fig. 1(a)]. Although the links are weighted, the exact values of the weights do not have to be known for our purposes.

Our goal is to control the system by only interacting directly with layer I. We study linear discrete-time dynamics,

$$\mathbf{x}_\text{I}(t) = \mathbf{A}_\text{I}\mathbf{x}_\text{I}(t - \tau_\text{I}) + \mathbf{B}\mathbf{u}(t - \tau_\text{I}) \quad \text{if } (t \bmod \tau_\text{I}) = 0,$$

$$\mathbf{x}_\text{II}(t) = \mathbf{A}_\text{II}\mathbf{x}_\text{II}(t-\tau_\text{II}) + \Delta_{\tau_\text{I}}(t)\mathbf{D}\mathbf{x}_\text{I}(t-\tau_\text{I}) \quad \text{if } (t \bmod \tau_\text{II}) = 0,$$

$$(1)$$

where $\mathbf{x}_\text{I}(t)$ and $\mathbf{x}_\text{II}(t) \in \mathbb{R}^N$ represent the state of nodes in layers I and II; the matrices $\mathbf{A}_\text{I}$ and $\mathbf{A}_\text{II} \in \mathbb{R}^{N \times N}$ are the transposed weighted adjacency matrices of layers I and II, capturing their internal dynamics. The weighted diagonal matrix $\mathbf{D} \in \mathbb{R}^{N \times N}$ captures how layer I affects layer II.
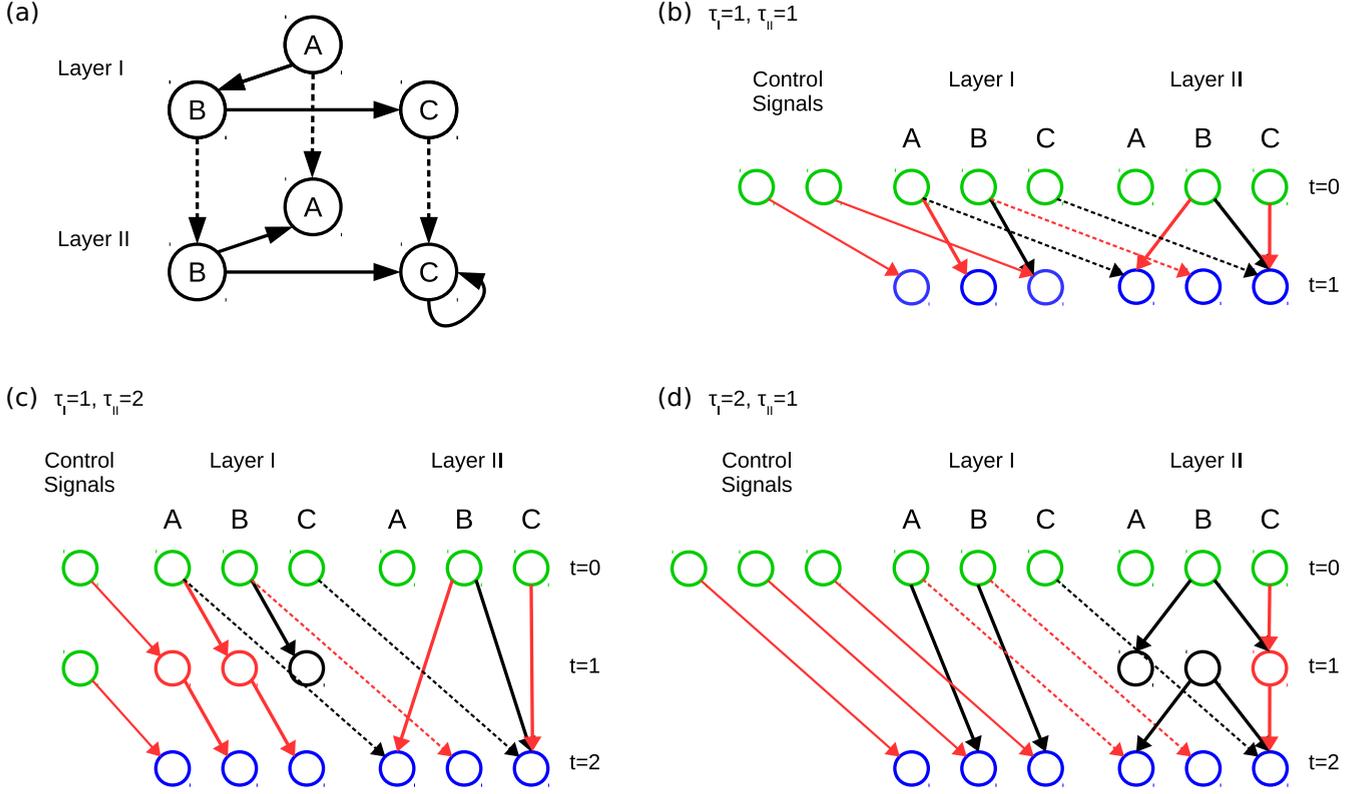
FIG. 1. Structural controllability of two-layer multiplex networks. (a) A two-layer network. (b)–(d) To determine N_i, we construct the dynamic graph representing the time evolution of the system from $t_0 = 0$ to $t_1 = \max(\tau_I, \tau_{II})$. The system is controllable only if all nodes at t_1 (blue) are connected to nodes at t_0 or nodes representing control signals (green) via disjoint paths (red). (b) In case of no time-scale separation ($\tau_I = \tau_{II} = 1$), each disjoint control path consists of a single link, yielding $N_i = 2$. (c) If layer I updates twice as frequently as layer II ($\tau_I = 1$, $\tau_{II} = 2$), we are allowed to inject control signals at time steps $t = 0$ and 1, reducing the number of inputs to $N_i = 1$. (d) On the other hand, if layer II is faster ($\tau_I = 2$, $\tau_{II} = 1$), layer II needs to support longer control paths, yielding $N_i = 3$.

Vector $\mathbf{u}(t) \in \mathbb{R}^M$ provides the set of independent inputs and the matrix $\mathbf{B} \in \mathbb{R}^{N \times M}$ defines how the inputs are coupled to the system. To differentiate between the function $\mathbf{u}(t)$ and an instance of the function at a given time step, we refer to a component $u_i(t)$ of vector $\mathbf{u}(t)$ as an independent input, and we call its value at time step t', $u_i(t = t')$, a signal.

Finally, $\tau_I, \tau_{II} \in \{1, 2, \ldots\}$ are the time-scale parameters of each subsystem, meaning that the state of layer I is updated according to Eq. (1) every τ_Ith time step; and layer II is updated every τ_{II}th time step. Also,

$$\Delta_{\tau_I}(k) = \begin{cases} 1 & \text{if } (k \bmod \tau_I) = 0, \\ 0 & \text{if } (k \bmod \tau_I) \neq 0, \end{cases} \qquad (2)$$

is the Kronecker comb, meaning that layer I directly impacts the dynamics of layer II if the two layers simultaneously update. We investigate three scenarios: (i) the subsystems operate on the same time scale, i.e., $\tau_I = \tau_{II} = 1$; (ii) layer I updates faster, i.e., $\tau_I = 1$ and $\tau_{II} > 1$; and (iii) layer II updates faster $\tau_I > 1$ and $\tau_{II} = 1$.

We seek full control of the system as defined by Kalman [28], meaning that with the proper choice of $\mathbf{u}(t)$, we can steer the system from any initial state to any final state in finite time. To characterize controllability, we aim to design a matrix $\mathbf{B}$ such that the system is controllable and the number of independent control inputs, M, is minimized. The minimum number of inputs, N_i, serves as our measure of how difficult it is to control the system.

To find a robust and efficient algorithm to determine N_i, we rely on the framework of structural controllability [29]. We say that a matrix $\mathbf{A}^*$ has the same structure as $\mathbf{A}$, if the zero-nonzero elements of $\mathbf{A}$ and $\mathbf{A}^*$ are in the same position, and only the value of the nonzero entries can be different; in other words, in the corresponding network the links connect the same nodes and only the link weights can differ. A linear system of Eq. (1) defined by matrices $(\mathbf{A}_I, \mathbf{A}_{II}, \mathbf{D}, \mathbf{B})$ is structurally controllable if there exists matrices with the same structure $(\mathbf{A}_I^*, \mathbf{A}_{II}^*, \mathbf{D}^*, \mathbf{B}^*)$ such that the dynamics defined by $(\mathbf{A}_I^*, \mathbf{A}_{II}^*, \mathbf{D}^*, \mathbf{B}^*)$ are controllable according to the definition of Kalman. Note that ultimately we are interested in controllability and not structural controllability. Yet, structural controllability is a useful tool because (i) if a linear system is structurally controllable, it is controllable for almost all link weight combinations [29] and (ii) determining structural controllability can be mapped to a graph combinatorial problem allowing for efficient and numerically robust algorithms.

III. MINIMUM INPUT PROBLEM

Before addressing the minimum input problem of multiplex networks, we revisit the case of single-layer networks by providing an alternative explanation of the minimum input

theorem of Liu *et al.* [10]. This approach readily lends itself to being extended to multiplex, multi-time-scale networks, thus providing the basis for Sec. III B, in which we develop an algorithm to determine N_i for two-layer multiplex networks.

A. Single-layer networks

The linear discrete-time dynamics associated to a single-layer weighted directed network $\mathcal{L}$ are formulated as

$$\mathbf{x}(t) = \mathbf{A}\mathbf{x}(t-1) + \mathbf{B}\mathbf{u}(t-1), \qquad (3)$$

where $\mathbf{x}(t)$, $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{u}(t)$ are defined similarly as in Eq. (1) [Fig. 2(a)]. To obtain a graph combinatorial condition for structural controllability, we rely on the dynamic graph $\mathcal{D}_T$, which represents the time evolution of a system from $t = 0$ to $t = T$ [30–32]. Each node v_i in $\mathcal{L}$ is split into $T + 1$ copies $\{v_{i,0}, v_{i,1}, \ldots, v_{i,T}\}$, each copy $v_{i,t}$ represents the state of node v_i at time step t. We add a directed link $(v_{i,t} \rightarrow v_{j,t+1})$ for $t = 0, 1, \ldots, T - 1$ if they are connected by a directed link $(v_i \rightarrow v_j)$ in the original network, representing that the state of node v_j at time $t + 1$ depends on the state of its in neighbors at the previous time step. To account for the controller, for each independent input we create T nodes $u_{i,t}$ ($i = 1, 2, \ldots, M$; $t = 0, 1, \ldots, T - 1$), each representing a control signal (i.e., the value of the ith input at time step t). We draw a directed link $(u_{i,t} \rightarrow v_{j,t+1})$ for $t = 0, 1, \ldots, T - 1$ if $b_{ji} \neq 0$, where b_{ji} is an element of matrix $\mathbf{B}$.

According to Theorem 15.1 of Ref. [30], a linear system $(\mathbf{A}, \mathbf{B})$ is structurally controllable if and only if in the associated dynamic graph $\mathcal{D}_N$ node sets $U = \{u_{i,t} | i = 1, 2, \ldots, M; t = 0, 1 \ldots, N - 1\}$ [green nodes in Fig. 2(b)] and $V_N = \{v_{i,t=N} | i = 1, 2, \ldots, N\}$ (blue nodes) are connected by N disjoint paths (red links); i.e., there exists a set of disjoint paths $C = \{P_1, P_2, \ldots, P_N\}$ such that U contains the set of starting points and V_T is the set of end points. A path P of length l between nodes v_{i_0} and v_{i_l} is a sequence of l consecutive links $[(v_{i_0} \rightarrow v_{i_1}), (v_{i_1} \rightarrow v_{i_2}), \ldots, (v_{i_{l-1}} \rightarrow v_{i_l})]$ such that each node is traversed only once. Node v_{i_0} is the

starting point and v_{i_l} is the end point of P. Two paths P_1 and P_2 are disjoint if no node is traversed by both P_1 and P_2; a set of paths is disjoint if all paths in the set are pairwise disjoint.

A possible interpretation of this result is that if a P_i path has starting point u_{j,t_0} and end point v_{k,t_1}, we say that the signal $u_j(t_0)$ is assigned to set $x_k(t_1)$, the state of node v_k at time t_1, through path P_i. Therefore, we refer to path P_i as a control path. The clear meaning of the dynamic graph and the control paths makes this condition useful to formulate proofs and to interpret results. However, it is rarely implemented to test controllability of large networks, because the size of the dynamical graph grows as N^2, rendering such algorithms too slow. In the following, we provide a condition that only requires the dynamic graph $\mathcal{D}_1$ as input; therefore, it is more suitable for practical purposes.

It was shown in Refs. [10,30,33] that a linear system $(\mathbf{A}, \mathbf{B})$ is structurally controllable if and only if (i) in $\mathcal{D}_1$ we can connect nodes $U \cup V_0 = \{u_{i,t=0} | i = 1, 2, \ldots, M\} \cup \{v_{i,t=0} | i = 1, 2 \ldots, N\}$ [green nodes in Fig. 2(c)] and nodes $V_1 = \{v_{i,t=1} | i = 1, 2 \ldots, N\}$ (blue nodes) via N disjoint paths (red links) and (ii) all nodes are accessible from the inputs. This result can be understood as a self-consistent version of the previous condition involving $\mathcal{D}_N$: Instead of keeping track of the entire control paths as we previously did, we concentrate on a single time step. Consider the dynamic graph $\mathcal{D}_1$ representing the time evolution of the system from $t = 0$ to $t = 1$, and assume that the system is controllable. By definition we can set the state of each node independently at $t = 0$; therefore, we can treat them as control signals to control the system at a later time step. Now let us aim to control the system at $t = 1$; according to our previous condition, it is necessary that N disjoint paths exist between nodes $U \cup V_0 = \{u_{i,t=0} | i = 1, 2, \ldots, M\} \cup \{v_{i,t=0} | i = 1, 2 \ldots, N\}$ and nodes $V_1 = \{v_{i,t=1} | i = 1, 2 \ldots, N\}$. This is exactly requirement (i), together with the accessibility requirement (ii) it is a sufficient and necessary condition. Note that D_1 is a bipartite network (each link is connected to exactly one
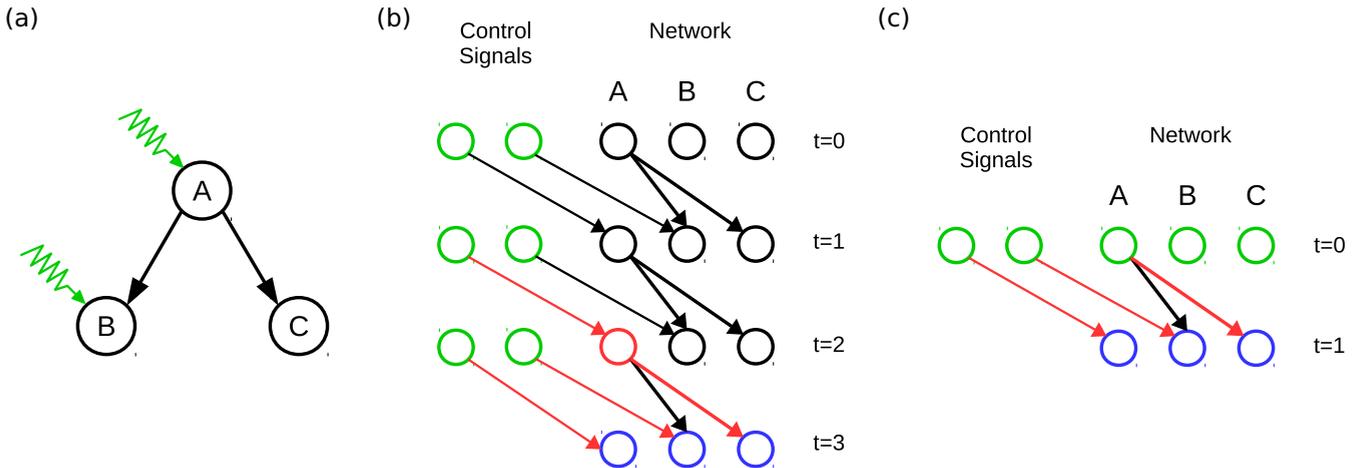
FIG. 2. Structural controllability of single-layer networks. (a) A single-layer network; we apply inputs to nodes v_A and v_B. (b) The dynamic graph $\mathcal{D}_N$ representing the time evolution of the dynamics from $t = 0$ to $t = N$. The system is controllable because we can connect the set of nodes representing control signals (green) to the set of nodes at $t = N$ (blue) via disjoint paths (red). (c) The dynamic graph $\mathcal{D}_1$ representing the time evolution of the dynamics from $t = 0$ to $t = 1$. The system is controllable, because we can connect the control signals and nodes at $t = 0$ (green) to the set of nodes at $t = 1$ (blue) via disjoint paths (red), and all nodes are accessible from control signals.

node in $U \cup V_0$ and one node in V_1) and each disjoint path in $\mathcal{D}_1$ is a single link.

The minimum input problem aims to identify the minimum number of inputs that guarantee controllability for a given network; in other words, the goal is to design a $\mathbf{B} \in \mathbb{R}^{N \times M}$ for a given $\mathbf{A}$ such that M is minimized. For this we consider the dynamic graph $\mathcal{D}_1$ without nodes representing control signals. We then find a maximum cardinality matching, where a matching is a set of links that do not share an end point. The matching is a set of disjoint paths connecting node sets V_0 and V_1. Controllability requires N disjoint paths between $U \cup V_0$ and V_1; therefore, $N_{\mathrm{i}} = N - N_{\mathrm{match}}$, where N_{match} is the size of the maximum matching (if $N_{\mathrm{match}} = N$, $N_{\mathrm{i}} = 1$). Allowing the inputs to be connected to multiple nodes, we can guarantee that all nodes are accessible from the inputs. Thus, we recovered the minimum input theorem of Liu *et al.* [10].

In summary, by relying on a self-consistent condition for structural controllability we re-derived the known result that identifying N_{i} is equivalent to finding a maximum matching in $\mathcal{D}_1$. In the next section we show that this self-consistent approach lends itself to being extended to the multiplex, multi-time-scale model defined by Eq. (1), allowing us to derive an analogous method to identify N_{i}.

B. Multiplex networks

To find the minimum number of inputs N_{i} for multiplex, multi-time-scale networks, we first extend the definition of the dynamic graph. We define the dynamic graph $\mathcal{D}_{\tau_{\mathrm{II}}}$ such that it captures the time evolution of a multiplex system defined by $(\mathbf{A}_{\mathrm{I}}, \mathbf{A}_{\mathrm{II}}, \mathbf{D}, \mathbf{B})$ and Eq. (1) from $t = 0$ to $t = \tau_{\mathrm{II}}$. For sake of brevity, we assume that $\tau_{\mathrm{I}} = 1$ and $\tau_{\mathrm{II}} \geqslant 1$, the case of $\tau_{\mathrm{I}} > 1$ and $\tau_{\mathrm{II}} = 1$ is treated similarly [Fig. 1(d)]. Each node v_i^{I} in layer I is split into $\tau_{\mathrm{II}} + 1$ copies $\{v_{i,0}^{\mathrm{I}}, v_{i,1}^{\mathrm{I}}, \ldots, v_{i,\tau_{\mathrm{II}}}^{\mathrm{I}}\}$; each node v_i^{II} in layer II is split into two copies $\{v_{i,0}^{\mathrm{II}}, v_{i,\tau_{\mathrm{II}}}^{\mathrm{II}}\}$, because layer II does not update during the intermediate time steps. We draw a link from $v_{i,t}^{\mathrm{I}}$ to $v_{j,t+1}^{\mathrm{I}}$ ($t = 0, 1, \ldots, \tau_{\mathrm{II}} - 1$) if they are connected in layer I by a directed link ($v_i^{\mathrm{I}} \to v_j^{\mathrm{I}}$), and similarly we connect $v_{i,0}^{\mathrm{II}}$ to $v_{j,\tau_{\mathrm{II}}}^{\mathrm{II}}$ if they are connected in layer II. In addition, we draw a link between each pair $v_{i,0}^{\mathrm{I}}$ and $v_{i,\tau_{\mathrm{II}}}^{\mathrm{II}}$ to account for the interconnectedness.

As a natural extension of the self-consistent approach introduced in Sec. III A, assume that the system is controllable. If the system is controllable, we can set the state of each node independently at $t = 0$. To control the system at $t = \tau_{\mathrm{II}}$, all nodes at $t = \tau_{\mathrm{II}}$ in $\mathcal{D}_{\tau_{\mathrm{II}}}$ (blue nodes in Fig. 1) have to be connected to a node at $t = 0$ or to a control signal (green nodes) via a disjoint path (red links). In other words, a linear two-layer system $(\mathbf{A}_{\mathrm{I}}, \mathbf{A}_{\mathrm{II}}, \mathbf{D}, \mathbf{B})$ is structurally controllable only if there exists $2N$ disjoint paths in the dynamic graph connecting node set $U \cup V_0 = \{u_{i,t} | i = 1, 2, \ldots, M; t = 0, 1, \ldots, \tau_{\mathrm{II}} - 1\} \cup \{v_{i,0}^{\mathrm{I}} | i = 1, 2, \ldots, N\} \cup \{v_{i,0}^{\mathrm{II}} | i = 1, 2, \ldots, N\}$ and node set $V_{\tau_{\mathrm{II}}} = \{v_{i,\tau_{\mathrm{II}}}^{\mathrm{I}} | i = 1, 2, \ldots, N\} \cup \{v_{i,\tau_{\mathrm{II}}}^{\mathrm{II}} | i = 1, 2, \ldots, N\}$.

To test whether the system is controllable by M independent inputs, we need to find a $\mathbf{B} \in \mathbb{R}^{N \times M}$ such that the system is controllable. We do not have to check all possibilities, because if such $\mathbf{B}$ exists, then the system is also controllable for $\mathbf{B}' \in \mathbb{R}^{N \times M}$, where $\mathbf{B}'$ has no zero elements; therefore, we only check the case when each input is connected to each node in

layer I. Given matrices $(\mathbf{A}_{\mathrm{I}}, \mathbf{A}_{\mathrm{II}}, \mathbf{D}, \mathbf{B}')$, we now have to count the number of disjoint paths connecting $U \cup V_0$ and $V_{\tau_{\mathrm{II}}}$ in the corresponding dynamic graph $\mathcal{D}_{\mathrm{II}}$. We find these paths using maximum flow: We set the capacity of each link and each node to 1, we then find the maximum flow connecting source node set $U \cup V_0$ to target node set $V_{\tau_{\mathrm{II}}}$ using any maximum flow algorithm of choice. If the system is structurally controllable, the maximum flow equals to $2N$; if it is less than $2N$, additional inputs are needed.

We can now identify the minimum number of inputs N_{i} by systematically scanning possible values of M. A simple approach is to first set $M = 1$ and test if the system is controllable. If not, increase M by one. Repeat this until the smallest M yielding full control is found. Significant increase in speed is possible if we find the minimum value of M using bisection. We initially know that $N_{\mathrm{i}}^{\mathrm{upper}} = N \geqslant N_{\mathrm{i}} \geqslant N_{\mathrm{i}}^{\mathrm{lower}} = 1$. We set $M = (N_{\mathrm{i}}^{\mathrm{upper}} + N_{\mathrm{i}}^{\mathrm{lower}})/2$ and test if the system is controllable. If yes, we set $N_{\mathrm{i}}^{\mathrm{upper}} = M$; if no, we set $N_{\mathrm{i}}^{\mathrm{lower}} = M$. We repeat this until $N_{\mathrm{i}}^{\mathrm{upper}} = N_{\mathrm{i}}^{\mathrm{lower}}$, which provides N_{i}. For implementation, we used Google OR-TOOLS and IGRAPH PYTHON packages [34,35].

The one-to-one coupling between layer I and layer II guarantees that full control is possible with at most N independent inputs; therefore, we often normalize N_{i} by N, i.e., $n_{\mathrm{i}} = N_{\mathrm{i}}/N$.

Note that in the above argument we rely on the test of structural controllability based on the dynamic graph, which was originally introduced for single-time-scale networks [30]. The sufficiency of the condition relies on the fact that the zero is the only degenerate eigenvalue of a matrix $\mathbf{A}$ if the nonzero elements of $\mathbf{A}$ are uncorrelated. However, this might not remain true for the spectrum of $\mathbf{A}^{\tau}$, where $\tau > 1$, due to correlations arising in the nonzero elements of $\mathbf{A}^{\tau}$. If a $\lambda \neq 0$ eigenvalue has larger geometric multiplicity than the multiplicity of 0, N_{i} would be larger than predicted by the dynamic graph; if a $\lambda \neq 0$ eigenvalue has larger geometric multiplicity than 1 but smaller than the multiplicity of zero, it does not affect N_{i}, but may require connecting an input to multiple nodes [12]. In the $\tau_{\mathrm{I}} > 0$ and $\tau_{\mathrm{II}} = 1$ case, a control signal is only injected into layer II every τ_{I} time step [Fig. 1(d)]; therefore, the spectrum of $\mathbf{A}_{\mathrm{II}}^{\tau_{\mathrm{I}}}$ becomes relevant. However, we are interested in large and sparse complex networks whose spectra are dominated by the zero eigenvalue [12]. Therefore, it is reasonable to expect that the spectrum of $\mathbf{A}^{\tau}$ will be dominated by zero eigenvalues as well, meaning that the minimum number of inputs is correctly given by this graph combinatorial condition. Furthermore, the one-to-one coupling between the layers guarantees that control is possible by only interacting with layer I directly.

So far, we developed a method to characterize controllability of a multiplex, multi-time-scale system based on the underlying network structure and the time scale of each of its layers. In the next section, we rely on these tools to systematically study how network characteristics and time scales affect N_{i}.

IV. RESULTS

In this section we investigate how different time scales and the degree distribution of each layer affect controllability. For time scales, we consider three scenarios: (i) the subsystems

operate on the same time scale, i.e., $\tau_I = \tau_{II} = 1$; (ii) layer I updates faster, i.e., $\tau_I = 1$ and $\tau_{II} > 1$; and (iii) layer II updates faster $\tau_I > 1$ and $\tau_{II} = 1$. To uncover the effect of degree distribution, we consider layers with Poisson (ER) or scale-free (SF) degree distribution, the latter meaning that the distribution has a power-law tail.

We generate scale-free layers using the static model [36]: We start with N unconnected nodes. Each node v_i is assigned two hidden parameters, $w_{in}(i) = i^{-\zeta_{in}}$ and $w_{out}(i) = i^{-\zeta_{out}}$, where $i = 1,2,\ldots,N$. The weights are then shuffled to eliminate any correlations of the in and out degree of individual nodes and between layers. We then randomly place L directed links by choosing the start and end points of the link with probability proportional to $w_{in}(i)$ and $w_{out}(i)$, respectively. For large N this yields the degree distribution

$$P_{in/out}^{SF}(k) = \frac{[c(1-\zeta_{in/out})^{1/\zeta_{in/out}}]}{\zeta_{in/out}} \frac{\Gamma(k-1/\zeta_{in/out},c[1-\zeta_{in/out}])}{\Gamma(k+1)},$$
(4)

where $c = L/N$ is equal to the average degree and $\Gamma(n,x)$ is the upper incomplete Γ function. For large k, $P_{in/out}^{SF}(k) \sim k^{-(1+1/\zeta_{in/out})} = k^{-\gamma_{in/out}}$, where $\gamma_{in/out} = 1 + 1/\zeta_{in/out}$ is the exponent characterizing the tail of the distribution.

To reduce the number of parameters, we only study layers with symmetric degree distribution, e.g., $P(k) = P_{in}(k) = P_{out}(k)$; however, the in and out degrees of a specific node can be different.

A. No time-scale separation ($\tau_I = \tau_{II} = 1$)

In the special case when both layers operate on the same time scale, i.e., $\tau_I = \tau_{II} = 1$ [Fig. 1(b)], there is no qualitative difference between the dynamics of the layers. The reason why the system cannot be treated as a single large network is that we are only allowed to directly interact with layer I. Recently, Iudice *et al.* developed methodology to identify N_i if the control signals can only be connected to a subset of

nodes [16]. However, the one-to-one coupling between the layers enables us to find N_i using a simpler approach.

Finding N_i for a single-layer network is equivalent to finding a maximum matching of the network [10]. A matching is a set of directed links that do not share starting or end points, and a node is unmatched if there is no link in the matching pointing at it. Liu *et al.* showed that full control of a network is possible if each unmatched node is controlled directly by an independent input; therefore, N_i is provided by the minimum number of unmatched nodes. To determine N_i for a two-layer network, we first find a maximum matching of the combined network of layer I and layer II. If there are no unmatched nodes in layer II, we only have to interact with layer I; therefore, we are done. If a node v_i^{II} is unmatched in layer II, v_i^I is necessarily matched by some node v_j^I; otherwise, the size of the matching could be increased by adding $(v_i^I \rightarrow v_i^{II})$. By taking out the link $(v_j^I \rightarrow v_i^I)$ from the matching and including $(v_i^I \rightarrow v_i^{II})$ the size of the maximum matching does not change, and we moved the unmatched node from layer II to layer I. We repeat this for all unmatched nodes in layer II. (Note that it may be necessary to connect inputs to additional nodes so that all nodes are reached by the control signals. Due to the one-to-one coupling between the layers, this too can be accomplished by interacting only with layer I.) This simplified method allows faster identification of N_i using the Hopcroft-Karp algorithm [37] and analytically solving $n_i = N_i/N$ for random networks based on calculating the fraction of always matched nodes, as described in Appendix A [38–41].

First, we measure n_i while fixing the average degree of layer II (c_{II}) and varying the average degree of layer I (c_I). For both ER-ER and SF-SF networks, we find that n_i decreases for increasing values of c_I and converges to $n_i^{II} = N_i^{II}/N$, the normalized number of inputs needed to control layer II in isolation [Fig. 3(a)]. The latter observation is easily understood. n_i is determined by the fraction of unmatched nodes in the combined network of the two layers; if c_I is high enough, layer I is perfectly matched. Therefore, all unmatched
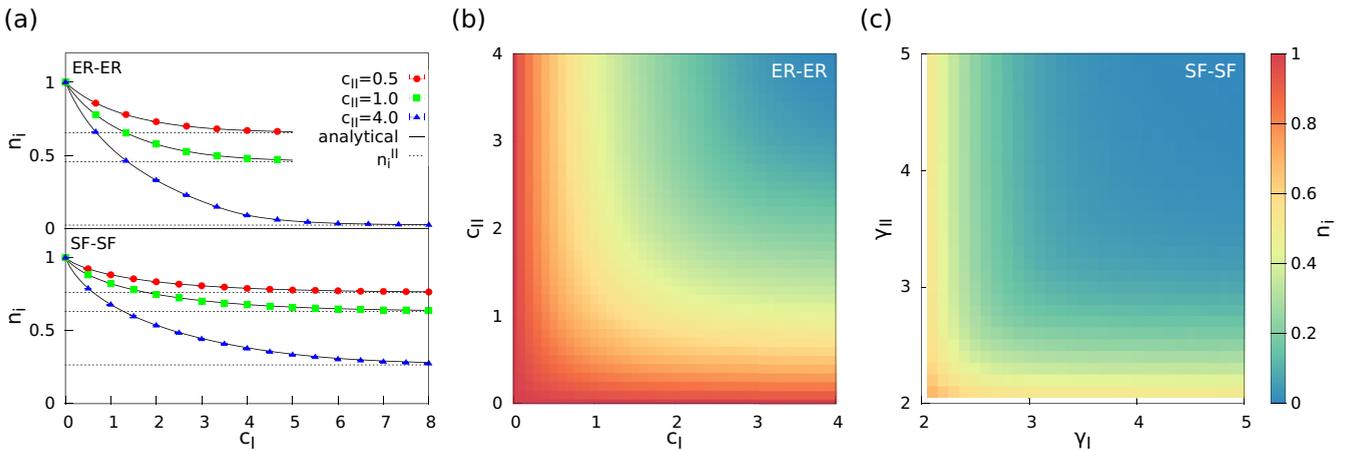
FIG. 3. No time-scale separation. (a) Number of inputs n_i in function of c_I for ER-ER and SF-SF ($\gamma_I = \gamma_{II} = 2.5$) networks. The circles represent simulations, the continuous line is the analytical solution, and the dashed line is the analytical solution of n_i^{II}, the number of independent inputs necessary to control layer II in isolation [10]. (b) n_i for ER-ER networks with varying average degrees c_I and c_{II}. In both layers $P(k) = P(k_{in}) = P(k_{out})$; therefore, the heat map is symmetric with respect to the diagonal. Increasing c in either layer enhances controllability. (c) n_i for SF-SF networks with $c_I = c_{II} = 4.0$ and varying degree exponents γ_I and γ_{II}. Increasing degree heterogeneity in either layer increases n_i. Each data point is the average over ten randomly generated networks with $N = 10\,000$. The standard deviation of the measurements remains below 0.01.
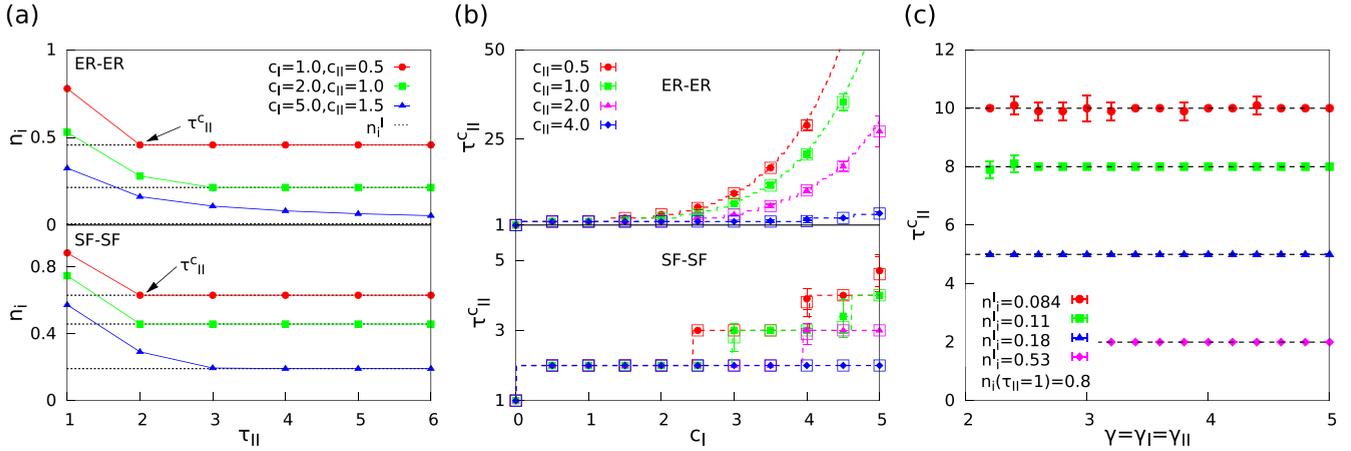
FIG. 4. Layer I updates faster. (a) Number of inputs n_i for single ER-ER and SF-SF ($\gamma_I = \gamma_{II} = 2.5$) networks with $N = 10\,000$ and varying time-scale parameter τ_{II}. The number of inputs n_i monotonically decreases with increasing τ_{II}, and for $\tau_{II} \geqslant \tau_{II}^c$ and $n_i = n_i^I$. (b) The critical time-scale parameter τ_{II}^c for ER-ER and SF-SF ($\gamma_I = \gamma_{II} = 2.5$) networks with varying average degree c_I and c_{II}. The crosses represent direct measurements of τ_{II}^c; the squares represent the approximation obtained by applying Eq. (7) to measurements of $n_i(\tau_{II} = 1)$ and n_i^I; and the dashed line is an approximation obtained using analytically calculated expectation values of $n_i(\tau_{II} = 1)$ and n_i^I. (c) We measure τ_{II}^c for SF-SF networks with the same $n_i(\tau_{II} = 1)$ and n_i^I as a function of $\gamma = \gamma_I = \gamma_{II}$. Equation (7) predicts that τ_{II}^c remains constant (dashed line), in line with our observations. For (b),(c), each data point is the average over ten randomly generated networks with $N = 10\,000$ and error bars represent the standard deviation.

nodes are in layer II. Based on the same argument, n_i^I also serves as a lower bound for n_i.

Varying both c_I and c_{II} for ER-ER and both γ_I and γ_{II} for SF-SF with constant average degrees $c_I = c_{II}$, we find that dense networks require fewer inputs than sparse networks [Fig. 3(b)] and degree heterogeneity makes control increasingly difficult [Fig. 3(c)], in line with results for single-layer networks [10]. We also observe that n_i is invariant to exchanging layer I and layer II. This is explained by the fact that the size of the maximum matching is invariant to flipping the direction of all links, and on the ensemble level this is the same as swapping the two layers for networks with $P(k_{in}) = P(k_{out})$.

In summary, for no time-scale separation controllability is equally affected by the network structure of both layers, and n_i is greater or equal to the number of inputs necessary to control any of its layers in isolation. Similarly to single-layer networks, networks with low average degree and high degree heterogeneity require more independent inputs than sparse homogeneous networks.

B. Layer I updates faster ($\tau_I = 1$, $\tau_{II} > 1$)

In the previous section we found that the network structure of the two layers equally affect n_i if $\tau_I = \tau_{II} = 1$. This is not the case if the time scales are different; for example, if layer I updates faster than layer II, we expect that we need fewer inputs than in the same time-scale case by the virtue of having more opportunity to interact with the faster system [Fig. 1(b)]. In this section we systematically study this effect using the algorithm described in Sec. III B and analytical arguments.

By measuring n_i for ER-ER and SF-SF networks as a function of τ_{II}, we find that n_i monotonically decreases with increasing τ_{II} [Fig. 4(a)], confirming our expectations. For both ER-ER and SF-SF networks $n_i(\tau_{II})$ converges to $n_i^I = N_i^I/N$, which is the normalized number of inputs needed to control layer I in isolation. This can be understood by the following

argument. Suppose that $\tau_{II} = N$, the maximum number of time steps needed to impose control on any network with N nodes [42]. We use the state of layer I at $t = 0$ to set the state of layer II at $t = N$, and we have N time steps to impose control on layer I as if it was just by itself. For a given network we define the critical time-scale parameter τ_{II}^c as the minimum value of τ_{II} for which $n_i(\tau_{II}) = n_i^I$. Above the critical time-scale separation, layer I completely determines $n_i(\tau_{II})$ independent of the structure of layer II; in other words, the multiplex nature of the system no longer plays a role in determining n_i.

Measuring τ_{II}^c we find that for both ER-ER and SF-SF networks τ_{II}^c monotonically increases with increasing c_I for fixed c_{II} and decreases with increasing c_{II} for fixed c_I [Fig. 4(b)]. That is, τ_{II}^c is the highest if layer I is dense and layer II is sparse. SF-SF networks have significantly lower τ_{II}^c than ER-ER networks with the same average degree.

To understand the observed pattern, we provide an approximation to calculate τ_{II}^c. We call a node v_i^I externally controlled if in the dynamic graph $v_{i,\tau_{II}}^I$ is connected to an external signal $u_{j,t}$ via a disjoint control path [e.g., nodes v_A^I and v_B^I in Fig. 1(c)], and the number of such nodes is denoted by $N_e(\tau_{II})$. We have previously shown that we require N_i^I independent inputs at τ_{II}^c. For each independent input and each time step, we have one control signal $u_{i,t}$; therefore, we need time scale parameter

$$\tau_{II}^c = \lceil N_e(\tau_{II}^c)/N_i^I \rceil \qquad (5)$$

to insert enough signals required by the $N_e(\tau_{II}^c)$ externally controlled nodes, where $\lceil \cdot \rceil$ is the ceiling function. Equation (5) is not yet useful as it contains τ_{II}^c on both sides. Observing that $N_e(\tau_{II})$ is a monotonically increasing function of τ_{II} and $N_e(\tau_{II} = 1) = N_i(\tau_{II} = 1)$, we can write

$$N_i(\tau_{II} = 1) \leqslant N_e(\tau_{II}^c) \leqslant N. \qquad (6)$$

In the special case when layer II is fully connected, $\tau_{II}^c = 1$ and $N_e(\tau_{II}^c = 1) = N_i(\tau_{II} = 1)$. In the case when layer II is entirely

disconnected, i.e., is composed of isolated nodes, $N_e(\tau_{II}) = N = N_i(\tau_{II} = 1)$. These two opposite limiting cases suggest that it is reasonable to approximate $N_e(\tau_{II}^c)$ by its lower bound,

$$\tau_{II}^c \approx \lceil N_i(\tau_{II} = 1)/N_i^I \rceil, \tag{7}$$

which entirely depends on quantities that we can easily measure or analytically compute. We find that Eq. (7) preforms remarkably well: Figure 4(b) compares direct measurements of τ_{II}^c to approximations obtained by using measurements and analytically computed values of $N_i(\tau_{II} = 1)$ and N_i^I. The approximation based on measurements outperforms the analytical calculations because the analytical results provide the expectation value of the numerator and denominator for ER and SF network ensembles, and therefore the ceiling function is applied to the fraction of averages, instead of averaging after applying the ceiling function. To further test Eq. (7), we fix $n_i(\tau_{II} = 1)$ and n_i^I and we analytically calculate c_I and c_{II} for SF-SF networks with varying degree exponent $\gamma = \gamma_I = \gamma_{II}$ using the framework developed in Appendix A. Then we generate SF-SF networks and measure τ_{II}^c as a function of γ. The approximation predicts that τ_{II}^c remains constant, in line with our observations [Fig. 4(c)].

The good performance of Eq. (7) is partly due to the role of the ceiling function because it is insensitive to changes in the numerator that are small compared to N_i^I. Indeed, errors are more pronounced if $N_i(\tau_{II} = 1)/N_i^I$ is close to an integer [e.g., data point $c_I = 4.5$ and $c_{II} = 1$ in Fig. 4(b) for SF-SF] or $N_i(\tau_{II} = 1) \gg N_i^I$ [e.g., data points $n_i^I = 0.084$ in Fig. 4(c)].

What we learn from this approximation is that τ_{II}^c depends only indirectly on the degree distribution of layer I and layer II through the control properties of the system without time-scale separation – $N_i(\tau_{II} = 1)$ and N_i^I. In Sec. IV A, we showed that $N_i(\tau_{II} = 1) \geqslant N_i^{II}$; therefore, τ_{II}^c is expected to be large if layer I is easy to control (e.g., it is dense and has homogeneous degree distribution) and layer II is hard to control (e.g., it is sparse and has heterogeneous degree distribution).

In summary, if layer I updates faster, time-scale separation enhances controllability up to a critical time-scale parameter τ_{II}^c, above which $n_i(\tau_{II}) = n_i^I$ and is completely determined by layer I. The critical time-scale parameter τ_{II}^c largely depends on the controllability of the system without time-scale separation;
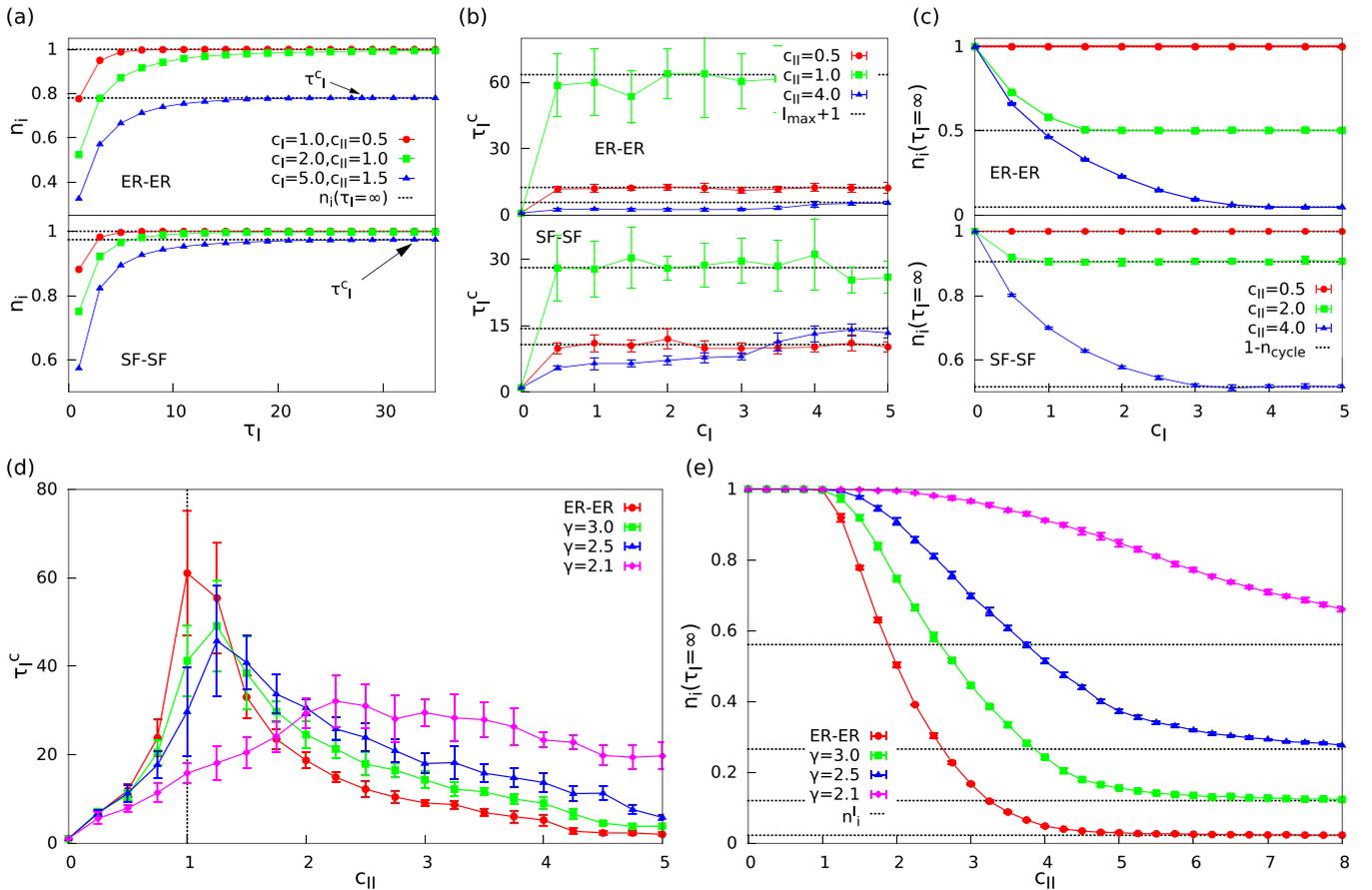
FIG. 5. Layer II updates faster. (a) Number of inputs n_i for single ER-ER and SF-SF ($\gamma_I = \gamma_{II} = 2.5$) networks with $N = 10\,000$ and varying time-scale parameter τ_I. The number of inputs n_i monotonically increases with increasing τ_I; for $\tau_I \geqslant \tau_I^c$, $n_i = n_i(\tau_I = \infty)$. (b) τ_I^c as a function of c_I. For $c_I \leqslant 1$, τ_I^c quickly reaches its upper bound; for $c_I > 1$, the convergence is somewhat delayed. (c) $n_i(\tau_I = \infty)$ as a function of c_I. Increasing c_I facilitates control, until n_i reaches its lower bound. (d) τ_I^c as a function of c_{II} with fix $c_I = 4.0$. The peak of τ_I^c corresponds to the critical point where the giant strongly connected component in layer II emerges. (e) $n_i(\tau_I = \infty)$ as a function of c_{II} with fix $c_I = 4.0$. For $c_{II} < 1$, layer II does not contain cycles; therefore, $n_i(\tau_I = \infty) = 1$. For large c_{II}, layer II can be completely covered with cycles, and $n_i(\tau_I = \infty)$ is determined by n_i^I. For (b)–(e), each data point is the average over ten randomly generated networks with $N = 10\,000$ and error bars represent the standard deviation.

it is expected to be large if layer I is easy to control and layer II is hard to control.

C. Layer II updates faster ($\tau_I > 1$, $\tau_{II} = 1$)

Finally, we investigate the case when layer II operates faster than layer I, i.e., $\tau_I > 1$ and $\tau_{II} = 1$ [Fig. 1(d)]. Measurements show that n_i monotonically increases in function of τ_I for both ER-ER and SF-SF networks, and n_i remains constant if $\tau_I \geqslant \tau_I^c$, where τ_I^c is defined for a single network [Fig. 5(a)]. To understand these results, consider the following argument: Some nodes of layer II are internally controlled, meaning that the state of these nodes at $t = \tau_I$ is set by the state of nodes within layer II at $t = 0$ connected to them via disjoint control paths [node v_C^{II} in Fig. 1(d)], while the rest of the nodes of layer II have to be controlled by nodes of layer I. The maximum number of internally controlled nodes is set by the number of disjoint paths of length τ_I. A directed open path traversing l links in layer II yields a path in the dynamic graph of at most length l; therefore, if $\tau_I > l$ the path can no longer be used for control. For example, in Fig. 1(a) path ($v_B^{II} \rightarrow v_A^{II}$) consists of a single link; therefore, we can use it for control if $\tau_I = 1$ [Fig. 1(b)] and it is no longer useful if $\tau_I > 1$ [Fig. 1(d)]. However, a cycle can support a path in the dynamic graph of any length, e.g., the self-loop ($v_C^{II} \rightarrow v_C^{II}$) in Fig. 1. This predicts that

$$n_i(\tau_I = \infty) \geqslant 1 - n_{\text{cycle}}, \qquad (8)$$

where $n_{\text{cycle}} = N_{\text{cycle}}/N$ is the maximum fraction of nodes that can be covered with cycles in layer II. Furthermore, it also means that

$$\tau_I^c \leqslant l_{\max} + 1, \qquad (9)$$

where $l_{\max}$ is the maximum length of a control path that does not involve cycles, a quantity that only depends on the structure of layer II. We provide the formal definition $l_{\max}$ and algorithms to measure n_{cycle} and $l_{\max}$ in Appendix B.

Both $l_{\max}$ and n_{cycle} depend only on layer II; furthermore, both strongly depend on whether layer II contains a strongly connected component (SCC). Uncorrelated random directed networks—both ER and SF—undergo a percolation transition at $c = 1$ [43]. If $c < 1$, the network is composed of small tree components, meaning the $n_{\text{cycle}} = 0$ and $l_{\max}$ is equal to the diameter D of the network. If the system is in the critical point $c = 1$, the size of the largest component S diverges as $N \rightarrow \infty$, but the relative size S/N remains zero. The largest component contains a small number of cycles; therefore, D is only approximately equal to $l_{\max}$. If $c > 1$, a unique giant SCC emerges which contains cycles; therefore, $n_{\text{cycle}} > 0$ and $l_{\max}$ is no longer directly connected to the diameter. Rigorous mathematical results show that the diameter of the ER model scales as $D \sim \log(N)$ for $c \neq 1$, and $D \sim N^{1/3}$ for $c = 1$, the latter corresponding to the percolation transition point [44], suggesting that the critical time-scale parameter τ_I^c also depends on N. Indeed, Fig. 6 shows that τ_I^c monotonically increases with N for both ER-ER and SF-SF networks.

We now scan possible values of c_I while keeping c_{II} and N fixed, we find that $n_i(c_I)$ and $\tau_I^c(c_I)$ quickly converge to their respective lower and upper bounds provided by Eqs. (8) and (9) [Figs. 5(b) and 5(c)]. Varying c_{II} and keeping c_I fixed shows more intricate behavior: $\tau_I^c(c_{II})$ increases, peaks, and decreases again [Fig. 5(d)]. This is explained by changes in the structure of layer II: For small c_{II} the network is composed of small components with tree structure; increasing c_{II} agglomerates these components, thus increasing $l_{\max}$. For large c_{II}, a giant SCC exists, supporting many cycles; as c_{II} increases, more and more nodes can be covered with cycles reducing $l_{\max}$. At the critical point $c_{II}^* = 1$ the giant SCC emerges, and the largest component consists of N^α nodes ($0 < \alpha < 1$) with only few cycles, providing the peak of $\tau_I^c(c_{II})$. Although $c_{II}^* = 1$ for both ER and SF networks in the $N \rightarrow \infty$ limit, finite size effects delay the peak of τ_I^c for SF-SF networks. Below the transition point, τ_I^c is smaller for ER-ER networks than for SF-SF networks with the same average degree. In contrast,
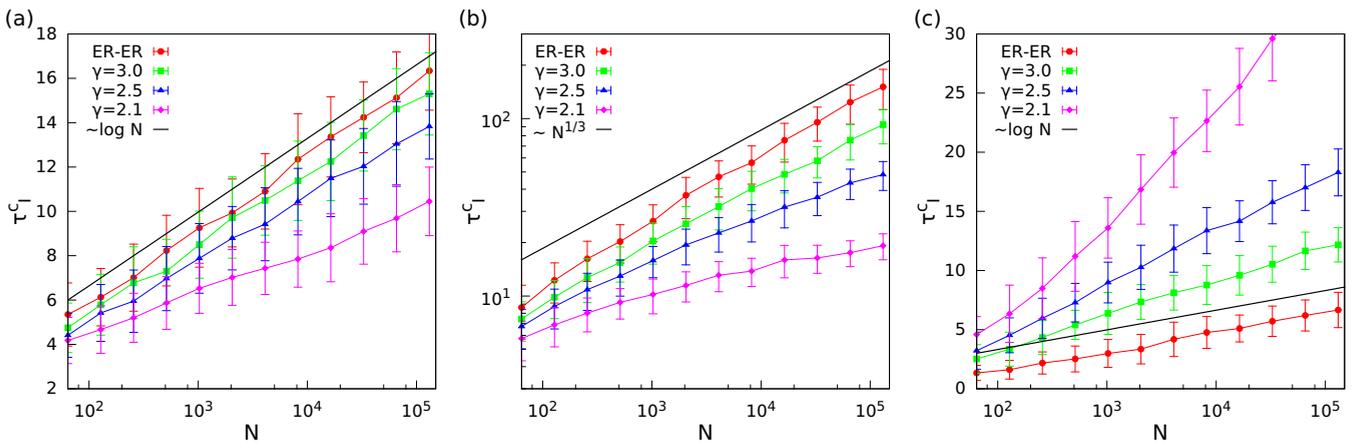
FIG. 6. Layer II updates faster: network size effects. Critical time-scale parameter for ER-ER networks and SF-SF networks with varying network size N. (a) Layer II has no giant strongly connected component ($c_{II} = 0.5 < 1$), $l_{\max}$ equals the diameter D of layer II, which scales as $D \sim \log N$ for ER networks, and the diameter of SF networks is smaller than the diameter of ER networks with the same average degree. The fact that $l_{\max} + 1 \geqslant \tau_I^c$ suggests that $\tau_I^c \sim \log(N)$. (b) At the critical point $c_{II} = 1.0$ the diameter of ER networks scales as $D \sim N^{1/3}$, suggesting that τ_I^c scales as a power law of N. (c) Above the critical point ($c_{II} = 4.0 > 1$) there is no direct connection between D and τ_I^c; nonetheless, observations suggest $\tau_I^c \sim \log N$. In contrast with the $c_{II} \leqslant 1$ case, τ_I^c increases more rapidly for SF-SF networks than for ER-ER networks. Each data point is the average over 100 randomly generated networks with $c_I = 4.0$ and error bars represent the standard deviation.

above the transition point SF-SF networks have larger τ_I^c. A likely explanation is that the cycle cover of SF networks is smaller than the cycle cover of ER networks with the same average degree; thus, more nodes can potentially participate in the longest control path that does not involve cycles.

The number of inputs above the critical time-scale parameter $n_i(\tau_I = \infty)$ is also affected by the cycle cover of layer II [Fig. 5(e)]: For $c_{II} < 1$, layer II does not contain cycles yielding $n_i(\tau_I = \infty) = 1$; for large c_{II}, layer II can be completely covered with cycles, and $n_i(\tau_I = \infty)$ is determined by n_i^I, the number of inputs needed to control layer I in isolation.

In summary, if layer II updates faster, time-scale separation reduces controllability up to a critical time-scale parameter τ_I^c. For the model networks, the value of τ_I^c depends on whether layer II has a giant SCC; τ_I^c has the highest value at the percolation threshold of layer II. If layer II does not contain a giant SCC, degree heterogeneity decreases τ_I^c; above the percolation threshold homogeneous networks have lower τ_I^c. For all time-scale parameters, it remains true that ER-ER networks require less independent inputs than SF-SF networks with the same average degree.

V. CONCLUSIONS

Here we explored controllability of interconnected complex systems with a model that incorporates common properties of these systems: (i) It consists of two layers, each described by a complex network; (ii) the operation of each layer is characterized by a different, but often comparable time scale; and (iii) the external controller interacts with only one layer directly. We focused on two-layer multiplex networks, meaning that we assume one-to-one coupling between the nodes of the two layers. Our motivation for this choice was to ensure analytical tractability and to isolate the specific role of time scales from the effect of more complex multilayer network structure. Results obtained for more general multilayer networks will ultimately be shaped by a variety of features such as complex interconnectivity structure, correlations in network structure, and details of dynamics. However, even by studying multiplex networks, we uncovered nontrivial phenomena, attesting that without understanding each individual effect, it is impossible to fully understand a system as a whole.

Using structural controllability we were able to solve the model, thereby directly linking controllability to a graph combinatorial problem. We investigated the effect of network structure and time scales by measuring the minimum number of independent inputs needed for control, N_i. Overall, we found that dense networks with homogeneous degree distribution require less inputs than sparse heterogeneous networks, in line with previous results for single-layer networks [10]. We showed that if we control the faster layer directly, N_i decreases with increasing time-scale difference, but only up to a critical value. Above the critical time-scale difference, N_i is completely determined by the faster layer and we do not have to take into account the multiplex structure of the system. This critical time-scale separation is expected to be large if the faster layer would be easy to control and the slower layer would be hard to control in isolation. If we interact with the slower layer, control is increasingly difficult for increasing time-scale difference, again up to a critical value, above which

N_i still depends on the structure of both layers. In this case the critical time-scale difference largely depends on the longest control path that does not involve cycles in the faster layer.

Although our model offers only a stylized description of real systems, it is a tractable first step towards understanding the role of time scales in control of interconnected networks. By identifying the network characteristics that affect important measures of controllability, such as minimum number of inputs needed for control and critical time-scale difference, our results serve as a starting point for future work that aims to relax some of the model's assumptions. Some of these extensions are relatively straightforward using the tool set developed here, for example, the effect of higher-order network structures can be studied by adding correlations to the underlying networks. Other extensions are more challenging, e.g., if the interconnection between the layers is incomplete or the layers contain different number of nodes, the minimum input problem is computationally more difficult; therefore, investigating such systems would require development of efficient approximation schemes. Structural control theory does not take the link weights into account; therefore, answering questions that depend on the specific strength of the connections require the development of different tools. For example, for continuous-time systems the time scales are encoded in the strength of the interactions, or the minimum control energy also depends on value of the link weights.

APPENDIX A: ANALYTICAL SOLUTION FOR $\tau_I = \tau_{II} = 1$

In this section we derive an analytical solution of $n_i = N_i/N$ in case of $\tau_I = \tau_{II} = 1$ for two-layer random networks with predefined degree distribution as defined in Sec. IV. This network model is treelike in the $N \to \infty$ limit; therefore, it lends itself to the generating function formalism. The approach described here is based on calculating the fraction of nodes that are matched in all possible maximum matchings [39]. This solution is substantially simpler than the one described in Ref. [10]; however, it applies only to bipartite networks (or to bipartite representations of directed networks) and cannot be generalized to unipartite networks.

We aim to calculate the expected size of the maximum matching of the following undirected bipartite network $\mathcal{B}$. Layer I $\mathcal{L}_I$ and layer II $\mathcal{L}_{II}$ are generated independently using either the ER model or the SF model; V_I and E_I are the node and link sets of $\mathcal{L}_I$ and V_{II} and E_{II} are the node and link sets of $\mathcal{L}_{II}$, respectively. Each node in $v_i^I \in V_I$ is split into two copies $v_{i,0}^I \in V_0^I$ and $v_{i,1}^I \in V_1^I$; we draw a link $(v_{i,0}^I - v_{j,1}^I)$ if there exists a link $(v_i^I \to v_j^I)$ in $\mathcal{L}_I$. We treat $\mathcal{L}_{II}$ similarly. We then add links $(v_{i,0}^I - v_{i,1}^{II})$ for all i. That is, all links in $\mathcal{B}$ connect exactly one node in $V_0^I \cup V_0^{II}$ to one node in $V_1^I \cup V_1^{II}$. Nodes

in $V_0^{\mathrm{I}} \cup V_1^{\mathrm{I}}$ belong to layer I, and nodes in $V_0^{\mathrm{II}} \cup V_1^{\mathrm{II}}$ belong to layer II. The network $\mathcal{B}$ is the undirected version of the dynamical graph $\mathcal{D}_1$ without control signals.

In general, multiple possible maximum matchings may exist in a network. We first calculate the fraction of nodes that are matched in all possible maximum matchings. It was shown in Ref. [39] that in any network $\mathcal{G}$ a node v is always matched if and only if at least one of its neighbors is not always matched in $\mathcal{G} \setminus v$, where $\mathcal{G} \setminus v$ is the network obtained by removing node v from $\mathcal{G}$. We translate this rule to a set of self-consistent equations to calculate the expected fraction of always matched nodes in our random network model in the $N \to \infty$ limit. We provide comments on the issues of applying the rule proven for finite networks to infinite ones at the end of this section.

To proceed we define a few probabilities. We randomly select a link e connecting two nodes $v_{i,0}^{\mathrm{I}} \in V_0^{\mathrm{I}}$ and $v_{j,1}^{\mathrm{I}} \in V_1^{\mathrm{I}}$. Let θ_0^{I} be the probability that $v_{i,0}^{\mathrm{I}}$ is always matched in $\mathcal{B} \setminus e$ and θ_1^{I} be the probability that $v_{j,1}^{\mathrm{I}}$ is always matched in $\mathcal{B} \setminus e$. Similarly, we randomly select a link e connecting a node $v_{i,0}^{\mathrm{I}} \in V_0^{\mathrm{I}}$ with a node $v_{i,1}^{\mathrm{II}} \in V_1^{\mathrm{II}}$. Let $\theta_0^{\mathrm{I,II}}$ be the probability that node $v_{i,0}^{\mathrm{I}}$ is always matched in $\mathcal{B} \setminus e$ and $\theta_1^{\mathrm{I,II}}$ be the probability that node $v_{i,1}^{\mathrm{II}}$ is always matched in $\mathcal{B} \setminus e$. The probabilities θ_0^{II} and θ_1^{II} are defined similarly. According to the rule described above these quantities can be determined by the set of equations

$$
\begin{aligned}
\theta_0^{\mathrm{I}} &= 1 - H^{\mathrm{I}}(\theta_1^{\mathrm{I}})\theta_1^{\mathrm{I,II}}, \\
\theta_1^{\mathrm{I}} &= 1 - H^{\mathrm{I}}(\theta_0^{\mathrm{I}}), \\
\theta_0^{\mathrm{I,II}} &= 1 - G^{\mathrm{I}}(\theta_1^{\mathrm{I}}), \\
\theta_1^{\mathrm{I,II}} &= 1 - G^{\mathrm{II}}(\theta_0^{\mathrm{II}}), \\
\theta_0^{\mathrm{II}} &= 1 - H^{\mathrm{II}}(\theta_1^{\mathrm{II}}), \\
\theta_1^{\mathrm{II}} &= 1 - H^{\mathrm{II}}(\theta_0^{\mathrm{II}})\theta_0^{\mathrm{I,II}},
\end{aligned} \tag{A1}
$$

where $G^{\mathrm{I/II}}(x) = \sum_{k=0}^{\infty} P^{\mathrm{I/II}}(k) x^k$ are the generating functions of the degree distributions and $H^{\mathrm{I/II}}(x) = \sum_{k=1}^{\infty} k/\langle k \rangle P^{\mathrm{I/II}}(k) x^{k-1}$ are the generating functions of the excess degree distributions.

If we remove a node v which is not always matched, the size of the maximum matching does not decrease. However, if v is matched in all maximum matchings, the number of matched nodes will decrease by two. Therefore, to count the size of the maximum matching, we first count the number of nodes that are always matched. By doing so, we have double counted the case when an always matched node is matched by another always matched one. This case occurs for each link e that connects two nodes that are not always matched in $\mathcal{G} \setminus e$. Combining these two contributions, the expected number of links in the matching is

$$
\begin{aligned}
N_{\mathrm{match}} &= N\big[1 - G^{\mathrm{I}}(\theta_1^{\mathrm{I}})\theta_1^{\mathrm{I,II}}\big] + N\big[1 - G^{\mathrm{I}}(\theta_0^{\mathrm{II}})\big] \\
&\quad + N\big[1 - G^{\mathrm{II}}(\theta_1^{\mathrm{I}})\big] + N\big[1 - G^{\mathrm{II}}(\theta_0^{\mathrm{II}})\theta_0^{\mathrm{I,II}}\big] \\
&\quad - c_{\mathrm{I}} N\big(1 - \theta_0^{\mathrm{I}}\big)\big(1 - \theta_1^{\mathrm{I}}\big) - N\big(1 - \theta_0^{\mathrm{I,II}}\big)\big(1 - \theta_1^{\mathrm{I,II}}\big) \\
&\quad - c_{\mathrm{II}} N\big(1 - \theta_0^{\mathrm{II}}\big)\big(1 - \theta_1^{\mathrm{II}}\big),
\end{aligned} \tag{A2}
$$

where the first four terms count the number of nodes that are always matched in V_0^{I}, V_1^{I}, V_0^{II}, and V_1^{II}, respectively, and the last three terms correct the double counting. The expected number of independent inputs needed is determined by the number of unmatched nodes in V_1^{I} and V_1^{II}:

$$
N_{\mathrm{i}} = 2N - N_{\mathrm{match}}. \tag{A3}
$$

Due to the links between layer I and layer II, the size of the maximum matching is at least N, meaning that $N_{\mathrm{i}} \leqslant N$. Therefore, we normalize N_{i} by N, yielding

$$
\begin{aligned}
n_{\mathrm{i}} &= G^{\mathrm{I}}(\theta_1^{\mathrm{I}})\theta_1^{\mathrm{I,II}} + G^{\mathrm{I}}(\theta_0^{\mathrm{II}}) + G^{\mathrm{II}}(\theta_1^{\mathrm{I}}) + G^{\mathrm{II}}(\theta_0^{\mathrm{II}})\theta_0^{\mathrm{I,II}} - 2 \\
&\quad + c_{\mathrm{I}}\big(1 - \theta_0^{\mathrm{I}}\big)\big(1 - \theta_1^{\mathrm{I}}\big) + \big(1 - \theta_0^{\mathrm{I,II}}\big)\big(1 - \theta_1^{\mathrm{I,II}}\big) \\
&\quad + c_{\mathrm{II}}\big(1 - \theta_0^{\mathrm{II}}\big)\big(1 - \theta_1^{\mathrm{II}}\big).
\end{aligned} \tag{A4}
$$

Comments on matchings in the configuration model. The method we described to calculate the expected size of the maximum matching does not work for unipartite ER or SF networks generally. The reason for this is that above a critical average degree c^* a densely connected subgraph forms, which is referred to as the core of the network (sometimes leaf removal core or computational core) [45–47]. To derive Eq. (A1), we assume that the neighbors of a randomly selected node v are independent of each other in $\mathcal{B} \setminus v$ and removing a single node does not influence macroscopic properties, e.g., θ. The effect of the core is that these assumptions no longer hold and removing just a few nodes may drastically change the number of always matched nodes. A possible way of circumventing this problem is to introduce a new category of nodes: In addition to keeping track of nodes that are sometimes matched and always matched, we separately account for nodes that are almost always matched [38].

The reason why the calculation works for bipartite networks is that a core in the bipartite network will have two sides: All nodes on one side will be always matched and all nodes on other will be sometimes matched [39–41]. If the expected size of the core on the two sides is different, finite removal of nodes will not change macroscopic properties. If the expected sizes of the two sides of the core are the same, removal of finite nodes may change which side is always matched and which side is sometimes matched [39]. However, this does not change expected fraction of matched nodes and therefore does not interfere with the calculations.

APPENDIX B: ALGORITHMS

1. Cycle cover (N_{cycle})

To find the maximum cycle cover of a directed network $\mathcal{L}$, we assign weight 0 to each link in $\mathcal{L}$, and we add a self-loop with weight 1 to each node that does not already have a self-loop. Then we find the minimum weight maximum directed matching in $\mathcal{L}$ augmented with self-loops by converting the problem to a minimum cost maximum flow problem. The maximum matching is guaranteed to be perfect, because each node has a self-loop. The minimum weight perfect matching in the directed network corresponds to a perfect cycle cover
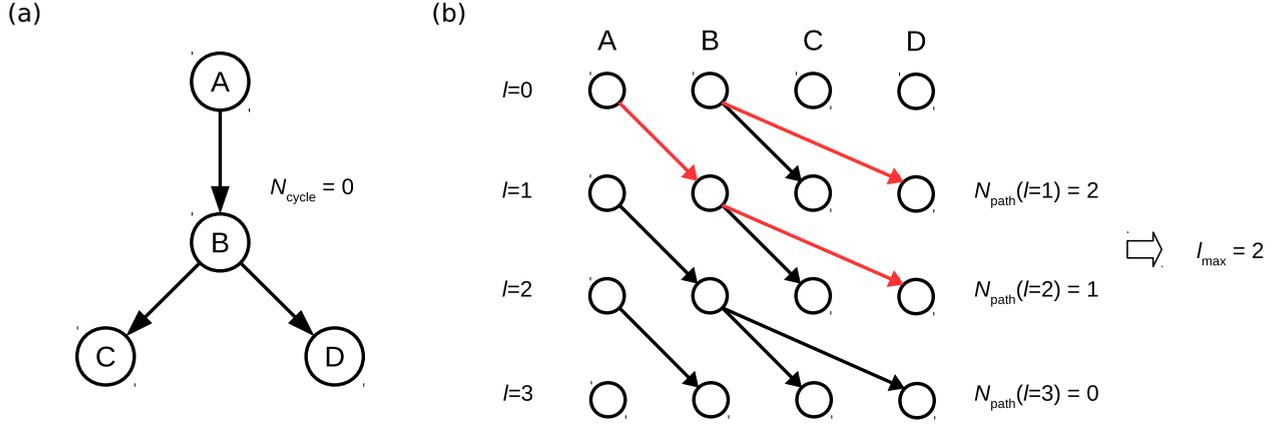
FIG. 7. $l_{\max}$, example 1. (a) A directed network with tree structure, therefore not containing cycles. The diameter $D = 2$ is the length of the longest path. (b) We count the maximum number of disjoint control paths $N_{\text{path}}(l)$ which connect nodes at time step 0 with nodes at time step l. We find that $l' = 3$ is the smallest value of l such that $N_{\text{path}}(l) = N_{\text{cycle}} = 0$; therefore, $l_{\max} = 2$. There are no cycles; therefore, $l_{\max} = D$.

where the number of self-loops with weight 1 is minimized. Therefore, the maximum cycle cover in $\mathcal{L}$ without extra self-loops is

$$N_{\text{cycle}} = N - W, \tag{B1}$$

where W is the sum of the weights of the links in the minimum weight perfect matching.

2. Longest control path not involving cycles ($l_{\max}$)

In this section we provide the algorithm to measure the longest control path not involving cycles $l_{\max}$ of layer II of a two-layer network for the case $\tau_{\text{I}} \geqslant 1$ and $\tau_{\text{II}} = 1$. The algorithm itself serves as the precise definition of $l_{\max}$.

Given a two-layer directed network $\mathcal{M}$, let N_{cycle} be the maximum number of nodes that can be covered by node disjoint cycles in layer II. To measure $l_{\max}$, first we construct the dynamical graph $\mathcal{D}_l^{\text{II}}$ representing the time evolution of the layer II between time $t = 0$ and $t = l$ as if it would be

isolated as defined in Sec. III A. We search for disjoint control paths connecting nodes at time step $t = 0$ with nodes at time step $t = l$; e.g., each control path connects a node $v_{i,0}^{\text{II}}$ with $v_{j,l}^{\text{II}}$. The maximum number of such paths $N_{\text{path}}(l)$ provides the maximum number of internally controlled nodes if $\tau_{\text{I}} = l$. To determine $N_{\text{path}}(l)$ we convert the problem to a maximum flow problem: We set the capacity of each link and each node in $\mathcal{D}_l^{\text{II}}$ to 1. We then find the maximum flow connecting source node set $V_0^{\text{II}} = \{v_{i,0}^{\text{II}} | i = 1, 2, \ldots, N\}$ to target node set $V_l^{\text{II}} = \{v_{i,l}^{\text{II}} | i = 1, 2, \ldots, N\}$ using a maximum flow algorithm of choice. The maximum flow provides $N_{\text{path}}(l)$. Finally, $l_{\max}$ is defined as one less than the smallest value of l such that

$$N_{\text{path}}(l) = N_{\text{cycle}}. \tag{B2}$$

Figures 7 and 8 provide two examples to illustrate the calculation of $l_{\max}$.
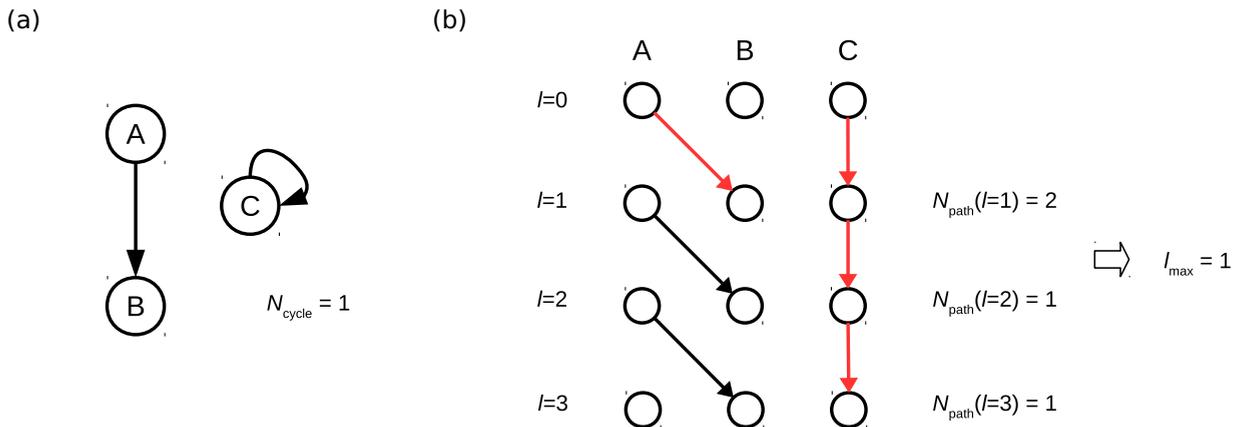
FIG. 8. $l_{\max}$, example 2. (a) A directed network containing a cycle. The size of the maximum cycle cover is $N_{\text{cycle}} = 1$. (b) We count the maximum number of disjoint control paths $N_{\text{path}}(l)$ which connect nodes at time step 0 with nodes at time step l. We find that $l' = 2$ is the smallest value of l such that $N_{\text{path}}(l) = N_{\text{cycle}} = 1$; therefore, $l_{\max} = 1$. $N_{\text{path}}(l)$ remains nonzero for $l > l_{\max}$, showing that cycles can support control paths of any length.

[1] Réka Albert and Albert-László Barabási, Statistical mechanics of complex networks, Rev. Mod. Phys. **74**, 47 (2002).

[2] Mark E. J. Newman, The structure and function of complex networks, SIAM Rev. **45**, 167 (2003).

[3] Mikko Kivelä, Alex Arenas, Marc Barthelemy, James P. Gleeson, Yamir Moreno, and Mason A Porter, Multilayer networks, J. Complex Networks **2**, 203 (2014).

[4] Stefano Boccaletti, G. Bianconi, R. Criado, Charo I. Del Genio, J. Gómez-Gardeñes, M. Romance, I. Sendina-Nadal, Z. Wang, and M. Zanin, The structure and dynamics of multilayer networks, Phys. Rep. **544**, 1 (2014).

[5] Sergey V. Buldyrev, Roni Parshani, Gerald Paul, H. Eugene Stanley, and Shlomo Havlin, Catastrophic cascade of failures in interdependent networks, Nature (London) **464**, 1025 (2010).

[6] Charles D. Brummitt, Raissa M. D'Souza, and E. A. Leicht, Suppressing cascades of load in interdependent networks, Proc. Natl. Acad. Sci. U.S.A. **109**, E680 (2012).

[7] Sergio Gomez, Albert Diaz-Guilera, Jesus Gomez-Gardeñes, Conrad J. Perez-Vicente, Yamir Moreno, and Alex Arenas, Diffusion Dynamics on Multiplex Networks, Phys. Rev. Lett. **110**, 028701 (2013).

[8] Xiao Fan Wang and Guanrong Chen, Pinning control of scale-free dynamical networks, Physica A (Amsterdam, Neth.) **310**, 521 (2002).

[9] Francesco Sorrentino, Mario di Bernardo, Franco Garofalo, and Guanrong Chen, Controllability of complex networks via pinning, Phys. Rev. E **75**, 046103 (2007).

[10] Yang-Yu Liu, Jean-Jacques Slotine, and Albert-László Barabási, Controllability of complex networks, Nature (London) **473**, 167 (2011).

[11] Wen-Xu Wang, Xuan Ni, Ying-Cheng Lai, and Celso Grebogi, Optimizing controllability of complex networks by minimum structural perturbations, Phys. Rev. E **85**, 026115 (2012).

[12] Zhengzhong Yuan, Chen Zhao, Zengru Di, Wen-Xu Wang, and Ying-Cheng Lai, Exact controllability of complex networks, Nat. Commun. **4**, 2447 (2013).

[13] Sean P. Cornelius, William L. Kath, and Adilson E. Motter, Realistic control of network dynamics, Nat. Commun. **4**, 1942 (2013).

[14] Márton Pósfai, Yang-Yu Liu, Jean-Jacques Slotine, and Albert-László Barabási, Effect of correlations on network controllability, Sci. Rep. **3**, 1067 (2013).

[15] Jianxi Gao, Yang-Yu Liu, Raissa M. D'Souza, and Albert-László Barabási, Target control of complex networks, Nat. Commun. **5**, 5415 (2014).

[16] Francesco Lo Iudice, Franco Garofalo, and Francesco Sorrentino, Structural permeability of complex networks to control signals, Nat. Commun. **6**, 8349 (2015).

[17] Airlie Chapman, Marzieh Nabi-Abdolyousefi, and Mehran Mesbahi, Controllability and observability of network-of-networks via cartesian products, IEEE Trans. Autom. Control **59**, 2668 (2014).

[18] Giulia Menichetti, Luca Dall'Asta, and Ginestra Bianconi, Control of multilayer networks, Sci. Rep. **6**, 20706 (2016).

[19] Zhengzhong Yuan, Chen Zhao, Wen-Xu Wang, Zengru Di, and Ying-Cheng Lai, Exact controllability of multiplex networks, New J. Phys. **16**, 103036 (2014).

[20] Yan Zhang, Antonios Garas, and Frank Schweitzer, Value of peripheral nodes in controlling multilayer scale-free networks, Phys. Rev. E **93**, 012309 (2016).

[21] Srilata Zaheer, Stuart Albert, and Akbar Zaheer, Time scales and organizational theory, Acad. Manage. Rev. **24**, 725 (1999).

[22] Bruce Alberts, Alexander Johnson, Julian Lewis, Martin Raff, Keith Roberts, and Peter Walter, *Molecular Biology of the Cell*, 4th ed. (Garland Science, New York, 2002).

[23] K. Lerman and R. Ghosh, Information contagion: An empirical study of the spread of news on Digg and Twitter social networks, in *Proceedings of the 4th International Conference on Weblogs and Social Media (ICWSM)* (AAAI, Menlo Park, CA, 2010).

[24] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon, What is twitter, a social network or a news media? in *Proceedings of the 19th International Conference on World Wide Web* (ACM, New York, 2010), p. 591.

[25] Eytan Bakshy, Itamar Rosenn, Cameron Marlow, and Lada Adamic, The role of social networks in information diffusion, in *Proceedings of the 21st International Conference on World Wide Web* (ACM, New York, 2012), p. 519.

[26] Bitly Science Team, You just shared a link. How long will people pay attention? http://blog.bitly.com/post/9887686919/you-just-shared-a-link-how-long-will-people-pay (2011) [online; accessed November 16, 2015].

[27] Ailing Qia, Xuewei Jua, Qing Zhanga, and Zengqiang Chenb, Structural controllability of discrete-time linear control systems with time-delay: A delay node inserting approach, Math. Probl. Eng. **2016**, 1429164 (2016).

[28] R. E. Kalman, Contributions to the theory of optimal control, Bol. Soc. Mat. Mex. **5**, 102 (1960).

[29] Ching Tai Lin, Structural controllability, IEEE Trans. Autom. Control **19**, 201 (1974).

[30] Kazuo Murota, Systems analysis by graphs and matroids, in *Algorithms and Combinatorics* (Springer-Verlag, Berlin, 1987), Vol. 3.

[31] René Pfitzner, Ingo Scholtes, Antonios Garas, Claudio J. Tessone, and Frank Schweitzer, Betweenness Preference: Quantifying Correlations in the Topological Dynamics of Temporal Networks, Phys. Rev. Lett. **110**, 198701 (2013).

[32] Márton Pósfai and Philipp Hövel, Structural controllability of temporal networks, New J. Phys. **16**, 123055 (2014).

[33] Christian Commault and Jean-Michel Dion, Input addition and leader selection for the controllability of graph-based systems, Automatica **49**, 3322 (2013).

[34] G. Csárdi and T. Nepusz, InterJournal **18**, 1695 (2006), http://igraph.org.

[35] Google Optimization Tools, https://developers.google.com/optimization/.

[36] K.-I. Goh, B. Kahng, and D. Kim, Universal Behavior of Load Distribution in Scale-Free Networks, Phys. Rev. Lett. **87**, 278701 (2001).

[37] John E. Hopcroft and Richard M. Karp, An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs, SIAM J. Comput. **2**, 225 (1973).

[38] Lenka Zdeborová and Marc Mézard, The number of matchings in random graphs, J. Stat. Mech.: Theory Exp. (2006) P05003.

[39] Tao Jia, Yang-Yu Liu, Endre Csóka, Márton Pósfai, Jean-Jacques Slotine, and Albert-László Barabási, Emergence of bimodality in controlling complex networks, Nat. Commun. **4**, 2002 (2013).

[40] Tao Jia and Márton Pósfai, Connecting core percolation and controllability of complex networks, Sci. Rep. **4**, 5379 (2014).

[41] Márton Pósfai, Structure and controllability of complex networks, Ph.D. thesis, Eötvös Loránd University, Budapest, Hungary, 2014.

[42] Rudolf Emil Kalman, Mathematical description of linear dynamical systems, J. Soc. Ind. Appl. Math, Ser. A **1**, 152 (1963).

[43] N. Schwartz, R. Cohen, D. Ben-Avraham, A-L. Barabási, and S. Havlin, Percolation in directed scale-free networks, Phys. Rev. E **66**, 015104 (2002).

[44] Asaf Nachmias and Yuval Peres, Critical random graphs: Diameter and mixing time, Ann. Probab. **36**, 1267 (2008).

[45] M. Bauer and O. Golinelli, Core percolation in random graphs: A critical phenomena analysis, Eur. Phys. J. B **24**, 339 (2001).

[46] L. Correale, M. Leone, A. Pagnani, M. Weigt, and Riccardo Zecchina, Core Percolation and Onset of Complexity in Boolean Networks, Phys. Rev. Lett. **96**, 018101 (2006).

[47] Yang-Yu Liu, Endre Csóka, Haijun Zhou, and Márton Pósfai, Core Percolation on Complex Networks, Phys. Rev. Lett. **109**, 205703 (2012).