Transittability of complex networks and its applications to regulatory biomolecular networks

Fang-Xiang Wu[1,2,*], Lin Wu[1], Jianxin Wang[3], Juan Liu[4], & Luonan Chen[5,6,*]

Supplementary Information

Contents

I. Linear dynamic systems and complete controllability

A. Linear dynamic system

Although most complex dynamic systems are nonlinear, the controllability of nonlinear systems is in many aspects structurally similar to that of linear systems (7, 8, 36, 37). Actually, to ultimately develop the control strategies for complex nonlinear networks, a necessary and fundamental step is to investigate the controllability (especially structural controllability) of complex networks with linear dynamics (37). A network of n nodes with linear dynamics can be described by the following equation

$$\dot{x}(t) = Ax(t) + Bu(t) \qquad \text{(S1)}$$

where the n-dimensional vector $x(t) = (x_1(t), \ldots, x_n(t))^T$ represents the state of the network with n nodes at time t. The $n \times n$ matrix $A=(a_{ij})$ describes the interaction relationship and strength between nodes, $a_{ij} \neq 0$ if node j directly affects node i, that is, there is an arc from node j to node i in the network, and otherwise $a_{ij} = 0$. The $n \times p$ matrix B is called the input control matrix that identifies the nodes on which the input control signals are directly acted. The p-dimensional vector $u(t) = (u_1(t), \ldots, u_p(t))^T$ represents the input control signals.

In general, the perturbation of one node in a complex network could affect other nodes and thus may change the state of the whole network. Now we are interested in the following question: given two specific states, by perturbing which nodes can the system (S1) be transited from one state to another in a finite time period? We call a node on which an input control signal is directly acted as a steering node. Furthermore we are interested in the following question: given two specific states, can we find a minimum set of steering nodes via which the system (S1) can be transited from one state to another in finite time? This is a problem of controllability of linear systems in the area of control theory.

Given an initial state x_0, the system (S1) has the following unique solution

$$x(t) = \exp(At)x_0 + \exp(At)\int_0^t \exp(-As)Bu(s)ds \qquad \text{(S2)}$$

where $\exp(At) = \sum_{i=0}^{\infty} \frac{(At)^i}{i!}$.

B. Complete controllability

In control theory, the system (S1) is said to be completely controllable if it can be steered from any initial state to any desired final state in finite time t_f. There are many equivalent sufficient and necessary conditions (28) to verify the complete controllability of linear system (S1). One of the easiest and widely used conditions is that the system (S1) is completely controllable if and only if the n×np Kalman's controllability matrix

$$C = [B, AB, \ldots, A^{n-1}B] \qquad \text{(S3)}$$

has the full rank, that is

$$rank(C) = n \qquad \text{(S4)}$$

Remark 1.1: When n is large, some elements of A^n would be very large (or very small) if the spectrum of A is greater (or less) than one, which is common in practice. Therefore, matrix C becomes ill-defined and rank (C) cannot be accurately calculated. In addition, in practice the entries of matrix A are unknown at all or not exactly known, which makes the result further inaccurate and intractable.

Remark 1.2: To address the points in Remark 1, recently Liu, et al (7) have studied the structural controllability of system (S1) in terms of complete controllability. Based on (S3) and (S4), they have concluded that the minimum number of input control signals is about 80% of nodes in a regulatory biomolecular network in order to have complete controllability(7), which seems to contradict some empirical findings in cellular reprogramming field(32). Actually, a stable phenotype (state) can be considered a high-dimensional attractor of the regulatory biomolecular network (34). The transitions between two stable phenotypes stem from the change of states of a portion of all nodes in a subspace of full state space (35). Therefore, when concerning the transition between two specific states, it is not necessary to have complete controllability.

Remark 1.3: From (S2), the complete controllability only depends on matrix pair (A, B) while it is independent of the initial state, the final state, input control signals and the steering time t_f. However, if system (S1) is completely controllable, the amount of input control signal u(t) and the steering time t_f are interdependent.

C. The minimum set of driver nodes for complete controllability

Liu, et al (7) have proved that the following minimum input theorem.

Minimum Input Theorem: The minimum number of inputs (N_I) or equivalently the minimum number of driver nodes (N_D) needed to fully control a network G(A) is one if there is a perfect matching in G(A) (In this case, any single node can be chosen as the driver node.) Otherwise, it equals the number of unmatched nodes with respect to any maximum matchings (in this case, the driver nodes are just the unmatched nodes.)

$$N_I = N_D = \max\{N-|M^*|,1\}$$

All concepts and notations here are the same as in the paper (7). Based on this minimum input theorem, they have studied controllability of 37 complex networks and calculated the minimum number of driver nodes to fully control each of these networks. Later on, Nepusz and Vicsek (8) have applied this minimum input theorem to the so-called switchboard dynamics of complex networks and calculated the minimum number of driver nodes to fully control each of 38 real networks in the paper (8).

However, in the following we illustrate that the perturbation of driver nodes identified by the minimum input theorem cannot completely control a network via a couple of counterexamples.

Counterexample 1: Consider the following simple network G(A) with 5 nodes and 6 edges as shown in Figure S1 a). In this network, one can easily find a perfect matching which consists of arcs (v_1,v_3), (v_3,v_1), (v_5,v_4), (v_4,v_2), and (v_2,v_5). According to the minimum input theorem (7), any node can be chosen as a driver node. However, if one choose either node v_1 or v_3 as the driver node, this network cannot be completely controlled by only applying input control signal to it (typically perturbing it) according to structural controllability theory (30-32, 48-51). Nevertheless, choosing any single node of v_2, v_4, or v_5, this network can be completely controlled.

Counterexample 2: Consider the following network G(A) with 6 nodes and 7 edges as shown in Figure S1 b). For this network, we can find the size of the maximum matching is five, that is, $M^*=5$, and v_6 is the unmatched node. According to the minimum input theorem (7), v_6 is the only driver node. However, if one chooses the node v_6 as the driver node, this system cannot be completely controlled by only applying input control signal to it according to structural controllability theory (30-32, 48-51).

From structural controllability theory (30-32, 48-51), a linear control system (A, B) is structurally completely controllable if and only if i) the digraph G(A,B) contains no inaccessible nodes and ii) the digraph G(A,B) contains no dilation. Actually the minimum input theorem developed by Liu, et al (7) can only guarantee that the condition ii) is true while the condition i) has been mistreated there. In the proof of their minimum input theorem, Liu, et al (7) treated condition i) by adding some extra links, which means superficially adding some non-existing regulatory relationships between driver nodes and other nodes in a biomolecular network. For example, in a biomolecular network like counterexample 1, the node v_1, or v_3 can be chosen as the driver node by minimum input theorem (7). If applying input control signal, which is generally the perturbation of molecular species (32) represented by v_1 or v_3, one cannot expect that such an input control signal can affect node v_2, v_4, or v_5 as there are not arcs from the chosen node v_1 or v_3 to them as either v_1 or v_3 does not regulate any node v_2, v_4, or v_5. The similar discussion can be made to counterexample 2. Therefore, the minimum input theorem developed by Liu, et al (7) only provides a necessary condition to completely control a network. In order to determine complete controllability, one should check if all nodes in the network can be accessed from the driver nodes identified by the minimum input theorem developed in (7).

Liu, et al (7) assume that one controller should output multiple input control signals, which is not the case in many situations (10, 21-26). As illustrated in counterexamples 1 and 2, a network cannot be guaranteed to have complete controllability by perturbing only the minimum set of driver nodes identified by the minimum input theorem (7). To completely control a network, the number of input control signals may actually be greater than the minimum number of driver nodes identified by the minimum input theorem (7).

II. Transittability

A. Basic concepts

Actually, in practice we do not need to steer system (S1) from any initial state to any desired final state, but to steer system (S1) from one specific state to another specific state. Therefore, we adopt the transittability of system (S1) in this study which concerns the possibility that two given specific states can be transited with a suitable choice of input control signals in finite time t_f. Mathematically, for two given specific states x_0 and x_1, the system (S1) is said to be transittable between x_0 and x_1 if it can be steered from the initial state x_0 to the desired final state x_1 with a suitable choice of input control signals in finite time t_f. As system (S1) is time-invariant, if it can be steered from the initial state x_0 to the desired final state x_1 in finite time, it can also be steered from the initial state x_1 to the desired final state x_0 with a suitable choice of input control signals in finite time.

To prove our theorems in next section, we need the following concepts and lemmas.

Let A be an n×n matrix and f(λ) be the characteristic polynomial of A, that is, $f(\lambda) = \det(\lambda I - A)$, which is an n-th degree polynomial. By Cayley-Hamilton theorem (28), f(A)=0. The minimal polynomial of A is a polynomial g(λ) with the smallest degree such that g(A)=0. Let d be the degree of the minimal polynomial of an n×n matrix A, then we have d≤n.

Lemma 2.1: Let d be the degree of minimal polynomial of an n×n matrix A. Then we have

$$\exp(At) = \sum_{i=1}^{d} \alpha_i(t) A^{i-1} \qquad \text{(S5)}$$

and scalar functions $\{\alpha_i(t), i = 1, \ldots, d\}$ are linearly independent on any time interval [0, t_f].

Proof: From the definitions of exp(At) and the minimal polynomial of A, it can be easily proved that (S5) is true. In the following, we prove that scalar functions $\{\alpha_i(t), i = 1,\ldots,d\}$ are linearly independent on any time interval [0, t_f]. Let $g(\lambda) = \lambda^d + a_d\lambda^{d-1} + \cdots a_2\lambda + a_1$ be the minimal polynomial of A, then we have

$$g(A) = A^d + a_d A^{d-1} + \cdots + a_2 A + a_1 I = 0$$

That is

$$A^d = -a_d A^{d-1} - \cdots - a_2 A - a_1 I \qquad \text{(S6)}$$

From the definitions of exp(At) and (S6), we have

$$\frac{d}{dt}\exp(At) = \sum_{i=1}^{d}\dot{\alpha}_i(t)A^{i-1} = A\exp(At) = \sum_{i=1}^{d}\alpha_i(t)A^i \qquad \text{(S7)}$$

And

$$\sum_{i=1}^{d}\alpha_i(0)A^{i-1} = I \qquad \text{(S8)}$$

From (S7) and (S6), we have

$$(\dot{\alpha}_1(t) + a_1\alpha_d(t))I + (\dot{\alpha}_2(t) - \alpha_1(t) + a_2\alpha_d(t))A + \cdots + (\dot{\alpha}_d(t) - \alpha_{d-1}(t) + a_d\alpha_d(t))A^{d-1} = 0$$

as the degree of the minimal polynomial of A is d, the above equation indicates

$$\begin{aligned} &\dot{\alpha}_1(t) + a_1\alpha_d(t) = 0 \\ &\dot{\alpha}_2(t) - \alpha_1(t) + a_2\alpha_d(t) = 0 \\ &\cdots \\ &\dot{\alpha}_d(t) - \alpha_{d-1}(t) + a_d\alpha_d(t) = 0 \end{aligned} \qquad \text{or} \qquad \dot{\alpha}(t) = E\alpha(t) \qquad \text{(S9)}$$

where

$$E = \begin{bmatrix} 0 & 0 & \cdots & 0 & -a_1 \\ 1 & 0 & \cdots & 0 & -a_2 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 1 & -a_d \end{bmatrix}, \quad \alpha(t) = \begin{bmatrix} \alpha_1(t) \\ \alpha_1(t) \\ \vdots \\ \alpha_k(t) \end{bmatrix}$$

From (S8) we have

$$(\alpha_1(0) - 1)I + \alpha_2(0)A + \cdots + \alpha_d(0)A^{d-1} = 0$$

as the degree of the minimal polynomial of A is d, the above equation indicates

$$\alpha(0) = [1,0,\cdots,0]^T \qquad \text{(S10)}$$

Thus scalar functions $\{\alpha_i(t), i=1,\ldots,d\}$ should satisfy linear differential systems (S9) with initial conditions (S10), which has a unique solution. Assume there exist constants c_i (i=1,2,…,d) such that

$$c_1\alpha_1(t) + c_2\alpha_2(t) + \cdots + c_d\alpha_d(t) = 0 \quad \text{or}$$

$$c^T\alpha(t) = 0 \qquad \text{on some time interval } [0,\ t_f] \qquad \text{(S11)}$$

where $c = [c_1, c_2, \ldots, c_d]^T$.

Then taking the k-th derivative on (S11), we have

$$c_1\alpha_1^{(k)}(t) + c_2\alpha_2^{(k)}(t) + \cdots + c_d\alpha_d^{(k)}(t) = 0, \text{ or}$$

$$c^T\alpha^{(k)}(t) = 0 \qquad \text{for k=1, 2,…,d-1, on the time interval } [0,\ t_f] \qquad \text{(S12)}$$

where $\alpha_i^{(k)}(t)$ is the k-th derivative of $\alpha_i(t)$ **.** Combining (S11), (S12) and (S9) yields

$$c^T\alpha^{(k)}(t) = c^T E^k \alpha(t) = 0 \qquad \text{for k=0, 1, 2,…,d-1, on the time interval } [0,\ t_f]$$

and thus

$$c^T\alpha^{(k)}(0) = c^T E^k \alpha(0) = 0 \qquad \text{for k=0, 1, 2,…,d-1}$$

Or

$$c^T[\alpha(0), E\alpha(0), \ldots, E^{d-1}\alpha(0)] = 0$$

which indicates $c_i = 0$(i=1,2,…,d). Therefore scalar functions $\{\alpha_i(t), i=1,\ldots,d\}$ are linearly independent on any time interval [0, t_f].

Lemma 2.2: If scalar functions $\{\alpha_i(t), i=1,\ldots,d\}$ are linearly independent on time interval [0, t_f], then for any p-dimensional column vectors p_1, p_2, …, p_d, a system of the following integral equations

$$\int_0^{t_f} \alpha_i(s)u(s)ds = p_i \quad , \quad \text{i=1,2,…,d} \qquad \text{(S13)}$$

has a solution $u(t) \in R^p$ for $t \in [0, t_f]$

Proof: As the group of scalar functions $\{\alpha_i(t), i=1,\ldots,d\}$ are linearly independent on any time interval [0, t_f], applying Gram-Schmidt process generating a group of orthonormal scalar functions $\{\beta_i(t), i=1,\ldots,d\}$ on [0, t_f]. The two groups of functions are related by

$$\begin{bmatrix} \alpha_1(t) \\ \alpha_2(t) \\ \vdots \\ \alpha_d(t) \end{bmatrix} = \begin{bmatrix} g_{11} & 0 & \cdots & 0 \\ g_{21} & g_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{d1} & g_{d2} & \cdots & g_{dd} \end{bmatrix} \begin{bmatrix} \beta_1(t) \\ \beta_2(t) \\ \vdots \\ \beta_d(t) \end{bmatrix}$$

where $g_{ii}>0$. From the above equation, we have

$$\alpha_i(t) = \sum_{k=1}^{i} g_{ik}\beta_k(t) \qquad \text{(S14)}$$

Construct the solution of (S13) as follows

$$u(t) = \sum_{l=1}^{d} f_l \beta_l(t)\,, \qquad \text{(S15)}$$

where f_l (l=1,2,…,d) are p-dimensional constant column vectors to be determined

Substituting equations (S14) and (S15) into (S13) yields

$$p_i = \int_0^{t_f} \sum_{k=1}^{i} g_{ik}\beta_k(t) \sum_{l=1}^{d} f_l \beta_l(t) ds = \sum_{k=1}^{i} g_{ik} \sum_{l=1}^{d} f_l \delta_{kl} = \sum_{k=1}^{i} g_{ik} f_k \quad \text{for } i=1,2,\ldots,d \qquad \text{(S16)}$$

From (S16), we can recursively solve for f_l (l=1, 2, …, d) as follows

As i = 1, equation (S16) becomes

$$p_1 = g_{11} f_1$$

Thus we have $f_1 = p_1 / g_{11}$

As i = 2, equation (S16) becomes $p_2 = \sum_{k=1}^{2} g_{2k} f_k = g_{21} f_1 + g_{22} f_2$

Thus we have $f_2 = (p_2 - g_{21} f_1) / g_{22}$

……

In summary, we have

$f_i = (p_i - \sum_{k=1}^{i-1} g_{ik} f_k) / g_{ii}$ for i>1 with $f_1 = p_1 / g_{11}$

B. Transittability Theorems

In this section, we present several theorems about the transittability of system (S1) and their proofs.

Theorem 2.3: The following statements are equivalent

1) There exists input control signal u(t), $t \in [0, t_f]$ by which the system (S1) can be transited between two specific states x_0 and x_1

2) $x_1 - \exp(At_f)x_0 \in \mathrm{Im}(\Phi_f)$, where $\Phi_f = \int_0^{t_f} \exp(A(t_f - s))BB^T \exp(A^T(t_f - s))ds$

3) $x_1 - \exp(At_f)x_0 \in span\{C\}, C = [B, AB, \ldots, A^{n-1}B]$

Proof of 1) ⇔2): Sufficiency: from 2) to 1). If $x_1 - \exp(At_f)x_0 \in \mathrm{Im}(\Phi_f)$, then there exists an n-dimensional vector ξ such that $x_1 - \exp(At_f)x_0 = \Phi_f \xi$. Let $u(t) = B^T \exp(A^T(t_f - s))\xi$, then at $t = t_f$, from (S2) we have

$$
\begin{aligned}
x(t_f) &= \exp(At_f)x_0 + \int_0^{t_f} \exp(A(t_f - s))Bu(s)ds \\
&= \exp(At_f)x_0 + \int_0^{t_f} \exp(A(t_f - s))BB^T \exp(A^T(t_f - s))ds\xi \\
&= \exp(At_f)x_0 + \Phi_f \xi = x_1
\end{aligned}
$$

Necessity: from 1) to 2). Suppose that 1) is true. For the sake of a contradiction, assume that $x_1 - \exp(At_f)x_0 \notin \mathrm{Im}(\Phi_f)$. Then there exist vectors $\mu \in \mathrm{Im}(\Phi_f)$ and $0 \neq \nu \in Ker(\Phi_f)$ such that $x_1 - \exp(At_f)x_0 = \mu + \nu$.

$\nu \in Ker(\Phi_f)$ indicates that

$$
0 = \nu^T \Phi_f \nu = \int_0^{t_f} \nu^T \exp(A(t_f - t))BB^T \exp(A^T(t_f - t))\nu dt = \int_0^{t_f} \left\| \nu^T \exp(A(t_f - t))B \right\|^2 dt
$$

and hence

$\nu^T \exp(A(t_f - t))B = 0$, for any $t \in [0, t_f]$

However, as 1) is true, there exists input control signal u(t), $t \in [0, t_f]$ and we have

$x_1 = \exp(At_f)x_0 + \exp(At_f)\int_0^{t_f} \exp(-As)Bu(s)ds$ or

$$\mu + \nu = x_1 - \exp(At_f)x_0 = \int_0^{t_f} \exp(A(t_f - s)Bu(s)ds$$

Multiplying by ν^T yields

$$\nu^T \mu + \nu^T \nu = \int_0^{t_f} \nu^T (A(t_f - s)Bu(s)ds = 0$$

As $\nu^T \mu = 0$, we have $\nu^T \nu = 0$, that is, $\nu = 0$, which is a contradiction.

Proof of 1) ⇔3): Necessity: from 1) to 3). If 1) is true, there exists an input control u(t), $t \in [0, t_f]$ such that

$$x_1 = \exp(At_f)x_0 + \exp(At_f)\int_0^{t_f} \exp(-As)Bu(s)ds \text{ or}$$

$$x_1 - \exp(At_f)x_0 = \int_0^{t_f} \exp(A(t_f - s))Bu(s)dsx_0 = \int_0^{t_f} \sum_{i=1}^{n} \alpha_i (t_f - s)A^{i-1}Bu(s)ds$$
$$= \sum_{i=1}^{n} A^{i-1}B\int_0^{t_f} \alpha_i(t_f - s)u(s)ds \in span\{B, AB, \ldots, A^{n-1}B\}$$

Sufficiency: from 3) to 1). Let d be the degree of the minimal polynomial of matrix A, then we can have

$$span[B, AB, \ldots, A^{n-1}B] = span[B, AB, \ldots, A^{d-1}B]$$

If 3) is true, there exist d p-dimensional column vectors p_i (i=1,…,d) such that

$$x_1 - \exp(At_f)x_0 = \sum_{i=1}^{d} (A)^{i-1}Bp_i \text{, for i=1,…,d}$$

From Lemmas 2.2 and 2.1, for d p-dimensional column vectors p_i (i=1,…,d) , there exists input control signal u(t) such that (S13) is true. Therefore, we have

$$x_1 - \exp(At_f)x_0 = \sum_{i=1}^{d} A^{i-1}Bp_i = \sum_{i=1}^{d} A^{i-1}B\int_0^{t_f} \alpha_i(s)u(s)ds = \int_0^{t_f} \exp(A(t_f - s))Bu(s)ds$$

which indicates that 1) is true.

Theorem 2.4: Consider two specific states x_0 and x_1 of the system (S1) with $x_1 \in span\{C\}$. The following statements are equivalent

1) The system (S1) can be transited between states x_0 and x_1

2) $rank[x_0 - x_1, B, AB, \ldots, A^{n-1}B] = rank(C)$

3) $rank(C) = rank(\overline{C})$, where $\overline{C} = [\overline{B}, A\overline{B}, \ldots, A^{n-1}\overline{B}]$ and $\overline{B} = [x_0 - x_1, B]$ (S17)

Proof: Note that

$$A^k span\{C\} \subseteq span\{C\}, \text{ for k=0, 1, 2,…} \quad \text{(S18)}$$

Thus for any value t_f, by Lemma 2.1 we have

$$\exp(At_f) span\{C\} \subseteq span\{C\} \quad \text{(S19)}$$

Proof of 1) ⇔2): Sufficiency: from 2) to 1). If $rank[x_0 - x_1, B, AB, \ldots, A^{n-1}B] = rank(C)$, then we have $x_0 - x_1 \in span\{C\}$ and thus $x_0 - x_1 + x_1 = x_0 \in span\{C\}$ as $x_1 \in span\{C\}$. Furthermore, we have $\exp(At_f)x_0 \in span\{C\}$ from (S19). Therefore, we have $x_1 - \exp(At_f)x_0 \in span\{C\}$, from 3) of Theorem 2.3, the system (S1) can be transited between states x_0 and x_1.

Necessity: from 2) to 1). If 1) is true, we have $x_1 \in span\{C\}$ and $x_1 - \exp(At_f)x_0 \in span\{C\}$ from 3) of Theorem 2.3. Furthermore, from (S19), we have $\exp(-At_f)x_1 - x_0 \in \exp(-At_f)span\{C\} \subseteq span\{C\}$ and $\exp(-At_f)x_1 \in span\{C\}$. Therefore, we obtain $x_0 \in span\{C\}$. Combining with $x_1 \in span\{C\}$, we have $rank[x_0 - x_1, B, AB, \ldots, A^{n-1}B] = rank(C)$.

Proof of 2) ⇔3): Sufficiency: from 3) to 2). Let $\eta_1, \eta_2, \ldots, \eta_s$ be the basis of $span\{C\}$. If 3) is true, $\eta_1, \eta_2, \ldots, \eta_s$ is a basis of $span\{[\overline{B}, A\overline{B}, \ldots, A^{n-1}\overline{B}]\}$ too. Therefore, there exist s constants c_i (i=1,2..,s) such that

$$x_0 - x_1 = c_1\eta_1 + c_2\eta_2 + \cdots + c_s\eta_s \quad \text{(S20)}$$

Therefore, $span\{C\} = span\{[x_0 - x_1, B, AB, \ldots, A^{n-1}B]\}$ and thus $rank[x_0 - x_1, B, AB, \ldots, A^{n-1}B] = rank(C)$.

Necessity: from 2) to 3). Note that matrix $[\overline{B}, A\overline{B}, \ldots, A^{n-1}\overline{B}]$ has more column vectors $A(x_0 - x_1), \ldots, A^{n-1}(x_0 - x_1)$ than matrix $[x_0 - x_1, B, AB, \ldots, A^{n-1}B]$. If 2) is true, there exist s constants c_i (i=1,2..,s) such that (S20) is true. Then we have

$A^k(x_0 - x_1) = c_1 A^k \eta_1 + c_2 A^k \eta_2 + \cdots + c_s A^k \eta_s$, for k=1,2,..,n-1 (S21)

By (S18), $A^k \eta_i \in span(C)$ (k=1,..,n-1, i=1,..,r). Therefore, $A^k(x_0 - x_1) \in span(C)$ for k=1,2,..,n-1, from which, we can conclude that 3) is true.

Similarly, we can have the following theorem.

Theorem 2.5: Consider two specific states x_0 and x_1 of the system (S1) with $x_0 \in span\{C\}$. The following statements are equivalent

1) The system (S1) can be transited between states x_0 and x_1
2) $rank[x_0 - x_1, B, AB, \ldots, A^{n-1}B] = rank(C)$
3) $rank(C) = rank(\overline{C})$, where $\overline{C} = [\overline{B}, A\overline{B}, \ldots, A^{n-1}\overline{B}]$ and $\overline{B} = [x_0 - x_1, B]$ (S17)

From Theorems 2.4 and 2.5, we can have the following conclusion.

Theorem 2.6: With either specific state x_0 or $x_1 \in span\{C\}$, the system (S1) can be transited between states x_0 and x_1 if and only if

$$rank(C) = rank(\overline{C}) \text{ , where } \overline{C} = [\overline{B}, A\overline{B}, \ldots, A^{n-1}\overline{B}] \text{ and } \overline{B} = [x_0 - x_1, B] \quad \text{(S17)}$$

For a given system (S1) and two specific states x_0 and x_1, assume that we have found a set of steering nodes (thus input control matrix B) by which the system can be transited between two states x_0 and x_1. The question may be raised: if more steering nodes are added, could the system be transited between two states x_0 and x_1 by this larger set of steering nodes? This question can be answered "Yes" by the following theorem. To state the theorem, we call B_e the extended matrix of an n×r matrix B if it is constructed by appending one or more n-dimensional vectors to matrix B.

Theorem 2.7: If the system (S1) can be transited between two states x_0 and x_1 with control matrix B, so can it with control matrix B_e.

Proof: If the system (S1) can be transited between two states x_0 and x_1 with control matrix B, by 3) in Theorem 2.3, we have

$$x_1 - \exp(At_f)x_0 \in span\{C\}, C = [B, AB, \ldots, A^{n-1}B]$$

On the other hand, B_e is the extended matrix of B, we have

$$span\{C\} \subset span\{[B_e, AB_e, \ldots, A^{n-1}B_e]\}$$

Therefore, we obtain

$$x_1 - \exp(At_f)x_0 \in span\{[B_e, AB_e, \ldots, A^{n-1}B_e]\}$$

which means that the system (S1) can be transited between two states x_0 and x_1 with control matrix B_e by 3) in Theorem 2.3.

If either state x_0 or x_1 is the equilibrium state of a complex network (which is usually true for the problems of interests), we can always shift the state x such that the equilibrium state is at the origin. Here without loss of generality assume the state x_1 is at the origin, i.e., x_1=0. From Theorem 2.6, we have the following result.

Theorem 2.8: The system (S1) is transittable between a specific state x_0 and the origin if and only if

$$rank(C) = rank(C_0) \quad \text{where } C_0 = [B_0, AB_0, \ldots, A^{n-1}B_0] \text{ and } B_0 = [x_0, B] \qquad \text{(S22)}$$

Remark 2.1: From Theorems 2.5, 2.6 and 2.8, in order to determine the transittability we need to calculate the ranks of matrices C and either C_0 or $\overline{C}$ (S17, S22). Form their definitions, calculating the rank of matrix C_0 or $\overline{C}$ is similar to that of C. Again, when n is large, the elements of A^n would be very large (or very small) and thus matrices C, C_0 and $\overline{C}$ becomes ill-defined. As a result, the rank of these matrices cannot be accurately calculated. In addition, the entries of matrix A and components of vectors x_0 and x_1 are unknown at all or not exactly known in practice, which makes the result further inaccurate and intractable.

In the literature, the value of $rank(C)$ is called the dimension of controllable subspace of system (A, B). Here we adopt this terminology and call the values of $rank(\overline{C})$ and $rank(C_0)$ the dimensions of controllable subspace of systems $(A,\overline{B})$ and (A,B_0).

III. Structural transittability

To solve the problem in calculating the rank of matrix C_0, $\overline{C}$ or C, here we adopt the concepts and structural transittability. To this end, we first introduce structural linear systems in this section. As the structural transittability is studied based on graph theory, we also discuss the graph representation of structural linear systems.

A. Structural linear systems and their graph representation

Matrix M is said to be structural if its entries are either fixed zeros or independent free parameters. $\tilde{M}$ is called admissible (with respect to M) if it can be obtained by fixing the free parameters of M at some specific values. The generic rank of a structural matrix M is defined as the maximum rank that M achieves as a function of its free parameters (38).

System (S1) is called a structural system if A and B in system (S1) are structural matrices, and is denoted by (A, B). Associated with a structural system (A,B), a directed graph (digraph) G(A, B)=(V,E) is defined on the set of nodes $V=V_A \cup V_B$, where $V_A=\{x_1,\ldots, x_n\}:=\{v_1,\ldots, v_n\}$ is the set of state vertices, corresponding to the n states while $V_B=\{u_1,\ldots, u_p\}:=\{v_{n+1},\ldots, v_{n+p}\}$ is the set of input vertices, corresponding to the p inputs, and the set of arcs $E=E_A \cup E_B$, where $E_A= \{(x_j, x_i)= (v_j, v_i) \mid a_{ij}\neq 0\}$ is the set of directed edges between states vertices while $E_B= \{(u_j, x_i)= (v_{n+j}, v_i) \mid b_{ij}\neq 0\}$ is the set of directed edges between input vertices and states vertices. One example is shown in Figure S2. By the above definitions, a structural system (A, B) uniquely determine a digraph G(A, B), vice versa. In a directed graph, an elementary path is a sequence of arcs $\{(v_{i0} \to v_{i1}),(v_{i1} \to v_{i2}),\cdots,(v_{ik-1} \to v_{ik})\}$ where all vertices $\{v_{i0},v_{i1},\cdots,v_{ik}\}$ are different, and when

$v_{i0} = v_{ik}$, it is called an elementary cycle. A stem is an elementary path originating from an input vertex in V_B. A set $U \subset V_A$ of nodes is said to be accessible if for every node $v \in U$ there exists a stem terminating at *v*. In Figure S2, nodes v_i (i=1, 2, 3, 4) are accessible while v_5 is not accessible. We can also define G(A)=(V_A, E_A). Matrix A is the structural matrix of network G(A) because A has the same structure as the transpose of the adjacency matrix of network G(A).

A structural system (A, B) is reducible if there exists a permutation matrix P such that

$$PAP^{-1} = \begin{bmatrix} A_1 & A_3 \\ 0 & A_2 \end{bmatrix}, \ PB = \begin{bmatrix} B_1 \\ 0 \end{bmatrix} \qquad \text{(S23)}$$

with $A_1 \in R^{n_1 \times n_1}$, $A_2 \in R^{n_2 \times n_2}$, and $B_2 \in R^{n_2 \times r}$, $1 \le n_1 \le n$ and $n_1 + n_2 = n$. Otherwise (A, B) is said to be irreducible. If (S23) is true, we also call the augmented matrix [A, B] is reducible, and otherwise irreducible. From (S23), we can see that if the structural system (A, B) is reducible, the nodes corresponding to the last n_2 columns are not accessible. Actually, it has been proved (38, 48-51) that a structural system (A, B) is irreducible if and only if there are no accessible nodes in its associated digraph G(A, B).

The structural system (A, B) is called the structurally (completely) controllable if there exist an admissible system $(\tilde{A}, \tilde{B})$ which is completely controllable (30-32), that is $rank[\tilde{B}, \tilde{A}\tilde{B}, \ldots, \tilde{A}^{n-1}\tilde{B}]$ = n. This concept of structural controllability was first introduced by Lin (30) in 1974 for linear systems with single input. In next few decades, this concept was extended to linear systems with multiple inputs (e.g.,32, 38, 48-51) and be substantially studied. One excellent result is as follows.

Theorem 3.1 (49): The structural system (S1) with (A, B) is structurally (completely) controllable if and only if the augmented matrix [A, B] has generically full rank, i.e., $\text{generic rank}([A, B]) = n$, and is irreducible.

This theorem has been presented in the different statement in several papers (30-32, 38, 48-51). Based on this result, Liu et al (7) have studied the controllability of complex networks and later on studied the control centrality of complex networks (52). As the irreducibility of a structural system can be efficiently determined by some graph-theoretic algorithms (39), the remainder is to determine the generic rank of the augmented matrix [A, B]. Given a structure matrix A, Liu et al.(7) employ a maximum assignment algorithm to find the minimum number of driver nodes and thus structural control matrix B such that the augmented matrix [A, B] has generically full rank.

B. Structural transittability theorems

Similarly to structural controllability, we define the structural transittability as follows: a structural system (S1) with (A,B) is structurally transittable between two specific structural states x_0 and x_1 if there exists an admissible system $(\tilde{A}, \tilde{B})$ (with respect to (A,B)) and admissible states $\tilde{x}_0, \tilde{x}_1$ (with respect to x_0, x_1, respectively) such that the system $(\tilde{A}, \tilde{B})$ is transittable between states $\tilde{x}_0$ and $\tilde{x}_1$.

The dimension (rank of controllability matrix C) of its controllable subspace of structural system (A, B) varies as a function of free parameters in matrices A and B. That is, for different admissible systems $(\tilde{A}, \tilde{B})$, the dimensions of their controllable subspaces might be different. As the maximum rank of matrix C is less than n, the dimension of controllable space of structural system (A, B) can reach a maximum value. We define this maximum value as the generic dimension of the controllable subspace of structure system (A, B) and denote it by GDCS(A,B). The GDCS(A,B) is a generic property (31, 32, 38) in the sense that for almost all admissible systems $(\tilde{A}, \tilde{B})$ (with respect to (A, B)), the dimension of their controllable subspace takes a constant (38) which is the maximum rank of C.

With above concepts, from Theorems 2.6 and 2.8, we can have the following results.

Theorem 3.2: 1) The structure system (A, B) is structurally transittable between two specific structural states x_0 and $x_1$with either belonging to $span\{C\}$ if and only if

$$GDCS(A, \bar{B}) = GDCS(A, B) \qquad \text{(S24)}$$

2) The structure system (A, B) is structurally transittable between a specific structural state x_0 and the origin if and only if

$$GDCS(A,B) = GDCS(A,B_0) \qquad \text{(S25)}$$

From Theorem 3.1, if the structural system (A, B) is irreducible, GDCS(A,B) = n if and only if the augmented matrix [A, B] has generically full rank, i.e. $\text{generic rank}([A,B]) = n$. However, when GDCS(A,B)=r<n, the $\text{generic rank}([A,B])$ is at least r (38), but not equal to r in general. For example, a structural system has the following (A, B)

$$A = \begin{bmatrix} 0 & 0 & a_{13} & a_{14} \\ 0 & 0 & 0 & a_{24} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad , \quad B = \begin{bmatrix} 0 \\ 0 \\ b_3 \\ b_4 \end{bmatrix}$$

We have GDCS (A, B) = 2, but $\text{generic rank}([A,B]) = 3$. Therefore, we cannot use the method in (7) to calculate the value of GDCS(A,B). Actually Hosoe has proved the result in (38).

Theorem 3.3(38): If the structural system (A, B) is irreducible. Then

$$\text{GDCS(A, B)} = \max_{G \in G^*}\{|E(G))|\} \qquad \text{(S26)}$$

where G^* denotes the set of subgraphs of digraph $G(A,B)$ which is defined as

$$G^* = \left\{ \begin{array}{l} G \subset G(A,B) \,|\, G \text{ consists of elemnenary cycles and at most p stems in } G(A,B). \\ \text{The elementary cycles and stems have no node in common.} \end{array} \right\}$$

$|E(G)|$ denotes the number of edges contained in G. The advantage of Equation (S26) is that we can calculate $\max_{G \in G^*}\{|E(G)|\}$ by solving an optimal assignment, providing us an efficient graph-theoretic algorithm to determine the exact value of GDCS(A, B) for any given structural system (A, B).

From Theorem 3.3, we can have the following theorem.

Theorem 3.4: If the structural system (A, B) is reducible. Then

$$\text{GDCS(A, B)} < \max_{G \in G^*}\{|E(G))|\} \qquad \text{(S27)}$$

Proof. As (A, B) is reducible, there is at least one node in some elementary cycle in graph $\hat{G} = \arg\max_{G \in G^*} \{|E(G)|\}$ which is not accessible from nodes in V_B, and thus cannot be controlled.

C. Algorithms for identifying the steering kernel

Given a state x_0 of a network G(A), we would like to find the minimum number of steering nodes such that the network G(A) can be transited between a state x_0 and an equilibrium state (e.g., the origin). A brute-force search requires us to compute $GDCS(A, B)$ and $GDCS(A, B_0)$ for 2^n-1 distinct sets of nodes, each set of nodes corresponding to an input control matrix B. In the following, we propose a graph-theoretic algorithm to find the minimum set of steering nodes for state transitions. From (38), although the optimal assignment is not unique, the weight summation of any optimal assignments is the same.

For a network G(A) and structural state x_0, assume the structure system (A, x_0) is irreducible. Let S be a subset of nodes corresponding to non-zero components in x_0. Let us define a weighted graph G'(A) as follows: 1) associate the weight $w_e = 1$ with every edge e of G(A); 2) add the edge e=v_iv_j and associate the weight $w_e = \varepsilon$ if e=v_iv_j is not in G(A) for $v_j \in S$, where ε is a small positive number and less than 1/n for a network with n nodes ; 3) add the loop e=v_iv_i and associate the weight $w_e = 0$ if e=v_iv_i is not in G(A) for $v_i \notin S$. For simplicity, ε can take the value of 0.001, 0.0001, 0.00001 or the like.

A collection of node disjoint cycles in G'(A) covering all nodes will be called a cycle partition. The weight of a cycle partition is the total weight of all edges of individual cycles of the partition. Note that any collection of node disjoint cycles in G'(A) can be completed to a partition by adding the loops for uncovered vertices. Now we are interested in finding the maximum weight cycle partition of G'(A). To do this, we define a weighted bipartite graph representation $G(A, W) = (V^+, V^-, W)$ of G'(A) as follows: $V^+ = \{r_1, \ldots r_n\}$ is the set of vertices corresponding to the n rows of matrix A while $V_A^- = \{c_1, \ldots c_n\}$ is another set of vertices corresponding to the n columns of matrix A. The weight matrix $W=(w_{ij})_{n\times n}$ is defined as follows:

$$w_{ij} = \begin{cases} 1 & \text{the (i, j) - th entry of matrix A is non - zeros} \\ \varepsilon & \text{the (i, j) - th entry of matrix A is zeros, but node } j \in S \\ 0 & \text{the (i, i) - th entry of matrix A is zeros and node } i \notin S \end{cases}$$

An optimal assignment of $G(A, W) = (V^+, V^-, W)$ corresponds to a maximum weight cycle partition of G'(A), in which each circle has exactly one of three different weights: zero, non-zero integer weight, and decimal weight. As a result, the weight of the maximum weight cycle partition of G'(A) is an integer or a decimal number. Based on the maximum weight cycle partition of G'(A), we can find the steering nodes (thus input control matrix B) and construct the maximum cover of G(A, B) consisting of disjoint elementary circles and stems as follows.

Case I: The weight of the maximum weight cycle partition of G'(A) is an integer. In this case, the maximum weight cycle partition of G'(A) is the maximum weight cycle partition of G(A) too. We can find a strongly connected component (SCC) partition of G(A) efficiently by a graph-theoretic algorithm (38). An SCC is called a source SCC (Figure S4 a)) if there is no directed edge pointing to it. Each source SCC has at least one node belonging to S as (A, x_0) is irreducible. Let node $j \in S$ be in a source SCC, then remove the edge pointing to node j in the circle containing node j and add one control node and an edge from the control node pointing to j (Figure S4 a')). Repeat this process for all source SCCs and construct the control matrix B such that each of its columns corresponds to one control node. As a result, the maximum weight cycle partition of G(A) become a subgraph G_M of G(A, B) consisting of disjoint elementary circles and stems whose number of edges equals to the number of edges of the maximum weight cycle partition.

Case II: The weight of the maximum weight cycle partition of G'(A) is a decimal number. In this case, at least one circle in the maximum weight cycle partition of G'(A) has the decimal weight. Let's consider a circle with the decimal weight (Figure S4 b)) of $k + l * \varepsilon$, where *k* and *l* are integers with $k \geq 0$ and $1 \leq l \leq n$.

By the definition of G'(A) , such a circle contains exactly *l* edges in G'(A), not in G(A) , each of which has the weight of ε and points to one node belonging to S. Remove each edge with the decimal weight which points to the node belonging to S in such a circle and add one control node and an edge from the control node pointing to the node belonging to S (Figure S 4 b'). By this way, each circle with decimal weight of $k + l * \varepsilon$ can be corresponded to a set of *l* stems whose number of edges equals to $k + l$. Apply this process to all circles with decimal weight and construct the control matrix B_1 such that each of its columns corresponds to one control node. As a result, all circles with the decimal weight with in the maximum weight cycle partition of G'(A) are induced to a set of disjoint stems. Let G_1 be a graph consisting of disjoint stems induced by all circles with the decimal weight and all circles accessible from the control nodes corresponding to control matrix B_1 in the maximum weight cycle partition of G'(A). Let $G(A_1)$ be the subnetwork of G(A) which is induced by all nodes in G(A) accessible from the control nodes corresponding to control matrix B_1. Let $G(A_2)$ =G (A) – $G(A_1)$. If it is a non-empty network, then $G(A_2)$ has the following properties: 1) All nodes in $G(A_2)$ cannot be affected by any control nodes corresponding to control matrix B_1; 2) $G(A_2)$ is covered by a subset of circles with integer weight in the maximum weight cycle partition of G'(A), which is also the maximum weight cycle partition of $G(A_2)$. Following the procedure in Case I, we can have a set of control nodes and a corresponding control matrix B_2 such that the maximum weight cycle partition of $G(A_2)$ becomes a subgraph G_2 which consists of disjoint elementary circles and stems whose number of edges equals to the number of edges of the maximum weight cycle partition of $G(A_2)$. Let B = [B_1, B_2] and $G_M = G_1 U G_2$. Then G_M is a subgraph of G(A, B) consisting of disjoint elementary circles and stems whose number of edges equals to $r + s$ if the weight of the maximum weight cycle partition is $r + s * \varepsilon$.

Therefore, through the above process, each maximum weight cycle partition of G'(A) can induce a subgraph G_M of G(A,B) consisting of disjoint circles and stems originating from added control nodes. Furthermore, we have the following results:

Theorem 3.5: $GDCS(A,B) = |E(G_M)| = GDCS(A,B_0)$ and the minimum number of steering nodes is the number of added control nodes.

Proof: First we prove that $GDCS(A,B) = |E(G_M)|$. According to [38], it is sufficient to prove that

1) All nodes in G(A) are accessible from added control nodes
2) The subgraph G_M induced by a maximum weight cycle partition of G'(A) through the above process is a maximum subgraph of G(A,B) consisting of disjoint circles and stems originating from added control nodes

As all node in G(A) are accessible from nodes in S, to prove 1) we only need to illustrate that each of all nodes in S are accessible from at least one added control node. If there is one node in S that is inaccessible from any added control nodes, in the maximum weight cycle partition of G'(A), the circle containing this node should be in a source SCC, which contradicts against the construction of G_M where we have made sure that all nodes in each source SCC are accessible from an added control nodes. Now we employ the proof by contradiction technique to prove 2). If there exists a subgraph of G(A,B) consisting of disjoint circles and stems originating from added control nodes that contains more edges than G_M, by reversing the process of constructing G_M, we can get a cycle partition of G'(A) that has the larger weight, which is a contradiction.

Second, we prove $GDCS(A,B_0) = |E(G_M)|$. Note that $G(A,B) \subseteq G(A,B_0)$. Thus G_M is a subgraph of $G(A,B_0)$ consisting of disjoint circles and stems originating from added control nodes and the node corresponding to x_0. We need to prove that G_M is a maximum subgraph of $G(A,B_0)$ consisting of disjoint circles and stems originating from added control nodes and the node corresponding to x_0. Actually, if there exists a subgraph of $G(A,B_0)$ consisting of disjoint circles and stems originating from added control nodes and the node corresponding to x_0 that contains more edges than G_M, by reversing the process of constructing G_M, we can get a cycle partition of G'(A) that has the larger weight, which is a contradiction.

Third, we prove the second statement of this theorem. Denote by G_M^- the graph removed one control node from G_M and B^- the corresponding control matrix. Then at least one node in G'(A) is inaccessible from the rest of all added control nodes and thus $GDCS(A,B^{-}) = \left|E(G_M^{-})\right| < \left|E(G_M^{-})\right| + 1 \leq \left|E(G_M)\right|$. On the other hand, replacing the removed control node by node x_0, we can still have $GDCS(A,[x_0,B^{-}]) = \left|E(G_M)\right|$. This indicates that $GDCS(A,B^{-}) < GDCS(A,[x_0,B^{-}])$ if the number of steering nodes is less than the number of added control nodes. Therefore, the minimum number of steering nodes is the number of added control nodes.

From Theorems 1, we further conclude:

Theorem 3.6: Assume that the weight of the maximum circle partition is $r + s*\varepsilon$, where r and s are integers and $s*\varepsilon < 1$. Let t be the number of source SCC in G(A). Then, we have $GDCS(A,B) = GDCS(A,B_0) = r + s$ and the minimum number of steering nodes to be $s + t$.

Proof: From the process of constructing G_M, we can see that $\left|E(G_M)\right| = r + s$. If the weight of the maximum weight circle partition of G'(A) is an integer number, such a circle partition is also a maximum weight circle partition of G(A) and we have s=0. Each SCC of G(A) needs at least one control node. Therefore, the minimum number of steering nodes is $s + t = t$. If the weight of the maximum weight circle partition of G'(A) is a decimal number, from the construction of G_M, each edge with the weight of ε corresponds to one control node while each SCC of G(A) needs at least one control node. Therefore, the minimum number of steering nodes is s+t from Theorem 1. In summary, the minimum number of steering nodes needed for $GDCS(A,B) = GDCS(A,B_0)$ being true is $s + t$.

Note that the computational complexity of solving an optimal assignment of a weighted bipartite graph is $O(n^3)$ according to reference (39) for the worst cases in which a network is a complete graph. For the sparse networks which are true in most cases, our computational complexity is less than $O(n^3)$. Actually Table S5 and Figure S9 show that the our computational complexity is approximately $O(n^{2.35})$ for real complex networks with the number of nodes from 32 to 27772.

IV. Supplementary Tables

Table S1. The relationships of sources and targets in p53-mediated DNA damage response network

Sources	Targets	Sources	Targets	Sources	Targets
Wip1	ATM*	Wip1	P53killer	PIP	Akt*
ATM*	P53	P53arrester	P53killer	PTEN	PIP
Mdm2n	P53	ATM*	P53killer	P53arrester	Wip1
P53arrester	P53	P53	P53killer	P53killer	P53DINP1
P53killer	P53	Mdm2cp	Mdm2n	P53arrester	P53DINP1
Mdm2n	P53arrester	ATM*	Mdm2n	P53killer	PTEN
ATM*	P53arrester	Mdm2n	Mdm2cp	P53arrester	P21
P53	P53arrester	Akt*	Mdm2cp	P53killer	P53AIP1
Wip1	P53arrester	Mdm2c	Mdm2cp	P53AIP1	CytoC
P53killer	P53arrester	Akt*	Mdm2c	Casp3	CytoC
P53DINP1	P53arrester	P53killer	Mdm2c	CytoC	Casp3
Mdm2n	P53killer	P53arrester	Mdm2c	ATM*	ATM2
P53DINP1	P53killer	Mdm2cp	Mdm2c	ATM2	ATM*

Table S2. The relationships of sources and targets in T helper differentiation cellular network

Sources	Targets	Sources	Targets	Sources	Targets
GATA3	GATA3	STAT6	IL-12R	T-BET	SOCS1
STAT6	GATA3	IL-18	IL-18R	IFN-γR	STAT1
T-BET	GATA3	STAT6	IL-18R	IFN-βR	STAT1
IFN-β	IFN-βR	GATA3	IL-4	GATA3	STAT4
IRAK	IFN-γ	STAT1	IL-4	IL-12R	STAT4
STAT4	IFN-γ	IL-4	IL-4R	IL-4R	STAT6
T-BET	IFN-γ	SOCS1	IL-4R	GATA3	T-BET
IFN-γ	IFN-γR	IL-18R	IRAK	STAT1	T-BET
IL-12	IL-12R	STAT1	SOCS1	T-BET	T-BET

Table S3. The relationships of sources and targets in yeast cell cycle network

Sources	Targets	Sources	Targets	Sources	Targets
Cln3	SBF	Clb5,6	Clb1,2	Mcm1/SFF	Swi5
Cln3	Cln3	Clb5,6	Mcm1/SFF	Mcm1/SFF	Mcm1/SFF
Cln3	MBF	Clb1,2	Mcm1/SFF	Cdc20&Cdc14	Cdc20&Cdc14
SBF	Cln1,2	Clb1,2	Sic1	Cdc20&Cdc14	Sic1
MBF	Clb5,6	Clb1,2	SBF	Cdc20&Cdc14	Clb5,6
Cln1,2	Cln1,2	Clb1,2	MBF	Cdc20&Cdc14	Swi5
Cln1,2	Sic1	Clb1,2	Cdh1	Cdc20&Cdc14	Cdh1
Cln1,2	Cdh1	Clb1,2	Cdc20&Cdc14	Cdc20&Cdc14	Clb1,2
Sic1	Clb5,6	Clb1,2	Swi5	Swi5	Swi5
Sic1	Clb1,2	Cdh1	Clb1,2	Swi5	Sic1
Clb5,6	Sic1	Mcm1/SFF	Cdc20&Cdc14		
Clb5,6	Cdh1	Mcm1/SFF	Clb1,2		

Table S4. The relationships of sources and targets in EMT network

Sources	Targets	Sources	Targets	Sources	Targets
SNAI1	MIR203	SNAI1	ZEB1	MIR200	ZEB1
MIR203	SNAI1	SNAI1	ZEB2	SNAI1	MIR200
ZEB2	MIR203	ZEB2	MIR200	SNAI1	CDH1
MIR203	ZEB2	MIR200	ZEB2	ZEB1	CDH1
ZEB1	MIR203	ZEB1	MIR200	ZEB2	CDH1

Table S5. The average running time for identifying the steering kernel for 27 complex networks.

Types	Netowrks	Nodes	Edges	Time(s)*
Regulatory	TRN-yeast-1	4441	12873	33.53224
	TRN-yeast-2	688	1079	0.095352
	TRN-EC-1	1858	4123	1.34513
	TRN-EC-2	418	519	0.01755
	OwnershipUSCorp	7253	6726	32.67434
Trust	CollegeStu	32	96	2.96E-04
	PrisonIn	67	182	5.93E-04
	Wiki-Vote	7115	103689	5.80563
Food web	GrassLand	88	137	0.00117
	LittleRock	183	2484	0.007909
	SeaGrass	49	226	5.30E-04
	Ythan	135	601	0.002513
Metabolic	C.elegans	1173	2864	0.57091
	E.coli	2275	5763	3.59725
	S.cerevisiae	1511	3833	1.18267
Electronic circuit	S208	122	189	0.002091
	S420	252	399	0.013744
	S838	512	819	0.089687
Citation	HepTh	27772	352807	8775.1684
Internet	p2p-1	10876	39994	631.27528
	p2p-2	8846	31839	334.53616
	p2p-3	8717	31525	317.9278
Intraorganization	Consulting	46	879	0.00131
	Freemans1	34	6995	8.73E-04
	Freemans2	34	830	0.00117
	Manufacturing	77	2228	0.010375
Social	UCIOnline	1899	20296	2.85639

* The average CPU running times of 1000 randomly defined transitions of each network on a computer with the following specifications: operating system: Windows 7 home premium; processor: Intel(R) Core(TM) i7-3740QM CPU @ 2.70GHz; Memory: 12GB. The algorithm is implemented in Java 1.6.

V. Supplementary Figures

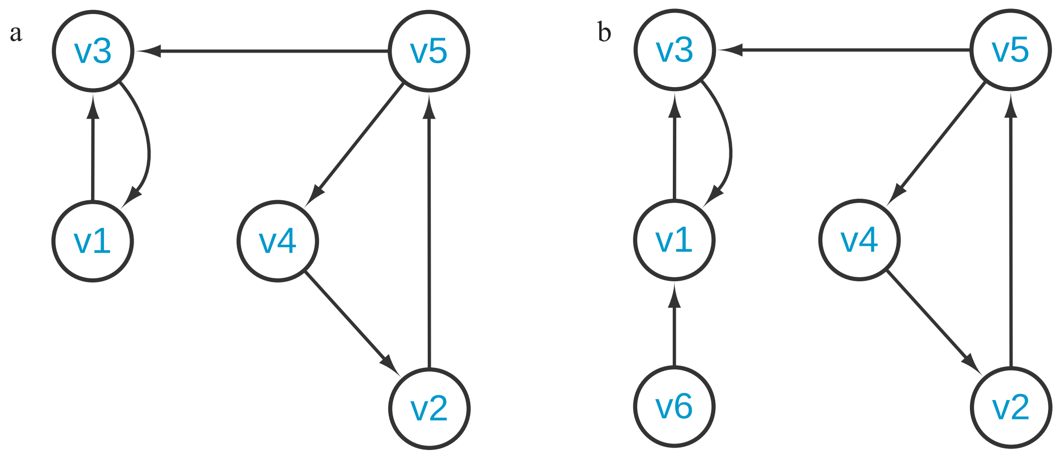
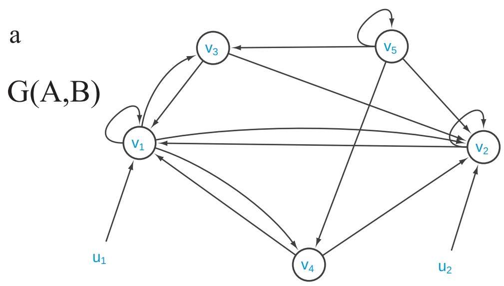


Figure S1. Two simple example networks which cannot be completely controlled by perturbing only the minimum set of driver nodes identified by the minimum input theorem (7).

a

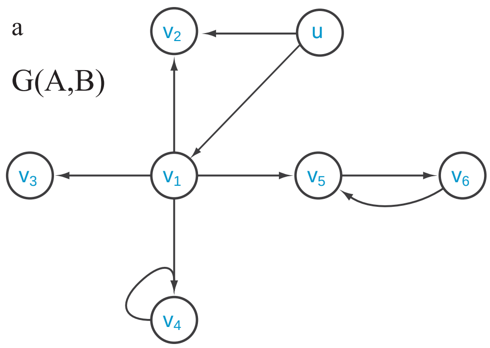


b

$$A=\begin{bmatrix} * & * & * & * & 0 \\ * & * & * & * & * \\ * & 0 & 0 & 0 & * \\ * & 0 & 0 & 0 & * \\ 0 & 0 & 0 & 0 & * \end{bmatrix}, B=\begin{bmatrix} * & 0 \\ 0 & * \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

GDSC(A,B) = 4 < 5, therefore the system (A,B) is not completely controllable

Figure S2: The association of a structural system (A, B) and its directed graph expression. “*” in matrix A and B represents the free parameters while “0” represents the fixed parameters. Each free parameter in (b) corresponds to an arc in the directed graph G(A,B) in (a).



$$A=\begin{bmatrix}0&0&0&0&0&0\\ *&0&0&0&0&0\\ *&0&0&0&0&0\\ *&0&0&*&0&0\\ *&0&0&0&0&*\\ 0&0&0&0&*&0\end{bmatrix}, B=\begin{bmatrix}*\\ *\\ 0\\ 0\\ 0\\ 0\end{bmatrix}$$

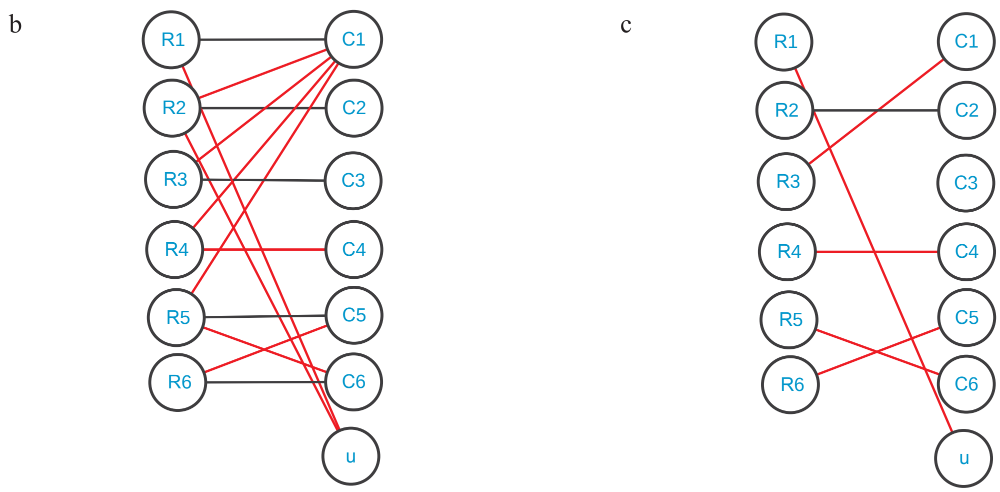


Figure S3: a) An example of G(A,B). “*” in matrix A and B represents the free parameters while “0” represents the fixed parameters. b) The weighted complete bipartite representation of G(A,B) shown in a) where each red edge has the weight of 1, each black edge has the weight of 0; c) one of the optimal assignments, from which we have GDCS(A,B)=5.

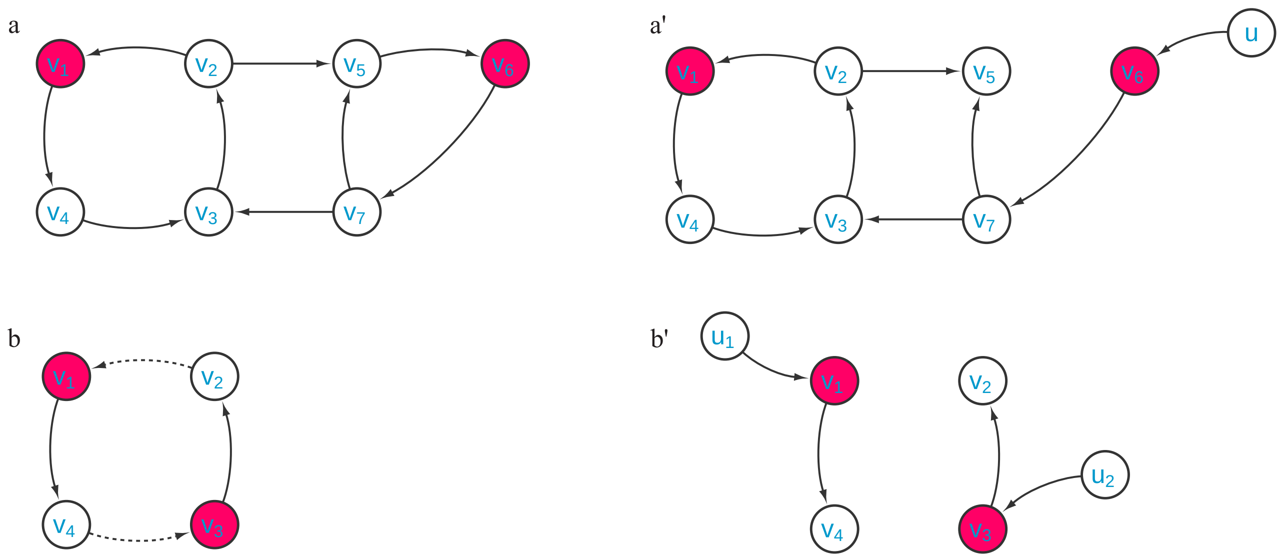


Figure S4. From circle partition to a subgraph consisting of the elementary circles and stems.

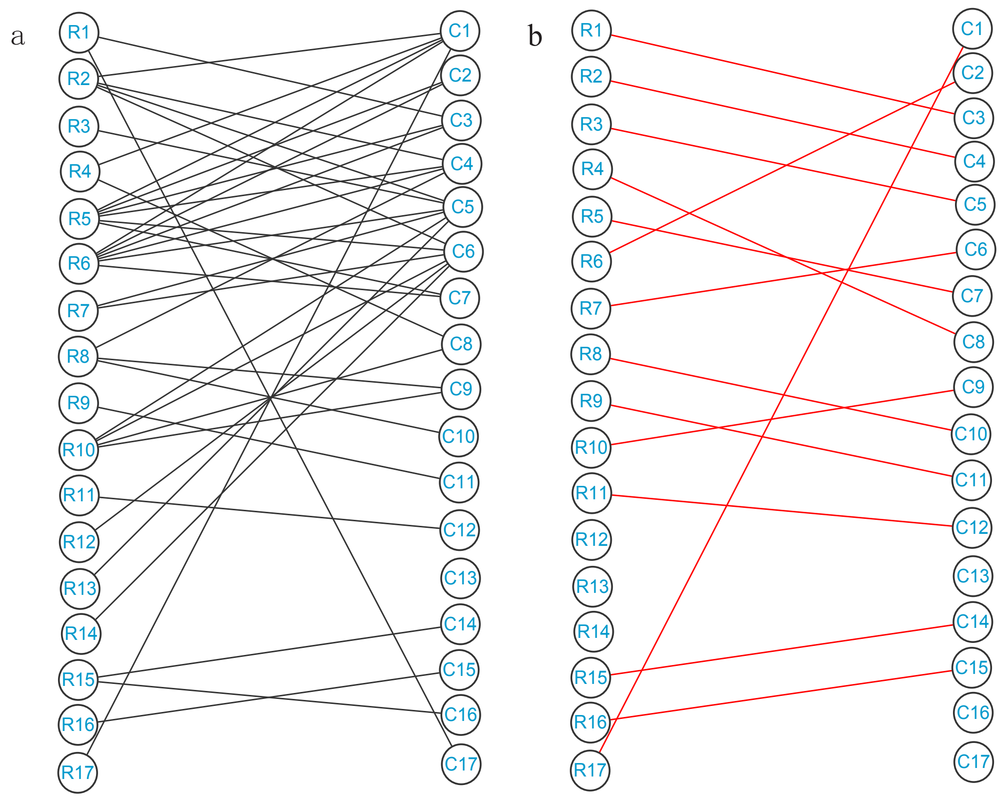


Figure S5. The bipartite representation of p53-mediated DNA damage response network (a) and one of its maximum matchings (b). In this figure, the nodes are numbered as follows: 1.ATM*; 2.P53; 3.Wip1; 4.Mdm2n; 5.P53arrester; 6.P53killer; 7.P53DINP1; 8.Mdm2cp; 9.Akt*; 10.Mdm2c; 11.PIP; 12.PTEN; 13.P21; 14.P53AIP1; 15.CytoC; 16.Casp3;17.ATM2. From the minimum input theorem (7), the minimum set of driver nodes consists of Wip1, PTEN and p53DINP1.

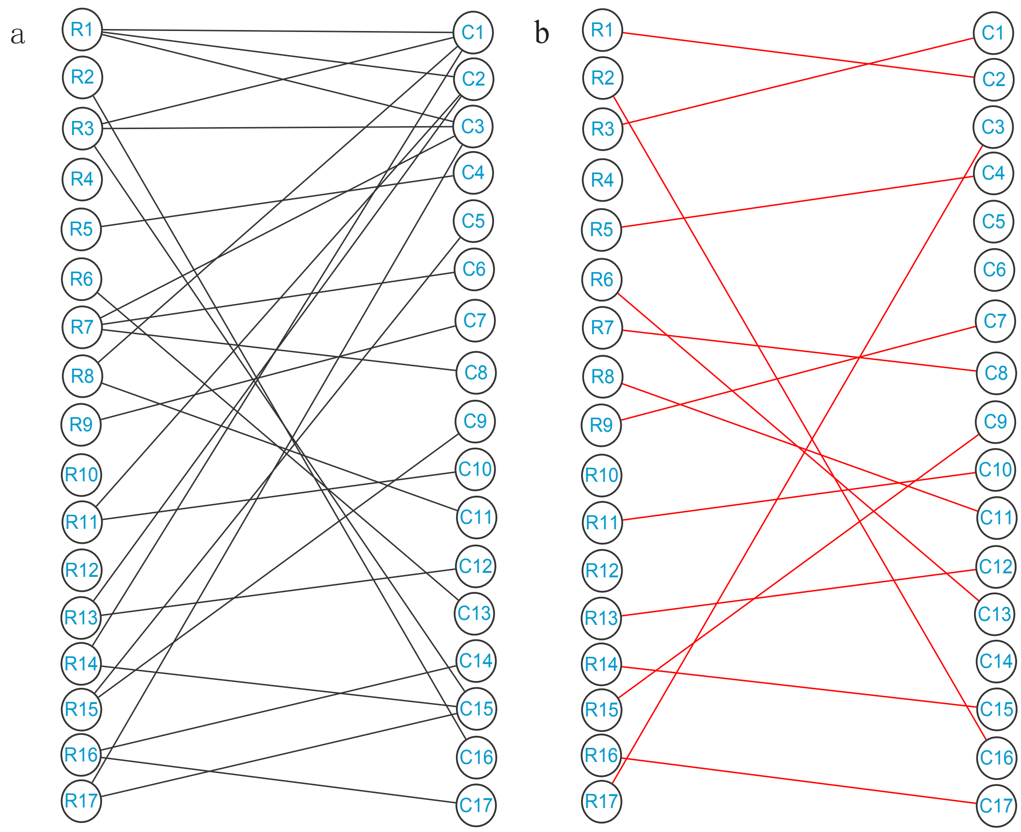


Figure S6. The bipartite representation of T helper differentiation cellular network (a) and one of its maximum matchings (b). In this figure, the nodes are numbered as follows: 1.GATA3; 2.STAT6; 3.T-BET; 4.IFN-¦Â; 5.IFN-¦ÂR; 6.IRAK; 7.IFN-¦Ã; 8.STAT4; 9.IFN-¦ÃR; 10.IL-12; 11.IL-12R; 12.IL-18; 13.IL-18R; 14.IL-4; 15.STAT1; 16.IL-4R; 17.SOCS1. From the minimum input theorem (7), the minimum set of driver nodes consists of IL-12, IL-18 and IFN-β.

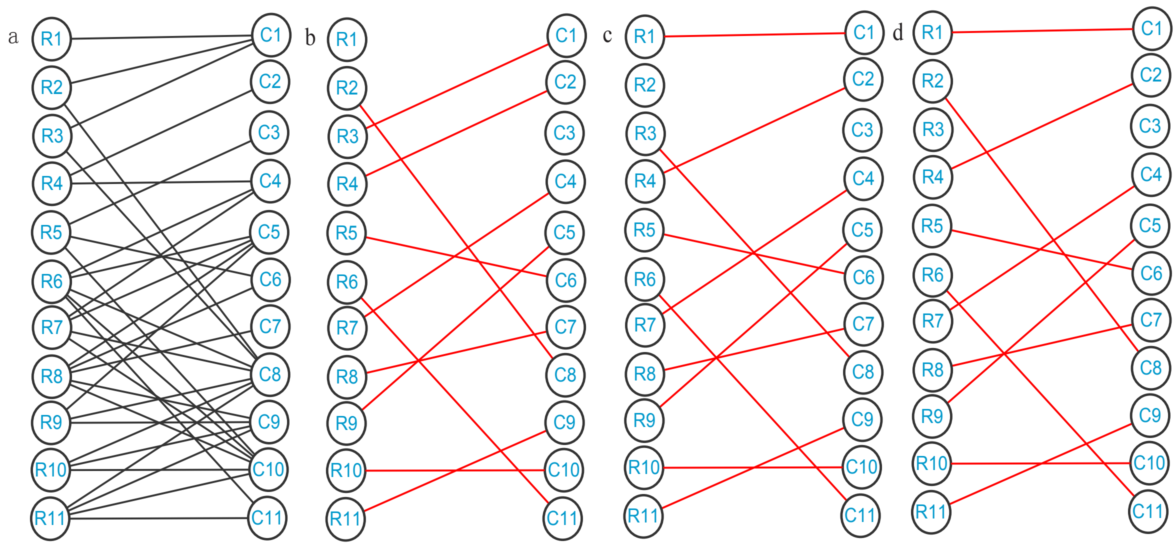


Figure S7. The bipartite representation of yeast cell cycle network (a) and three of its maximum matchings (b, c, d). In this figure, the nodes are numbered as follows: 1.Cln3; 2.SBF; 3.MBF; 4.Cln1,2; 5.Clb5,6; 6.Sic1; 7.Cdh1; 8.Clb1,2; 9.Mcm1/SFF; 10.Cdc20&Cdc14; 11.Swi5. From the minimum input theorem (7), three corresponding minimum sets of driver nodes are {Cln3}, {MBF} and {SBF}, respectively. However, perturbing either MBF or SBF cannot completely control this network as either of them cannot access to node Cln3 according to structural controllability theorem (30, 31).

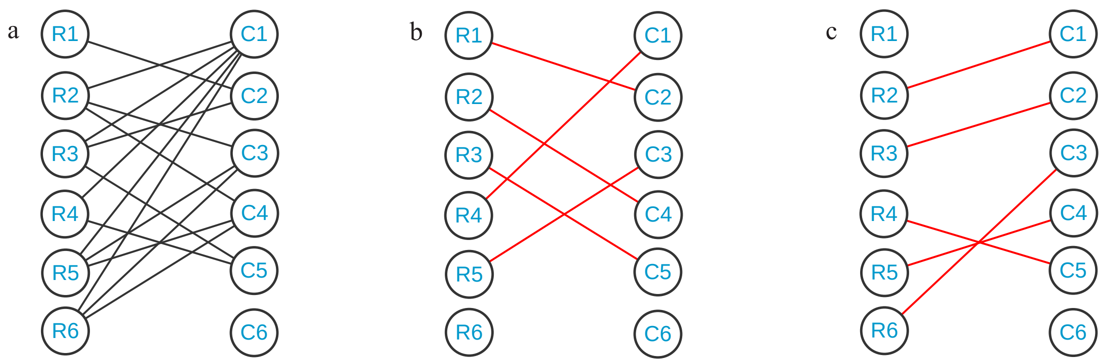


Figure S8. The bipartite representation of EMT network (a) and three of its maximum matchings (b,c). In this figure, the nodes are numbered as follows: 1.SNAI1; 2.MIR203; 3.ZEB2; 4.ZEB1; 5.MIR200; 6.CDH1. From the minimum input theorem (7), two corresponding minimum sets of driver nodes are {CDH1}, and {SNAI1}, respectively. However, perturbing CDH1 cannot completely control this network as it cannot access to any other node according to structural controllability theorem (30, 31).

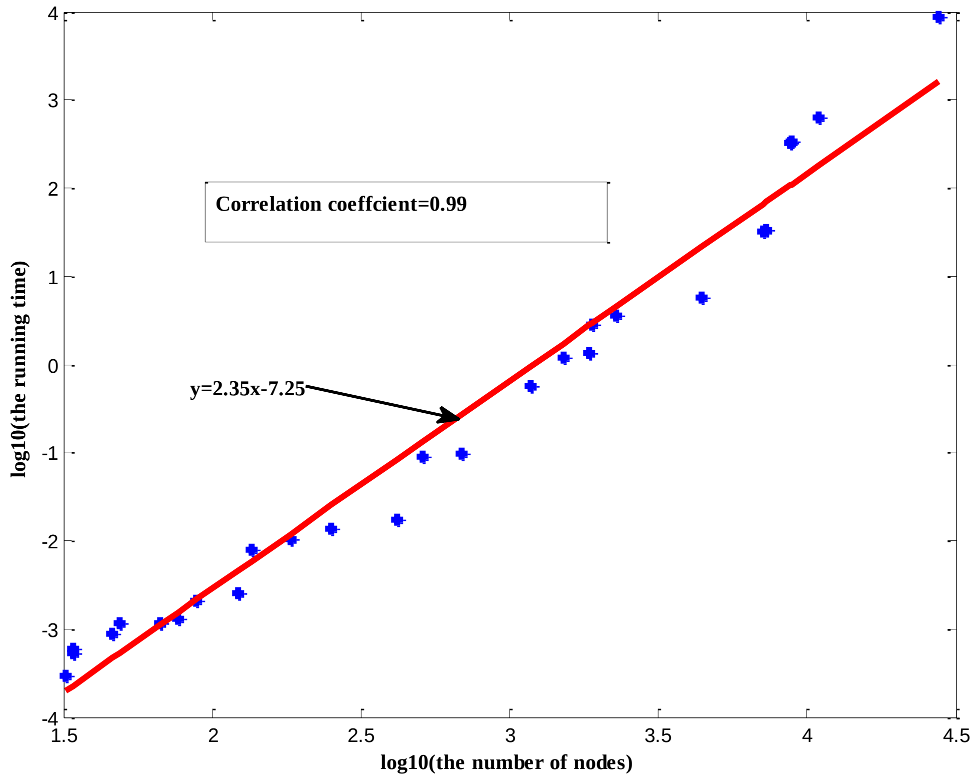


Figure S9. The scatter of the log10 of the CPU running time (y) vs. the log10 of the number of nodes(x) for 27 complex networks. From this figure, the log10 of the CPU running time is linear in the log10 of the number of nodes with a correlation coefficient of 0.99, and they fit a line equation y=2.35x-7.25, which indicates the computational complexity is $O(n^{2.35})$.

VI. Supplementary References

49. Hoese, S. & Matsumoto, K. On the irreducibility condition in the structural controllability theorem, *IEEE Trans Auto Contr.* **24**: 963-966 (1979).
50. Glover, K. & Silverman, L.M. Characterization of structural controllability, *IEEE Trans Auto Contr*. **21**: 534-537 (1976).
51. Blackhall, L. & Hill, D.J. On the structural controllability of networks of linear systems. *In: 2nd IFAC Workshop on Distributed Estimation and Control in Networked Systems*. 245–250 (2010).
52. Liu, Y.Y., Slotine, J.J., & Barabási, A.L. Control centrality and hierarchical structure in complex networks. *PLoS ONE* **7**, e44459 (2012).