Supplementary Information for The dynamical advantage of scale-free networks

Yimeng Qi[1,*], Songlin Zhuang[2,*], Zhihong Zhao[3,*], Xiaotian Lin[2], Xinghu Yu[4], Weichao Sun[1], FangzhouLiu[1,†], Charo I. del Genio[5], Baruch Barzel[6,7], Huijun Gao[1,†], Stefano Boccaletti[3,8,9]

1. Research Institute of Intelligent Control and Systems, Harbin Institute of Technology, Harbin 150001, China
2. Yongjiang Laboratory, Ningbo 315202, China
3. Institute of Interdisciplinary Intelligent Science, Ningbo Univ. of Technology, Ningbo, China
4. Ningbo Institute of Intelligent Equipment Technology Company Limited, Ningbo 294221, China
5. Institute of Smart Agriculture for Safe and Functional Foods and Supplements, Trakia University, Stara Zagora 6000, Bulgaria
6. Department of Mathematics, Bar-Ilan University, Ramat-Gan, Israel
7. The Gonda Multidisciplinary Brain Research Center, Bar-Ilan University, Ramat-Gan, Israel
8. Sino-Europe Complex Science Center, North University of China, Shanxi, Taiyuan 030051, China
9. CNR – Institute of Complex Systems, I-50019 Sesto Fiorentino, Italy

* These authors contributed equally to this work.

† Corresponding authors: fangzhou.liu@hit.edu.cn; hjgao@hit.edu.cn

Contents

1 Supplementary Section 1. Dynamical Systems

We consider three dynamical systems: the Kuramoto model of coupled phase oscillators ($d = 1$), and the Rössler and Lorenz systems ($d = 3$), and explicitly formulate their dynamics for single-node, network-coupled, and controlled network-coupled scenarios. A comprehensive summary of all parameters, coupling configurations, initial conditions, and numerical integration settings used in our simulations is provided in Table 1.

Table 1: Parameter configurations for the three considered dynamical systems. Simulation settings are reported at the bottom. (C) and (IC) denote compatible and incompatible control scenarios, respectively.

Category	**Parameter**	**Kuramoto**	**Lorenz**	**Rössler**
Node Dynamics	Dimension d	1	3	3
	State Variables	θ_i	x_i, y_i, z_i	x_i, y_i, z_i
	Intrinsic Params	$\omega_i \sim \mathcal{U}(-0.5, 0.5)$	$(\alpha, \rho, \beta) = (10, 28, 8/3)$	$(a, b, c) = (0.2, 0.2, 7)$
Coupling Setup	Coupled Variable	θ_i	y_i	z_i
	Gain K *(Fig. 2)*	0.01	0.01	0.01
	Gain K *(Others)*	1.0	10.0	5.0
	Gain K *(SI)*	0.5	5.0	1.0
Control Setup	Controlled Variable(s)	θ_i	z_i (C) x_i, y_i, z_i (IC)	y_i (C) x_i, y_i, z_i (IC)
	Compatible Gain ξ *(Fig. 2)*	1.0	2.0	2.0
	Compatible Gain ξ *(Others)*	1.0	10.0	10.0
	Compatible Gain ξ *(SI)*	1.0	10.0	10.0
Simulation Params	Initial States $x(0)$	$\theta_i \in [0, 2\pi)$	$\mathbf{x}_i \in [-0.5, 0.5]^3$	$\mathbf{x}_i \in [-0.5, 0.5]^3$
	Total Time T	25 s (Fig. 2); 25 s (Fig. 5: 20 s reaching, 5 s holding); 1000 s (Others)		
	IC Target State	Origin		
	Time Step	0.01 s		
	Integration Method	Fourth-order Runge-Kutta (RK4)		

Note: The extended horizon ($T = 1,000$ s) guarantees asymptotic convergence for precise MLE evaluations in compatible control. Shorter horizons ($T = 25$ s) are selectively utilized for conceptual illustration (Fig. 2 of the main text) and energy evaluation of pre-determined minimum driver node subsets (Fig. 5 of the main text), where prolonged steady-state simulation is computationally redundant.

1.1 Kuramoto Oscillator Dynamics

Single-node dynamics. The phase dynamics of an isolated Kuramoto oscillator are governed by

$$\dot{\theta} = \omega, \tag{1}$$

where $\theta(t)$ denotes the instantaneous phase of the oscillator and ω is its intrinsic natural frequency. For the specific isolated unit illustrated in Fig. 2b, we select a single realization with $\omega = 0.2$ and $\theta(0) = \pi/4$.

Network-coupled dynamics. For a population of N oscillators interacting over a network with adjacency matrix $A = (A_{ij})$, the phase dynamics of the i-th node expand to

$$\dot{\theta}_i = \omega_i + K \sum_{j=1}^{N} A_{ij} \sin(\theta_j - \theta_i), \tag{2}$$

where the natural frequencies ω_i are drawn from a uniform distribution $\omega \sim \mathcal{U}(-0.5, 0.5)$, and the initial phases

$\theta_i(0)$ are randomly generated within the interval $[0, 2\pi)$. $K = 0.01$ denotes the global coupling strength. The resulting trajectories for three representative nodes are showcased in Fig. 2e (left panel) and Fig. 2f (left panel) of the main text.

Targeting (compatible control). Targeting aims to synchronize the network to a dynamically compatible reference trajectory $\theta_i^{\mathrm{C}}(t)$, $\forall i$ generated by the uncontrolled network dynamics. Control is applied only to selected driver nodes $i \in S_{\min}^{\mathrm{C}}$:

$$\dot{\theta}_i = \omega_i + K \sum_{j=1}^{N} A_{ij} \sin(\theta_j - \theta_i) - \xi(\theta_i - \theta_i^{\mathrm{C}}(t)), \qquad i \in S_{\min}^{\mathrm{C}}, \tag{3}$$

with $\xi = 1.0$. This feedback drives the phases toward overlap with the compatible reference, as illustrated in Fig. 2e (right panel) of the main text.

Controllability (incompatible control). Controllability seeks to drive the network to an incompatible static state, here chosen as $\theta_i^{\mathrm{IC}} = 0$ for all nodes. Driver nodes $i \in S_{\min}^{\mathrm{IC}}$ are selected using structural controllability. Control inputs are computed via nonlinear model predictive control (NMPC):

$$\dot{\theta}_i = \omega_i + K \sum_{j=1}^{N} A_{ij} \sin(\theta_j - \theta_i) + u_i(t), \qquad i \in S_{\min}^{\mathrm{IC}}, \tag{4}$$

where $u_i(t)$ minimizes the cost

$$J = \int_0^T \left(\sum_{i=1}^{N} \theta_i(\tau)^2 + \mu \|\mathbf{u}(\tau)\|^2 \right) \mathrm{d}\tau. \tag{5}$$

The resulting controlled state is shown in Fig. 2f (right panel) of the main text.

1.2 Lorenz Oscillator Dynamics

Single-node dynamics. The temporal evolution of an isolated Lorenz oscillator is governed by

$$\begin{aligned} \dot{x} &= \alpha(y - x), \\ \dot{y} &= \rho x - y - xz, \\ \dot{z} &= xy - \beta z, \end{aligned} \tag{6}$$

where the intrinsic parameters are set to the standard chaotic regime with $\alpha = 10$, $\rho = 28$, and $\beta = 8/3$. For the specific realization depicted in Fig. 2c of the main text, the initial state $(x(0), y(0), z(0))$ is randomly initialized within the uniform interval $[-0.5, 0.5]$.

Network-coupled dynamics. When coupled on a network defined by the adjacency matrix $A = (A_{ij})$, interactions are introduced exclusively through the y-components. The dynamics of the i-th node thus expand to

$$\begin{aligned} \dot{x}_i &= \alpha(y_i - x_i), \\ \dot{y}_i &= \rho x_i - y_i - x_i z_i + K \sum_{j=1}^{N} A_{ij}(y_j - y_i), \\ \dot{z}_i &= x_i y_i - \beta z_i, \end{aligned} \tag{7}$$

where $K = 0.01$ dictates the global coupling strength. The resulting trajectories for three representative nodes are showcased in Fig. 2g (left panel) and Fig. 2h (left panel) of the main text.

Targeting (compatible control). Targeting drives all the nodes toward their respective compatible trajectories, denoted by $\mathbf{x}_i^{\mathrm{C}}(t) = (x_i^{\mathrm{C}}, y_i^{\mathrm{C}}, z_i^{\mathrm{C}})$, where control acts only on the z-variable of driver nodes $i \in S_{\min}^{\mathrm{C}}$:

$$\begin{aligned} \dot{x}_i &= \alpha(y_i - x_i), \\ \dot{y}_i &= \rho x_i - y_i - x_i z_i + K \sum_{j} A_{ij}(y_j - y_i), \\ \dot{z}_i &= x_i y_i - \beta z_i - \xi(z_i - z_i^{\mathrm{C}}(t)), \qquad i \in S_{\min}^{\mathrm{C}}, \end{aligned} \tag{8}$$

with $\xi = 2$. The resulting synchronization toward the reference is shown in Fig. 2g (right panel) of the main text.

Controllability (incompatible control). The controllability objective is to drive the network to the incom-

patible fixed point $\mathbf{x}_i^{\mathrm{IC}} = [0,0,0]^\top,\ \forall i$. Control inputs act on all state variables of each driver node $i \in S_{\min}^{\mathrm{IC}}$:

$$\begin{aligned}\dot{x}_i &= \alpha(y_i - x_i) + u_i^x(t),\\ \dot{y}_i &= \rho x_i - y_i - x_i z_i + K\sum_j A_{ij}(y_j - y_i) + u_i^y(t),\\ \dot{z}_i &= x_i y_i - \beta z_i + u_i^z(t),\end{aligned} \tag{9}$$

where $\mathbf{u}_i(t) = [u_i^x(t), u_i^y(t), u_i^z(t)]^\top \in \mathbb{R}^3$ is computed using NMPC by minimizing

$$J = \int_0^T \left(\sum_{i=1}^{N} \|\mathbf{x}_i(\tau) - \mathbf{x}_i^{\mathrm{IC}}\|^2 + \mu \|\mathbf{U}(\tau)\|^2 \right) \mathrm{d}\tau, \tag{10}$$

where $\mathbf{U}(t)$ stacks all control inputs $\mathbf{u}_i(t)$. The controlled trajectories are shown in Fig. 2h (right panel) of the main text.

1.3 Rössler Oscillator Dynamics

Single-node dynamics. The temporal evolution of an isolated Rössler oscillator follows

$$\begin{aligned}\dot{x} &= -y - z,\\ \dot{y} &= x + ay,\\ \dot{z} &= b + z(x - c),\end{aligned} \tag{11}$$

with parameters $a = 0.2$, $b = 0.2$, and $c = 7$. The trajectory shown in Fig. 2d of the main text is obtained without coupling or control.

Network-coupled dynamics. When coupled on a graph defined by the adjacency matrix $A = (A_{ij})$, interactions are introduced exclusively through the z-components. The dynamics of the i-th node thus expand to

$$\begin{aligned}\dot{x}_i &= -y_i - z_i,\\ \dot{y}_i &= x_i + ay_i,\\ \dot{z}_i &= b + z_i(x_i - c) + K\sum_j A_{ij}(z_j - z_i),\end{aligned} \tag{12}$$

with coupling strength $K = 0.01$. Three representative node trajectories are shown in Fig. 2i (left panel) and Fig. 2j (left panel) of the main text.

Targeting (compatible control). Targeting steers the network toward a compatible reference trajectory. Control is applied only to the y-variable of driver nodes $i \in S_{\min}^{\mathrm{C}}$:

$$\begin{aligned}\dot{x}_i &= -y_i - z_i,\\ \dot{y}_i &= x_i + ay_i - \xi(y_i - y_i^{\mathrm{C}}(t)), \qquad i \in S_{\min}^{\mathrm{C}},\\ \dot{z}_i &= b + z_i(x_i - c) + K\sum_j A_{ij}(z_j - z_i),\end{aligned} \tag{13}$$

with $\xi = 2.0$. Synchronization toward the reference is shown in Fig. 2i (right panel) of the main text.

Controllability (incompatible control). The network is driven toward the incompatible static state $\mathbf{x}_i^{\mathrm{IC}} = [0,0,0]^\top,\ \forall i$. Control inputs act on all state variables of driver nodes $i \in S_{\min}^{\mathrm{IC}}$:

$$\begin{aligned}\dot{x}_i &= -y_i - z_i + u_i^x(t),\\ \dot{y}_i &= x_i + ay_i + u_i^y(t),\\ \dot{z}_i &= b + z_i(x_i - c) + K\sum_j A_{ij}(z_j - z_i) + u_i^z(t),\end{aligned} \tag{14}$$

The inputs $\mathbf{u}_i(t) = [u_i^x(t), u_i^y(t), u_i^z(t)]^\top \in \mathbb{R}^3$ are obtained via NMPC by minimizing the same quadratic cost as in the Lorenz case. The resulting controlled state is shown in Fig. 2j (right panel) of the main text.

2 Supplementary Section 2. Control Framework and Evaluation Metrics

To compare control performance across different networks, dynamical systems, and target specifications, we evaluate each simulation using a unified set of structural, dynamical, and energy-based metrics.

2.1 Structural Metrics: Minimum Driver-Node Set

Structural metrics capture systematic differences in how the two control methods influence the overall network, independently of network size. This requires an analysis grounded in the underlying network topology. Since nodes constitute both the fundamental elements of the network and the primary targets of control interventions, the most natural structural metric is the minimum driver-node set.

2.1.1 Selection of $S_{\min}^{\mathrm{IC}}$ for Incompatible Control

Driver-node selection via star-expanded maximum matching. For pairwise networks, we adopt the maximum-matching procedure introduced in structural controllability theory [1] as a graph-theoretic reference for identifying nodes that receive external control inputs. However, directly applying this construction to undirected pairwise networks through the conventional bidirected representation may be problematic for the purposes of the present study. When each undirected edge $\{i, j\}$ is replaced by two directed edges $i \to j$ and $j \to i$, reciprocal edges naturally generate directed 2-cycles, while larger cyclic structures may also be completely covered by a maximum matching. As a result, many nodes belonging to such cycle-covered components do not appear as unmatched nodes and are therefore not identified as mandatory driver nodes according to the standard unmatched-node criterion. Although this behavior is consistent with the generic directed linear framework for which structural controllability was originally developed, it may be unsuitable as a standalone input-placement rule for bidirected representations of undirected pairwise interactions, where the two directions associated with an edge do not correspond to independent directed couplings.

A second limitation is that maximum matching depends exclusively on the zero–nonzero structure of the adjacency matrix A. Consequently, systems sharing the same sparsity pattern are treated as structurally equivalent, regardless of equality constraints among nonzero entries, reciprocity constraints, node-exchange symmetries, or the specific forms of the node dynamics and input matrix B. In symmetric networks, such constraints may generate invariant subspaces or indistinguishable node orbits that cannot be detected from the sparsity pattern alone [2]. Therefore, even when a network satisfies the maximum-matching condition at the topological level, the corresponding controlled system may still require additional input-placement analyses once reciprocal interactions, symmetry constraints, and nonlinear dynamics are explicitly taken into account.

To avoid treating an undirected pairwise interaction as two independent directed links, we employ a star-expanded representation prior to applying maximum matching. Given a pairwise graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, we introduce an auxiliary interaction node η_{ij} for each edge $e = \{i, j\} \in \mathcal{E}$. The resulting star-expanded graph is denoted by $\mathcal{G}^\star = (\mathcal{V} \cup \mathcal{V}_E, \mathcal{E}^\star)$, where $\mathcal{V}_E = \{\eta_{ij} : \{i, j\} \in \mathcal{E}\}$. For every pairwise edge $\{i, j\}$, we add the incidence links $i \to \eta_{ij}$, $j \to \eta_{ij}$, $\eta_{ij} \to i$, $\eta_{ij} \to j$. In this representation, the auxiliary node η_{ij} denotes the interaction channel associated with the original edge $\{i, j\}$ rather than an additional physical state variable.

We then apply the bipartite maximum-matching procedure to $\mathcal{G}^\star$. Let $\mathcal{U}_V(M)$ denote the set of unmatched incoming copies associated with the original state nodes, and let $\mathcal{U}_E(M)$ denote the set of unmatched incoming copies associated with the auxiliary interaction nodes. Because external control inputs in the Lorenz and Rössler network models act exclusively on the original state variables, auxiliary interaction nodes are not retained as physical driver nodes. Accordingly, for a matching M, we define the projected physical driver-node set as $S_{\min}^{\mathrm{IC}} = \mathcal{U}_V(M)$. The unmatched auxiliary nodes in $\mathcal{U}_E(M)$ are discarded after the matching step because they represent pairwise interaction channels rather than directly actuated dynamical states. The projected set $S_{\min}^{\mathrm{IC}}$ is then evaluated on the original pairwise network. Specifically, it determines the candidate input matrix B, whose nonzero rows correspond to the original nodes selected as driver nodes.

Maximum matching and minimal driver sets. We compute maximum matchings using an efficient bipartite representation in which each node i is split into two copies, i_{out} and i_{in}, and each directed edge $i \to j$ is mapped to a bipartite edge $i_{\mathrm{out}} \to j_{\mathrm{in}}$. A maximum matching is then obtained using a polynomial-time algorithm (e.g., Hopcroft–Karp), and the set of unmatched j_{in} nodes identifies the driver nodes required for structural

controllability. When multiple maximum matchings exist, the identity of individual driver nodes is not unique. In such cases, driver-node statistics (e.g., degree-conditioned selection probabilities) are estimated by sampling multiple maximum matchings through randomized tie-breaking and aggregating the resulting driver sets.

2.1.2 Selection of $S_{\min}^{\mathrm{C}}$ for Compatible Control

To determine whether a candidate control set S successfully stabilizes the system, we linearize the controlled dynamics around $\mathbf{x}^{\mathrm{C}}(t)$. For general time-dependent or non-stationary targets, the primary stability metric is the maximum Lyapunov exponent (MLE) of the controlled dynamics, denoted by $\lambda(S)$. A negative MLE indicates local contraction toward the target trajectory and robust stability against perturbations. Accordingly, the minimum-node selection problem can be formulated as

$$S_{\min}^{\mathrm{C}} = \arg\min_{S \subseteq V} \Big\{ |S| \;\Big|\; \lambda < 0 \Big\}.$$

MLE computation. For each candidate driver configuration, the controlled network dynamics are initialized from random initial conditions and evolved for a transient period of $T_{\mathrm{trans}} = 300$ time units with step size $\Delta t = 0.01$ using a fourth-order Runge–Kutta scheme. After discarding the transient, a reference trajectory $\mathbf{x}(t) \in \mathbb{R}^{Nd}$ is generated by integrating the controlled network dynamics, where $d = 1$ for Kuramoto oscillators and $d = 3$ for Lorenz and Rössler systems.

To estimate the MLE, we consider the evolution of an infinitesimal perturbation $\boldsymbol{\delta}(t) \in \mathbb{R}^{Nd}$ governed by the linearized dynamics along the reference trajectory. The perturbation is initialized as $\boldsymbol{\delta}(0) = [1/\sqrt{Nd}]^{Nd}$, corresponding to a normalized perturbation uniformly distributed across all degrees of freedom. The coupled evolution of the reference trajectory and the associated perturbation dynamics is then integrated for $T = 1000$ consecutive time windows, each consisting of 100 integration steps.

At the end of each time window, the perturbation norm $\|\boldsymbol{\delta}(t)\|_2$ is recorded and the perturbation is renormalized to unit norm before continuing to the next window. The MLE is computed as

$$\lambda = \frac{1}{T\,\Delta t} \sum_{n=1}^{T} \log \left\| \boldsymbol{\delta}(t_n) \right\|_2 , \tag{15}$$

where t_n denotes the final time of the n-th renormalization window.

For the special case of stationary targets stabilized by time-independent feedback, the stability analysis simplifies considerably. In this setting, local stability is determined directly from the spectrum of the effective Jacobian, $J_{\mathrm{eff}}(S) = J^{\star} - B_S K_S$, where $J^{\star}$ denotes the Jacobian of the uncontrolled dynamics evaluated at the target state, and B_S and K_S are the input and gain matrices associated with the selected control set S, respectively.

Greedy algorithm for constructing $S_{\min}^{\mathrm{C}}$. Determining the exact minimum stabilizing control set, $S_{\min}^{\mathrm{C}}$, through exhaustive search becomes computationally prohibitive for large networks because of the combinatorial growth in the number of candidate node configurations. To overcome this limitation, we employ a greedy ranking algorithm that iteratively constructs a stabilizing control set. At each iteration, the algorithm selects the node that produces the largest reduction in the maximum Lyapunov exponent.

Let $S \subseteq V$ denote the current active control set. For any unselected candidate node $i \in V \setminus S$, its marginal stabilizing contribution is quantified as

$$\Delta_i(S) = \lambda(S) - \lambda(S \cup \{i\}),$$

where $\lambda(S)$ denotes the fully optimized MLE under the current control architecture. A large positive value of $\Delta_i(S)$ indicates that adding node i substantially improves the local stability of the target state.

The procedure is initialized with the empty set, $S_0 = \varnothing$, for which the target is generally unstable. At iteration r, the algorithm evaluates $\Delta_i(S_r)$ for all candidate nodes $i \in V \setminus S_r$ and selects the optimal node $i_r^{\star} = \arg\max_{i \in V \setminus S_r} \Delta_i(S_r)$. The control set is then updated according to $S_{r+1} = S_r \cup \{i_r^{\star}\}$. Importantly, after each topological update, the control gains are re-optimized and the MLE is recomputed. This sequential procedure continues until strict local stability is achieved, i.e., $\lambda(S_r) < 0$, at which point the algorithm returns the final estimated stabilizing set $\widehat{S}_{\min}^{\mathrm{C}} = S_r$.

Algorithm Pseudocode:

Input: network A, target $\mathbf{x}^{\mathrm{C}}(t)$ or $\mathbf{x}^{\mathrm{C}}$, candidate node set V, MLE evaluation routine.

Initialize: $S \leftarrow \varnothing$.

While $\lambda(S) \geq 0$:

For each $i \in V \setminus S$, compute $\lambda(S \cup \{i\})$ after re-optimizing the control parameters.

Select

$$i^{\star} = \arg\min_{i \in V \setminus S} \lambda(S \cup \{i\}).$$

Update $S \leftarrow S \cup \{i^{\star}\}$.

Return: S.

The central idea is to minimize the cardinality of the directly actuated node set by iteratively selecting the nodes that provide the largest stabilizing effect. This approach transforms an otherwise prohibitive combinatorial optimization problem into a sequence of tractable one-step ranking problems. Importantly, the iterative re-evaluation accounts for the fact that a node's control effectiveness is not an intrinsic static property but depends on the current control configuration. Because the stabilizing effect of node combinations is generally non-additive, a node that appears ineffective when considered individually may become important once other complementary nodes are controlled. Re-evaluating node rankings at each iteration therefore allows these context-dependent effects to be captured.

Although the greedy algorithm does not guarantee identification of the global minimum stabilizing set, owing to the generally non-submodular dependence of the MLE on the selected control nodes, it performs effectively in practice. Its main advantage is that it directly optimizes the relevant dynamical stability metric rather than relying on structural surrogate measures that may not accurately reflect the underlying dynamics. In addition, repeated parameter re-optimization enables the ranking procedure to adapt to the evolving control configuration. The framework is sufficiently flexible to accommodate alternative ranking criteria, such as the absolute reduction in MLE, the relative reduction in MLE, or composite objectives that incorporate both stability and control-energy considerations, provided that the ranking is updated after each iteration. For small networks, the quality of the greedy approximation can be assessed through exhaustive enumeration. For larger networks, the method provides a scalable and interpretable approach for identifying stabilizing control nodes while maintaining a direct connection to the dynamical objective of the control problem. A formal complexity analysis is provided in Supplementary Note 4.

To evaluate the node-selection strategies, we benchmark the iterative greedy algorithm against a simpler degree-descending heuristic. As shown in Fig. 4 of the main text and Supplementary Figs. 1-4, simulations across a wide range of network topologies and dynamical systems reveal a systematic decrease in the MLE (λ) as additional nodes are included in the control set. The results show that selecting nodes in descending order of degree is generally sufficient to drive the system into the locally stable regime ($\lambda < 0$), and that the two approaches exhibit qualitatively similar stabilization behavior and comparable macroscopic trends. Motivated by this agreement, we employ the computationally efficient degree-descending heuristic as a proxy for the greedy algorithm in the structural analyses presented below.

2.1.3 Degree-Conditioned Driver Statistics

To systematically characterize the structural properties of the driver-node sets $S_{\min}^{\mathrm{IC}}$ and $S_{\min}^{\mathrm{C}}$, we consider the following metrics.

Degree-conditioned selection probability. For a node i of degree k, the probability of being selected as a driver node is defined as

$$P(i \in S_{\min} \mid k_i) = \frac{|\{i \in S_{\min} : k_i = k\}|}{|\{i : k_i = k\}|}, \; S_{\min} = \{S_{\min}^{\mathrm{C}}, \; S_{\min}^{\mathrm{IC}}\}. \tag{16}$$

When the number of nodes becomes sparse at large degree values, the same quantity is computed using logarithmically spaced degree bins $\mathcal{B}$:

$$P(S_{\min} \mid k \in \mathcal{B}) = \frac{|\{i \in S_{\min} : k_i \in \mathcal{B}\}|}{|\{i : k_i \in \mathcal{B}\}|}. \tag{17}$$

The corresponding results for all 48 network–dynamical systems are presented in Fig. 3 of the main text and Supplementary Figs. 5-8.

2.2 Dynamical Metrics: Accuracy and Efficiency

Our objective is to steer the system from an initial condition $\mathbf{x}(0) = \mathbf{x}_0$ toward a prescribed target state or trajectory, denoted by $\mathbf{x}^{\mathrm{C}}(t)$. To act on a subset of nodes S, we introduce a control signal $\mathbf{u}(t)$ through a binary input matrix B, where $B_{ij} = 1$ if the j-th state variable of node $i \in S$ is directly actuated, and $B_{ij} = 0$ otherwise. During the control process, the instantaneous state error $\varsigma(t)$ is defined as the Euclidean distance from the target:

$$\varsigma(t) = \|\mathbf{x}(t) - \mathbf{x}^{\mathrm{C}}(t)\|_2$$

To quantify control performance, we use the convergence time as the primary metric. It is defined as the minimum time required for the state error to fall below a prescribed tolerance threshold δ and remain below it thereafter:

$$t_{\mathrm{c}}^{\mathrm{IC}} = \inf\{t > 0 \mid \varsigma(\tau) \leq \delta,\ \forall \tau \geq t\}$$

and

$$t_{\mathrm{c}}^{\mathrm{C}} = \inf\{t > 0 \mid \lambda < 0 \ \text{ and } \ \varsigma(\tau) \leq \delta,\ \forall \tau \geq t\}.$$

This metric quantifies the time required for the selected control set S to drive the system into the target regime.

2.3 Energy Metrics: Cost and Gap

To characterize the energetic requirements of the control tasks, we consider the generic closed-loop network dynamics:

$$\dot{\mathbf{x}}(t) = \mathbf{F}(\mathbf{x}(t)) + \mathbf{G}(\mathbf{x}(t), A) + B\mathbf{u}(t).$$

Here, $\mathbf{F}(\mathbf{x}(t))$ represents the intrinsic nodal dynamics in the absence of interactions, whereas $\mathbf{G}(\mathbf{x}(t), A)$ describes the coupling among neighboring nodes as determined by the network adjacency matrix A. The control inputs $\mathbf{u}(t)$ are external signals applied to a subset of driver nodes $S \subseteq V$ through the binary input matrix B.

Once the system reaches the prescribed target trajectory $\mathbf{x}^{\mathrm{C}}(t)$, a control input $\mathbf{u}_{\mathrm{H}}(t)$ may be required to compensate for any mismatch between the target evolution and the natural dynamics of the uncontrolled system. Using the Moore–Penrose pseudoinverse $B^{\dagger}$, the corresponding minimum-norm sustaining input is given by

$$\mathbf{u}_{\mathrm{H}}(t) = B^{\dagger}\Big[\dot{\mathbf{x}}^{\mathrm{C}}(t) - \mathbf{F}(\mathbf{x}^{\mathrm{C}}(t)) - \mathbf{G}(\mathbf{x}^{\mathrm{C}}(t), A)\Big]$$

For a stationary target state ($\dot{\mathbf{x}}^{\mathrm{C}}(t) \equiv 0$), this expression reduces to $\mathbf{u}_{\mathrm{H}} = -B^{\dagger}\Big[\mathbf{F}(\mathbf{x}^{\mathrm{C}}) + \mathbf{G}(\mathbf{x}^{\mathrm{C}}, A)\Big]$. This relation shows that the control input must compensate for both the intrinsic nodal dynamics ($\mathbf{F}$) and the network-induced interactions ($\mathbf{G}$). The magnitude of the sustaining input determines the asymptotic holding cost, which motivates the following classification of control tasks:

- **Compatible control:** The target is consistent with the natural dynamics of the uncontrolled coupled system (e.g., an equilibrium state, periodic orbit, or intrinsic trajectory). In this case, $\dot{\mathbf{x}}^{\mathrm{C}}(t) = \mathbf{F}(\mathbf{x}^{\mathrm{C}}(t)) + \mathbf{G}(\mathbf{x}^{\mathrm{C}}(t), A)$, implying that the required sustaining input vanishes asymptotically, $\mathbf{u}_{\mathrm{H}}(t) \to 0$, once the target is reached. Consequently, the asymptotic holding energy is expected to be negligible ($E_{\mathrm{H}} \approx 0$).
- **Incompatible control:** The target is externally imposed and is not naturally supported by the collective network dynamics. Consequently, $\dot{\mathbf{x}}^{\mathrm{C}}(t) \neq \mathbf{F}(\mathbf{x}^{\mathrm{C}}(t)) + \mathbf{G}(\mathbf{x}^{\mathrm{C}}(t), A)$, requiring a persistent nonzero sustaining input ($\mathbf{u}_{\mathrm{H}}(t) \neq 0$) to maintain the system at the prescribed target. As a result, the holding energy E_{H} accumulates continuously and may constitute a substantial fraction of the overall control cost.

2.3.1 Energy Cost

To evaluate the energetic cost of control, we distinguish between the energy required to steer the system toward the target and the energy required to maintain it there. These contributions are referred to as the **reaching energy** (E) and the **holding energy** (E_{H}), respectively.

The instantaneous control power is defined as $W(t) = \|\mathbf{u}^2(t)\|$. Over a finite observation interval $[0, T]$, the total control energy E_{total} can be decomposed at the convergence time t_{c} as

$$E_{\mathrm{total}} = E + E_{\mathrm{H}} = \int_0^{t_{\mathrm{c}}} \|\mathbf{u}^2(t')\|\,\mathrm{d}t' + \int_{t_{\mathrm{c}}}^{T} \|\mathbf{u}^2(t')\|\,\mathrm{d}t'$$

where $t_c = \max\{t_c^{IC},\ t_c^{C}\}$ denotes the convergence time identified across the two control scenarios.

2.3.2 Energy Gap

To quantify the energetic difference between steering the network toward a dynamically compatible target and enforcing an incompatible target state, we define an energy gap measure between the two control protocols. Let E_C and E_{IC} denote the reaching energies associated with the compatible and incompatible control protocols, respectively, measured over the interval $[0,\ t_c]$. The corresponding results for all 16 network–dynamics combinations are shown in Supplementary Fig. 9.

We define the reaching-energy gap as

$$\varepsilon = \frac{E_{IC}}{E_C}, \tag{18}$$

where $\varepsilon > 1$ indicates that the incompatible target requires more energy to reach than the compatible target, while $\varepsilon < 1$ indicates the opposite. Values near unity correspond to comparable reaching costs under the two control protocols.

This quantity characterizes the overall energetic penalty associated with incompatibility between the imposed target and the intrinsic network dynamics. However, total energy does not distinguish between transient steering and long-term maintenance. To isolate the latter contribution, we additionally define a tail-power gap based on the average control power during the holding stage. If

$$\langle W_H \rangle = \frac{1}{T - t_c} \int_{t_c}^{T} \|\mathbf{u}^2(t')\| \, \mathrm{d}t'$$

denotes the average control power over the interval $[t_c,\ T]$, then

$$\varepsilon_H = \frac{\langle W_H^{IC} \rangle}{\langle W_H^{C} \rangle}. \tag{19}$$

Comparisons of $\langle W_H \rangle$ across all 16 network–dynamics combinations are presented in Fig. 10. The quantity ε_H measures the additional maintenance cost required to sustain the system near an incompatible target after the transient convergence phase. Together, ε and ε_H provide complementary information: the former quantifies the total energetic burden accumulated during the control process, whereas the latter isolates the extent to which the incompatible target requires persistent external forcing.

Figure 5 of the main text and Supplementary Fig. 23 show ε and ε_H as functions of the network heterogeneity ρ, defined in Eq. (8) of the main text. The results indicate that increasing heterogeneity tends to amplify the energetic disadvantage of incompatible control relative to dynamically compatible targeting, while simultaneously enhancing functional robustness.

3 Supplementary Section 3. Graph Ensembles

3.1 Synthetic Graphs with Fixed (N, F)

All synthetic graphs are generated with fixed numbers of nodes N and edges F to enable cross-topology comparisons at matched density. For each ensemble, we prescribe a target degree sequence $\{k_i\}_{i=1}^{N}$ consistent with a desired degree distribution $P(k)$ and construct simple graphs using the configuration model with stub matching. Self-loops and multi-edges generated during the pairing process are eliminated through repeated pairing and rejection until a simple graph containing exactly F edges is obtained.

For Erdős–Rényi (ER) graphs, edges are sampled uniformly at random from the $\binom{N}{2}$ possible undirected node pairs until exactly F edges are present. For scale-free (SF) graphs, node degrees are sampled from a power-law distribution $P(k) \propto k^{-\gamma}$ over a finite support $[k_{\min},\ k_{\min}]$, after which the degree sequence is rescaled to satisfy $\sum_i k_i = 2F$. The corresponding configuration-model graph is then generated from the resulting sequence.

Additional synthetic topologies (e.g., degree-homogeneous or narrow-tailed ensembles) are constructed analogously by specifying the appropriate degree distribution $P(k)$ and applying the same fixed-(N, F) generation procedure. Unless otherwise stated, all reported quantities are averaged over multiple independent realizations generated with different random seeds.

3.2 Size-Matched ER Rewiring for Empirical Networks

For each empirical network, we construct an ER-based null model with the same size (N, F). Given an empirical undirected simple graph with N nodes and F edges, an ER graph is generated by uniformly sampling F edges without replacement from the set of all $\binom{N}{2}$ possible node pairs. This yields a degree-homogeneous null model at matched density.

3.3 Network Subsampling and Extraction Protocols

To enable computationally tractable dynamical simulations on large empirical networks while preserving their principal structural characteristics, we employ targeted subgraph extraction procedures.

Random walk-based sampling for densely connected networks. For densely connected empirical networks (e.g., the Brain 1 network analyzed in the main text), we use a random walk-based sampling strategy initialized within a densely connected core. This approach is preferred over uniform random sampling because it better preserves network connectivity and avoids artificial fragmentation. The resulting subgraphs retain the heavy-tailed degree distribution $P(k)$ and the presence of high-degree hub nodes characteristic of the original network. These structural features are particularly relevant for the control analyses considered in this work.

Connected subgraph extraction for sparse empirical networks. For large empirical networks derived from technological infrastructures (Caida and Oregon) and social interaction systems (Email and Math), we extract connected subnetworks by selecting either the largest connected component or a connected induced subgraph. This procedure provides representative test systems for evaluating the two control paradigms while avoiding isolated nodes. The extracted subnetworks preserve the principal structural characteristics of the original networks, including their heterogeneous degree distributions, hierarchical organization, and community structure.

4 Supplementary Note 1. Description of Extended Synthetic and Real-World Datasets

To assess the robustness and generality of the results obtained for the canonical dynamical models considered in the main text (Kuramoto, Lorenz, and Rössler systems), we extend our simulations to an additional set of denser synthetic networks and four empirical real-world networks.

4.1 Denser Synthetic Networks

To investigate the influence of network density on the observed dynamical behavior, we generate an additional collection of synthetic networks. These networks belong to the same topological classes considered in the main text, including homogeneous networks with bounded degree distributions and heterogeneous scale-free networks characterized by $P(k) \sim k^{-\gamma}$, with scaling exponents $\gamma = 3$ (SF1), $\gamma = 2.5$ (SF2), and $\gamma = 2$ (SF3). The principal difference is that the average degree is increased, resulting in substantially denser network structures. This allows us to evaluate whether the observed dynamical behavior remains consistent in more densely connected regimes.

4.2 Empirical Real-World Networks

In addition to the synthetic topologies, we evaluate the dynamics on four empirical networks: Caida, Oregon, Email, and Math. These datasets are obtained from publicly available repositories, including the Stanford Network Analysis Platform (SNAP) and the Network Repository, and are selected to represent a diverse range of technological and social interaction systems.

- **Caida** and **Oregon**: These networks describe Internet-level connectivity through Autonomous System (AS) peering and routing relationships. Both exhibit heterogeneous degree distributions and hierarchical organization typical of large-scale communication infrastructures. The Caida dataset represents inferred business relationships among Internet Service Providers [3], whereas the Oregon dataset is derived from BGP (Border Gateway Protocol) route-view measurements collected by the University of Oregon Route Views Project [4].

- **Email**: This dataset corresponds to the Enron email communication network, in which nodes represent email addresses and edges indicate at least one exchanged message[5].
- **Math**: This network is derived from the MathSciNet database and represents a co-authorship network in which nodes correspond to researchers and edges denote joint publications[6].

The sizes of both the original datasets and the extracted subnetworks used in the dynamical simulations are reported in Supplementary Table 2. The corresponding topological characteristics and degree distributions are compared in Supplementary Fig. 11.

Table 2: **Topological characteristics of the original and subsampled empirical networks.** The original networks originate from a variety of technological and social domains. To enable computationally demanding dynamical simulations, connected subnetworks are extracted using targeted sampling procedures designed to preserve key structural characteristics, including degree heterogeneity and hierarchical organization.

Dataset	Original Nodes	Original Edges	Subnetwork Nodes	Subnetwork Edges
Caida	10,500	22,500	3,000	4,485
Oregon	10,670	22,002	3,000	4,770
Email	36,692	183,831	5,000	8,250
Math	332,000	820,000	5,000	8,078

5 Supplementary Note 2. Robustness of Evaluated Metrics Across Extended Network Datasets

In the main text, we illustrate the differences between the two control paradigms using representative ER and SF1 networks. To assess the robustness of these observations, we extend the analysis to the remaining network classes considered in the main text, as well as to the additional synthetic and empirical datasets introduced in Supplementary Note 1, under all three dynamical models.

Robustness of structural metrics. We first extend the analysis of driver-node statistics, characterized by the relationship between $P(i \in S_{\min} \mid k_i)$ and k_i. Supplementary Figs. 5-8 present the results for the remaining network classes considered in the main text, while Supplementary Figs. 17-20 report the corresponding results for the denser synthetic networks and empirical datasets. In addition, the relationship between the functional robustness ϕ and the network heterogeneity ρ, highlighted in Fig. 3 of the main text, is evaluated across all supplementary datasets (Supplementary Fig. 12).

Robustness of energy metrics. The dependence of the energy-gap measures (ε and ε_{H}) on network heterogeneity ρ, reported in the main text, is also examined across the full set of supplementary datasets. The corresponding results are shown in Supplementary Fig. 23.

Generalizability of algorithmic performance. To reduce the computational cost associated with determining $S_{\min}^{\mathrm{C}}$, we employ the degree-descending heuristic as an alternative to the greedy algorithm. Supplementary Figs. 1-4 and 13-16 compare the performance of the two approaches across the remaining original networks as well as the extended synthetic and empirical datasets.

Consistency of the observed trends. Across the full benchmark set, the qualitative trends reported in the main text remain largely unchanged. The distributions of $|S_{\min}|$, the behavior of λ, and the energy-consumption patterns exhibit similar characteristics across the extended datasets, indicating that the principal observations are not restricted to a specific network class.

In particular, compatible control is generally associated with high-degree nodes, whereas incompatible control is more frequently associated with low-degree nodes or with a more distributed pattern of control across the network. The energy analysis further suggests that network structure facilitates control toward dynamically compatible targets, especially in heterogeneous networks. In contrast, incompatible control typically requires larger control effort and greater maintenance cost, particularly in highly heterogeneous networks.

These observations are consistent with the positive relationship between the energy-gap measures and network

heterogeneity shown in Fig. 5s–t of the main text and Supplementary Fig. 23. Moreover, a clear distinction emerges between the energetic requirements of the two control paradigms. For targets aligned with the natural attractors of the system, the long-term maintenance cost is generally negligible compared with the energy required during the initial convergence phase. By contrast, targets that are not supported by the intrinsic dynamics require persistent external forcing and therefore incur a substantial holding cost.

This distinction becomes increasingly pronounced as network heterogeneity increases. Under compatible control, heterogeneous networks generally require less energy than homogeneous networks, whereas the opposite trend is observed for incompatible control. These results are consistent with the interpretation that heterogeneous networks exhibit greater functional robustness, facilitating recovery toward naturally supported dynamical states while simultaneously resisting sustained deviations from them.

6 Supplementary Note 3. Complexity Analysis of Degree-Descending Heuristic and Greedy Algorithms

To evaluate the computational efficiency of the proposed node-selection strategies, we analyze the complexity of both the degree-descending heuristic and the greedy iterative algorithm. Let N denote the number of nodes, F the number of edges, and $C_{\mathrm{MLE}}(N)$ the computational cost of a single Maximum Lyapunov Exponent (MLE) evaluation. We further denote by $k^\star = |S^{\mathrm{C}}_{\mathrm{min}}|$ the size of the exact minimum stabilizing set. The sizes of the stabilizing sets identified by the two approximation methods are denoted by k_{degree} and k_{greedy}, respectively. The complexity bounds derived below remain valid regardless of the specific numerical procedure used to evaluate the MLE, including Jacobian-based methods, variational-equation integration, and simulation-based estimators.

6.1 Degree-Descending Heuristic

The degree-descending heuristic is motivated by the empirical observation that high-degree nodes tend to contribute more strongly to stabilization under compatible control. The algorithm consists of three stages.

Degree computation. Computing the degree of all nodes in an undirected network requires traversing the node and edge sets. This step has time complexity $O(N + F)$.

Sorting. The nodes are then sorted in descending order according to degree. Using a comparison-based sorting algorithm such as Merge Sort or Heap Sort, this step requires $O(N \log N)$ operations.

Sequential validation. Nodes are added sequentially following the degree ranking, and the MLE is evaluated after each addition to determine whether the system has become stable. In the worst case, identifying a stabilizing set of size k_{degree} requires exactly k_{degree} MLE evaluations, yielding a complexity of $O(k_{\mathrm{degree}} \cdot C_{\mathrm{MLE}})$.

Combining these contributions, the total complexity of the degree-descending heuristic is

$$O(N + F) + O(N \log N) + O(k_{\mathrm{degree}} \cdot C_{\mathrm{MLE}}).$$

For sparse networks ($F \ll N^2$) and computationally expensive dynamical evaluations ($C_{\mathrm{MLE}} \gg N$), this expression simplifies to

$$O(N \log N + k_{\mathrm{degree}} \cdot C_{\mathrm{MLE}}).$$

6.2 Greedy Algorithm

Unlike the degree-descending heuristic, the greedy algorithm directly optimizes a dynamical objective by selecting, at each iteration, the node that produces the largest reduction in the MLE.

At the i-th step ($i \in [1, k_{\mathrm{greedy}}]$), the algorithm evaluates the MLE for all remaining candidate nodes. The number of candidate evaluations performed at this stage is therefore $N - i + 1$. The total number of MLE evaluations required to construct a stabilizing set of size k_{greedy} is

$$\sum_{i=1}^{k_{\mathrm{greedy}}} (N - i + 1) = k_{\mathrm{greedy}} \cdot N - \frac{k_{\mathrm{greedy}}(k_{\mathrm{greedy}} - 1)}{2}$$

.

Since $k_{\text{greedy}} \leq N$, the number of evaluations scales asymptotically as $O(k_{\text{greedy}} \cdot N)$. The resulting overall time complexity is therefore $O(k_{\text{greedy}} \cdot N \cdot C_{\text{MLE}})$.

6.3 Comparison and Practical Considerations

Both the greedy algorithm and the degree-descending heuristic are approximation methods for the minimum stabilizing-set problem. Determining the exact optimum through exhaustive enumeration is generally infeasible for large networks because of the combinatorial growth in the number of candidate node subsets. Specifically, exhaustive search requires evaluating

$$O\left(C_{\text{MLE}} \sum_{i=1}^{k^\star} \binom{N}{i}\right)$$

candidate configurations.

The greedy algorithm mitigates this combinatorial cost by replacing exhaustive enumeration with a sequence of local optimization steps. Nevertheless, its complexity remains dominated by repeated MLE evaluations, resulting in an overall cost of $O(k_{\text{greedy}} \cdot N \cdot C_{\text{MLE}})$. Since MLE estimation itself typically involves repeated numerical integration and Jacobian updates, this cost can become substantial for large-scale dynamical systems. By contrast, the degree-descending heuristic relies on a purely topological ranking and therefore requires only a single sorting operation followed by sequential validation. Its overall complexity is $O(N \log N + k_{\text{degree}} \cdot C_{\text{MLE}})$, which is substantially lower than that of the greedy approach when N is large.

From a theoretical perspective, both methods provide upper bounds on the optimal stabilizing-set size and therefore satisfy

$$k_{\text{greedy}} \geq k^\star, \qquad k_{\text{degree}} \geq k^\star.$$

Empirically, however, simulations of the dynamical systems considered in this work (Fig. 4 in the main text and Supplementary Figs. 1-4, 13-16) indicate that the two methods often yield stabilizing sets of comparable size, with

$$k_{\text{degree}} \approx k_{\text{greedy}}.$$

These observations suggest that the degree-descending heuristic can provide a useful approximation to the greedy solution while substantially reducing computational cost. For this reason, it serves as a practical alternative for large-scale network-control analyses where repeated dynamical optimization becomes computationally demanding.

7 Supplementary Note 4. Justification of Variable Simulation Horizons

The total simulation horizon T is selected according to the objective of each numerical experiment in order to adequately capture the relevant dynamical behavior.

For the determination of driver-node sets (Figs. 3 and 4 in the main text, together with Supplementary Figs. 12-16), a relatively long simulation horizon of $T = 1000$ s is employed. This allows the system dynamics to approach their asymptotic regime and provides sufficient temporal data for reliable estimation of the MLE associated with each candidate control set.

By contrast, shorter simulation horizons are used when the objective is to visualize transient dynamics or to evaluate previously identified driver-node sets. For the illustrative examples presented in Fig. 2 of the main text, a simulation horizon of $T = 25$ s is sufficient to demonstrate the qualitative differences between the two control paradigms.

Similarly, in Fig. 5 of the main text, the driver-node sets have already been determined and the focus shifts to quantifying the energetic cost of control. In this case, a simulation horizon of $T = 25$ s is adopted and divided into two stages: a 20 s reaching phase, during which the system approaches the target state, followed by a 5 s holding phase used to evaluate the maintenance cost of control. With the control sets fixed, this time window is sufficient to characterize both the transient convergence dynamics and the subsequent steady-state behavior,

enabling the computation of the energy-based metrics introduced in the main text.

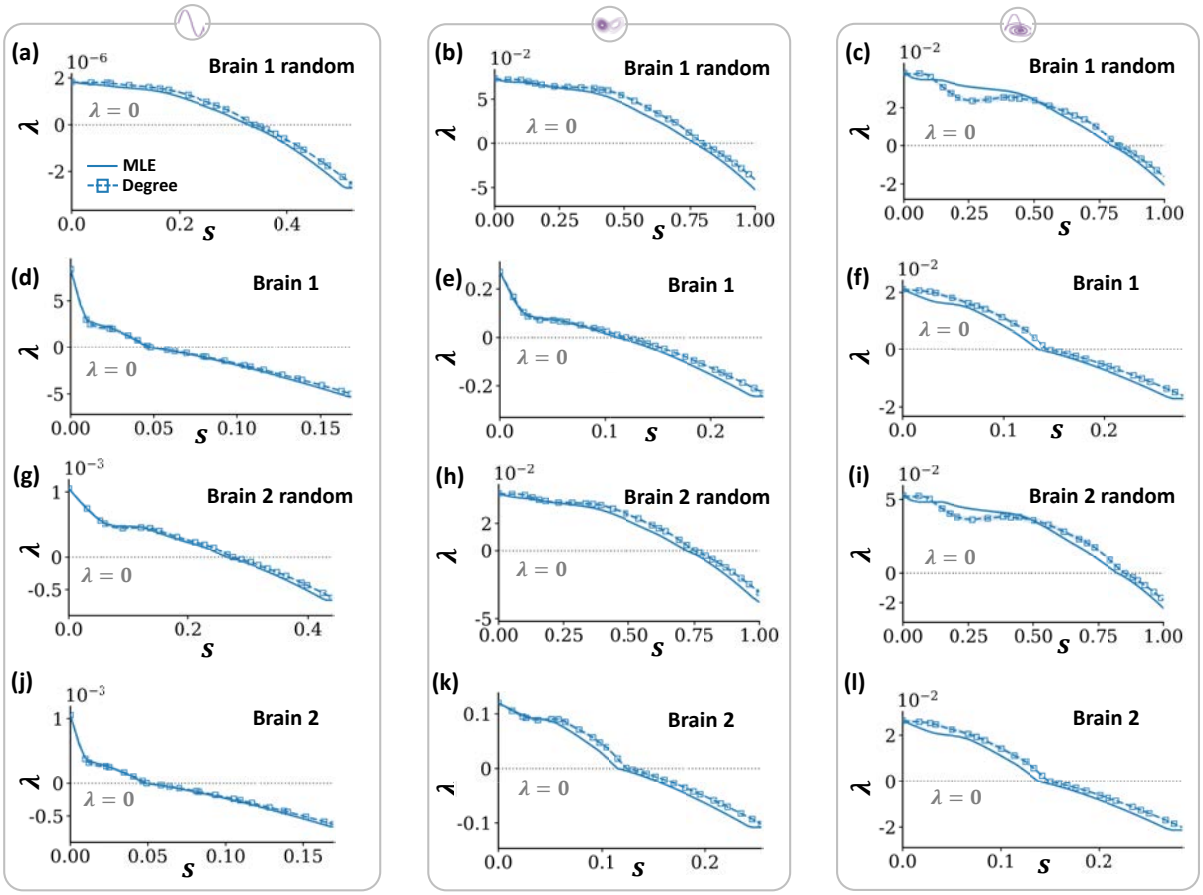

Figure 1: **Comparison between MLE-based algorithm and degree-based algorithm**. **a**, **d**, **g**, **i** λ versus the fraction of nodes s introduced into $S^{\mathrm{C}}_{\mathrm{min}}$ for different networks with Kuramoto dynamics, as obtained from the MLE-based algorithm (solid line). As nodes are added, λ decreases until it crosses the $\lambda = 0$ line (grey dashed), at which point no additional nodes are required and the minimal set is complete. We also show the results from the degree-based approach (squares), which are nearly indistinguishable. Similar results for Lorenz (**b**, **e**, **h**, **k**) and Rössler dynamics (**c**, **f**, **i**, **l**).

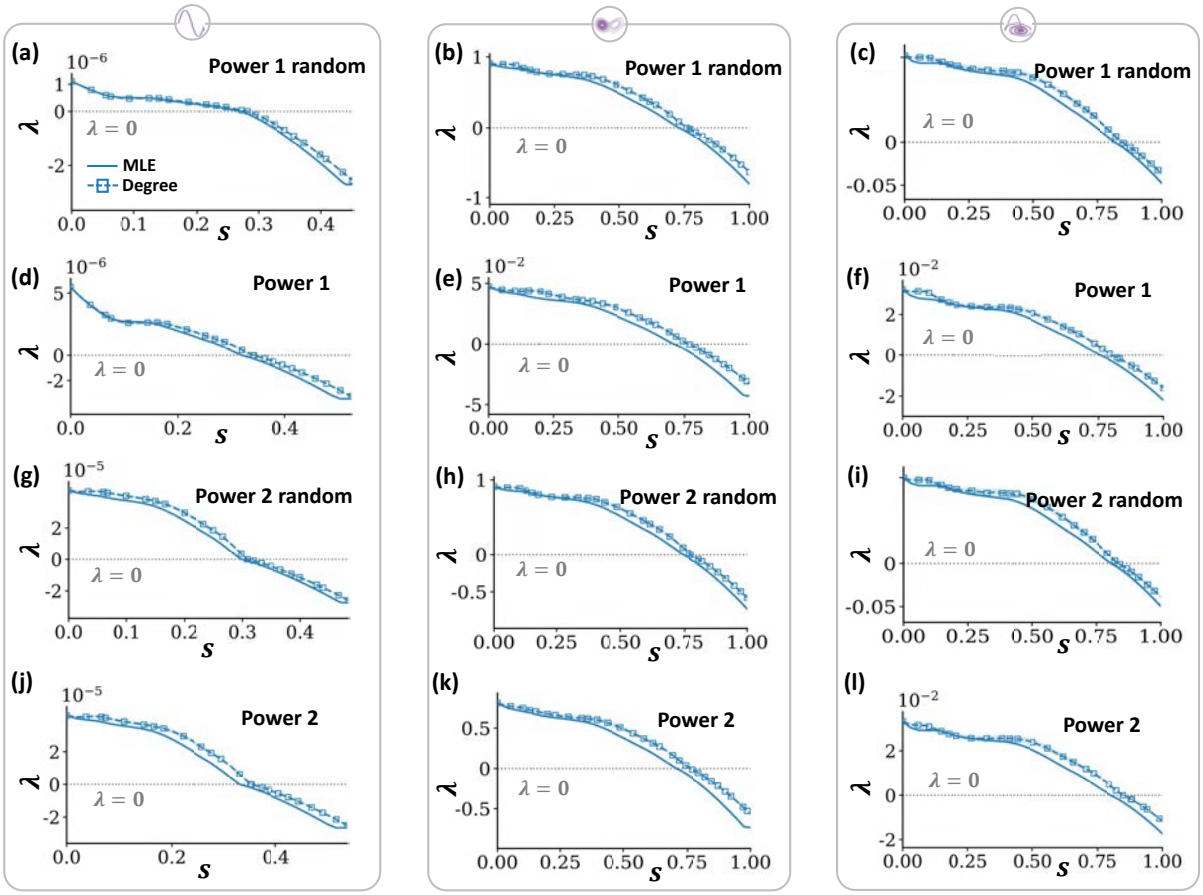

Figure 2: **Comparison between MLE-based algorithm and degree-based algorithm**. **a**, **d**, **g**, **i** λ versus the fraction of nodes s introduced into $S^{\mathrm{C}}_{\mathrm{min}}$ for different networks with Kuramoto dynamics, as obtained from the MLE-based algorithm (solid line). As nodes are added, λ decreases until it crosses the $\lambda = 0$ line (grey dashed), at which point no additional nodes are required and the minimal set is complete. We also show the results from the degree-based approach (squares), which are nearly indistinguishable. Similar results for Lorenz (**b**, **e**, **h**, **k**) and Rössler dynamics (**c**, **f**, **i**, **l**).

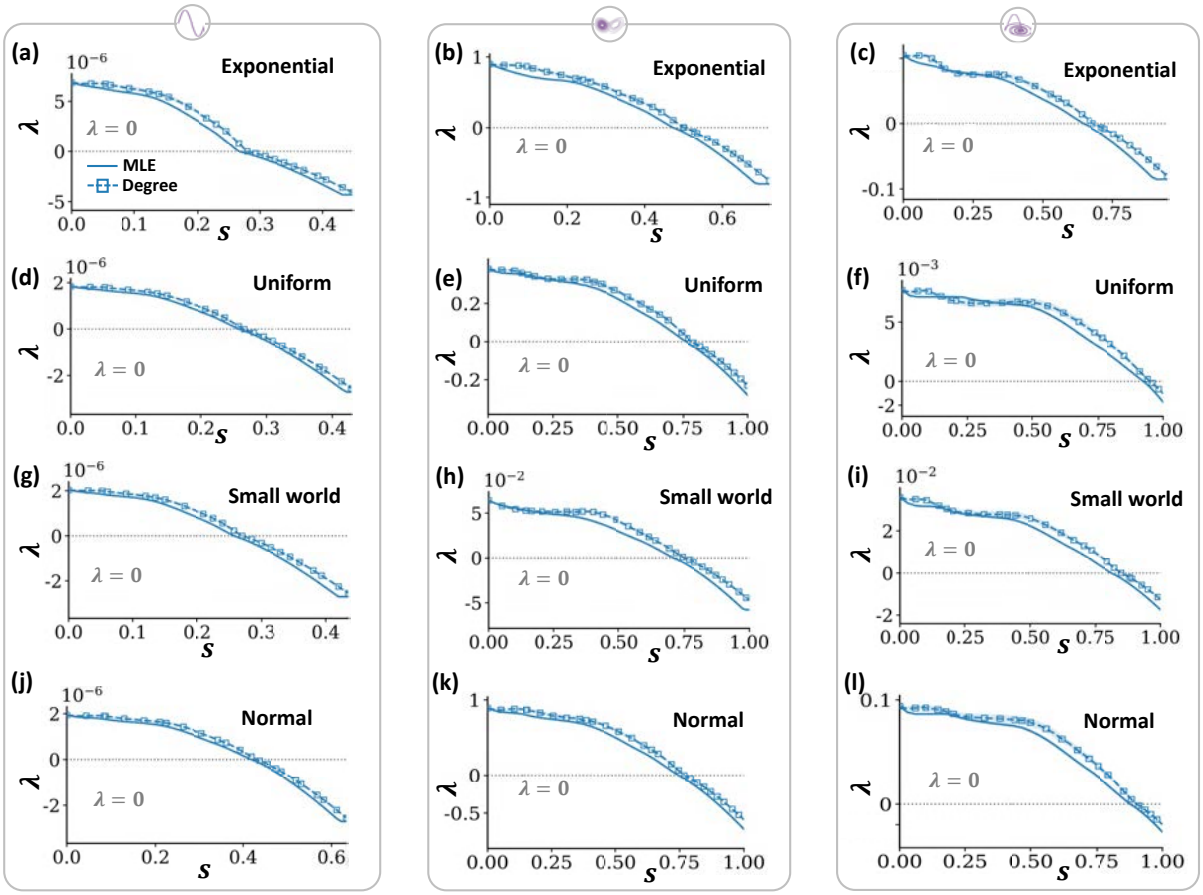

Figure 3: **Comparison between MLE-based algorithm and degree-based algorithm**. **a**, **d**, **g**, **i** λ versus the fraction of nodes s introduced into $S^{\mathrm{C}}_{\mathrm{min}}$ for different networks with Kuramoto dynamics, as obtained from the MLE-based algorithm (solid line). As nodes are added, λ decreases until it crosses the $\lambda = 0$ line (grey dashed), at which point no additional nodes are required and the minimal set is complete. We also show the results from the degree-based approach (squares), which are nearly indistinguishable. Similar results for Lorenz (**b**, **e**, **h**, **k**) and Rössler dynamics (**c**, **f**, **i**, **l**).

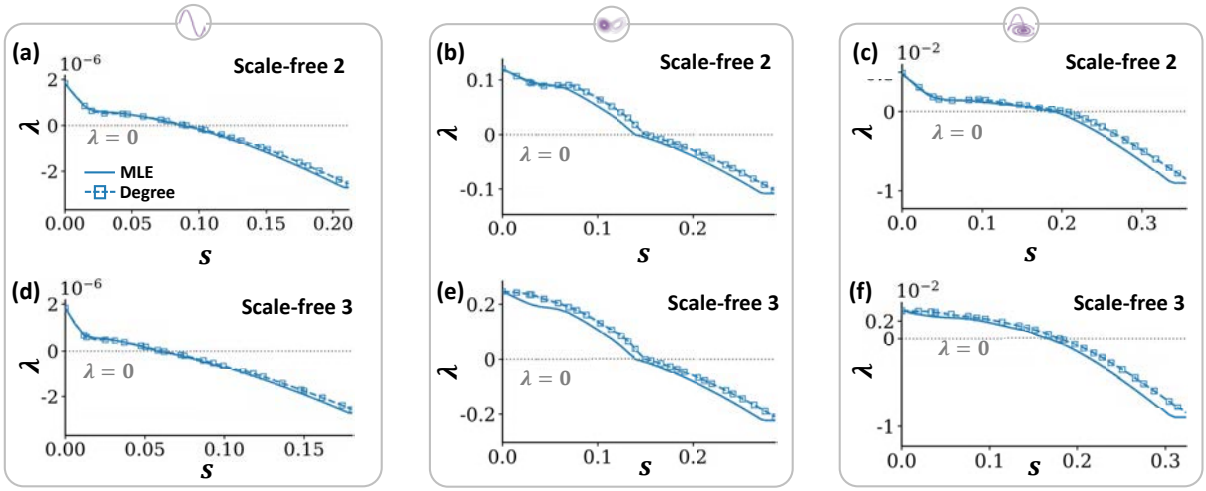

Figure 4: **Comparison between MLE-based algorithm and degree-based algorithm**. **a**, **d** λ versus the fraction of nodes s introduced into $S^{\mathrm{C}}_{\mathrm{min}}$ for different networks with Kuramoto dynamics, as obtained from the MLE-based algorithm (solid line). As nodes are added, λ decreases until it crosses the $\lambda = 0$ line (grey dashed), at which point no additional nodes are required and the minimal set is complete. We also show the results from the degree-based approach (squares), which are nearly indistinguishable. Similar results for Lorenz (**b**, **e**) and Rössler dynamics (**c**, **f**).

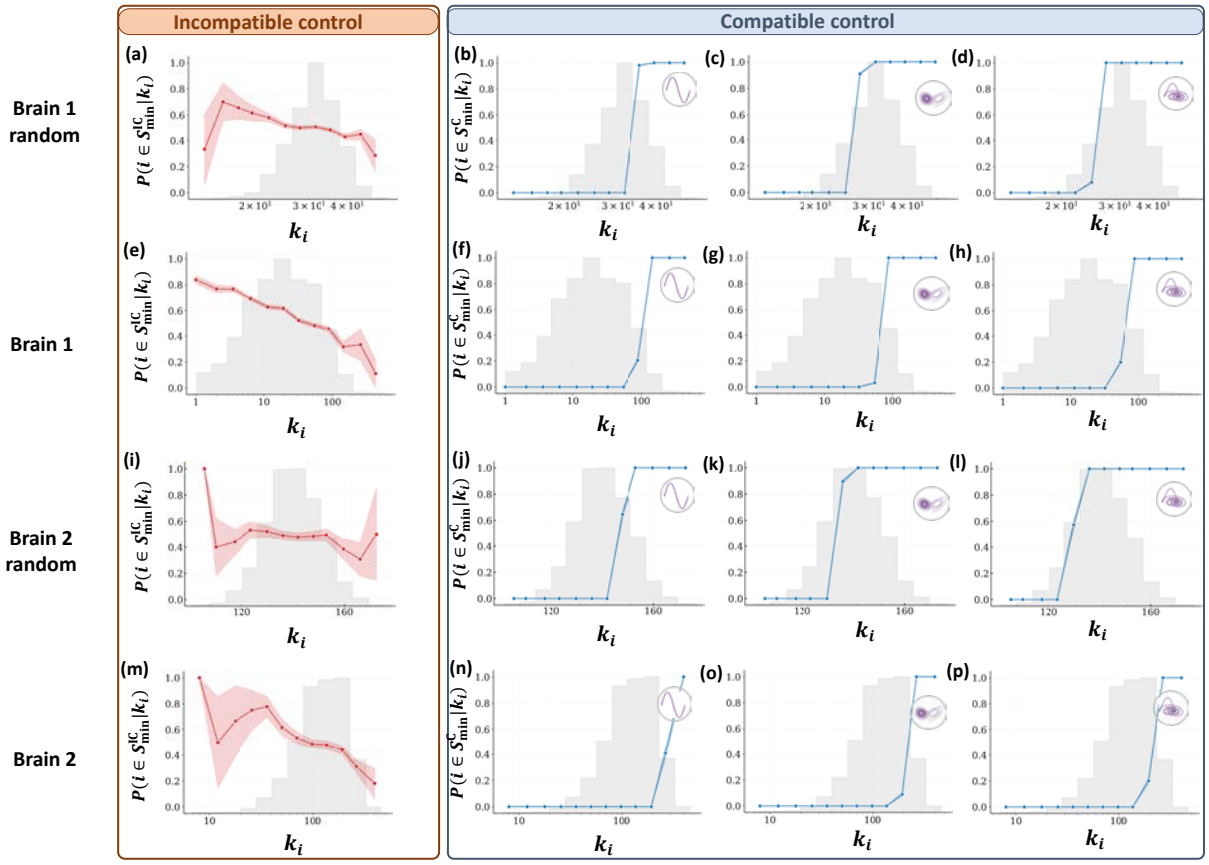

Figure 5: **Conditional probability versus node degree for different networks**. In the left panel (**a**, **e**, **i**, **m**), the conditional probability $P(i \in S^{\mathrm{IC}}_{\mathrm{min}} \mid k_i)$ versus k_i (red). The probability declines with k_i, corroborating that incompatible control favors low-degree nodes and tends to avoid hubs. The degree distribution $P(k)$ is also shown (grey). In the right panel (all the other subfigures), the conditional probability $P(i \in S^{\mathrm{C}}_{\mathrm{min}} \mid k_i)$ versus k_i for the three dynamics (blue). In contrast to incompatible control, here the probability shows a sharp preference for high-degree nodes.

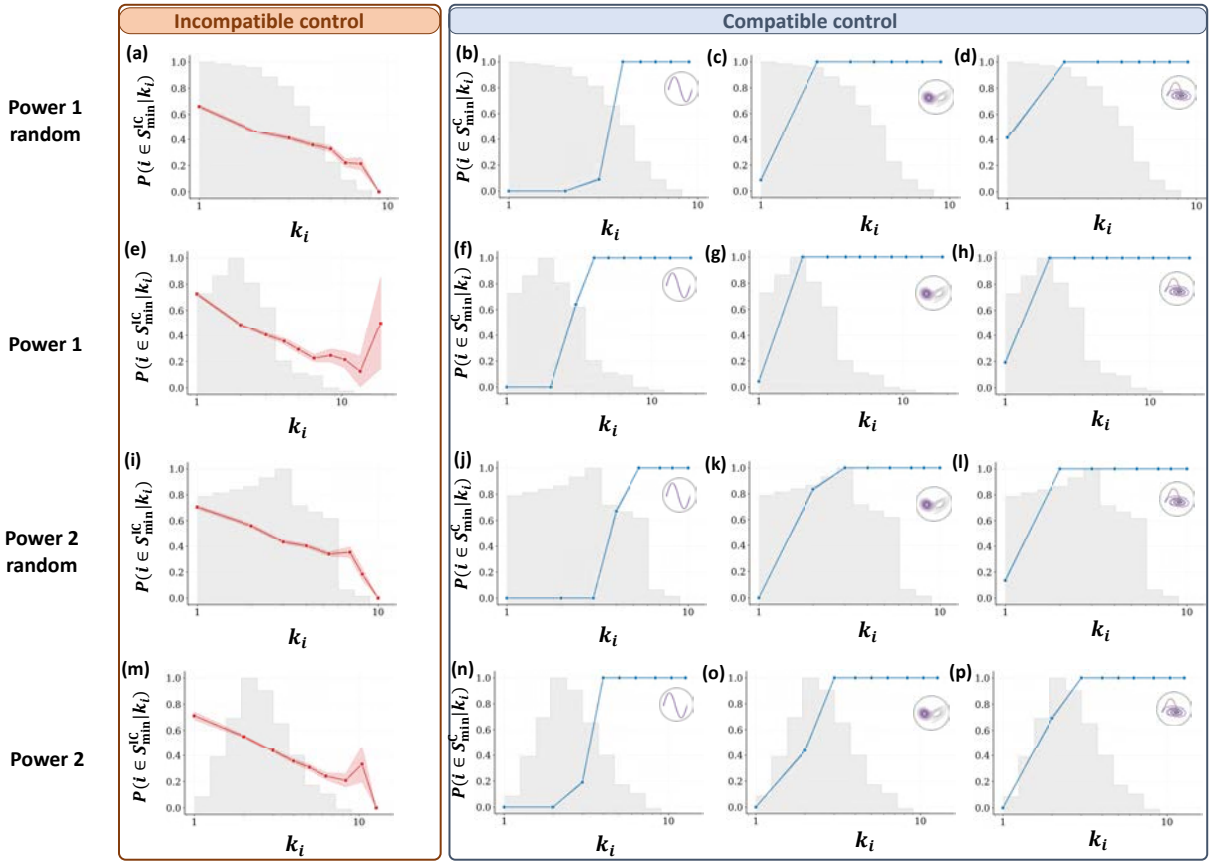

Figure 6: **Conditional probability versus node degree for different networks**. In the left panel (**a**, **e**, **i**, **m**), the conditional probability $P(i \in S^{\mathrm{IC}}_{\mathrm{min}} \mid k_i)$ versus k_i (red). The probability declines with k_i, corroborating that incompatible control favors low-degree nodes and tends to avoid hubs. The degree distribution $P(k)$ is also shown (grey). In the right panel (all the other subfigures), the conditional probability $P(i \in S^{\mathrm{C}}_{\mathrm{min}} \mid k_i)$ versus k_i for the three dynamics (blue). In contrast to incompatible control, here the probability shows a sharp preference for high-degree nodes.

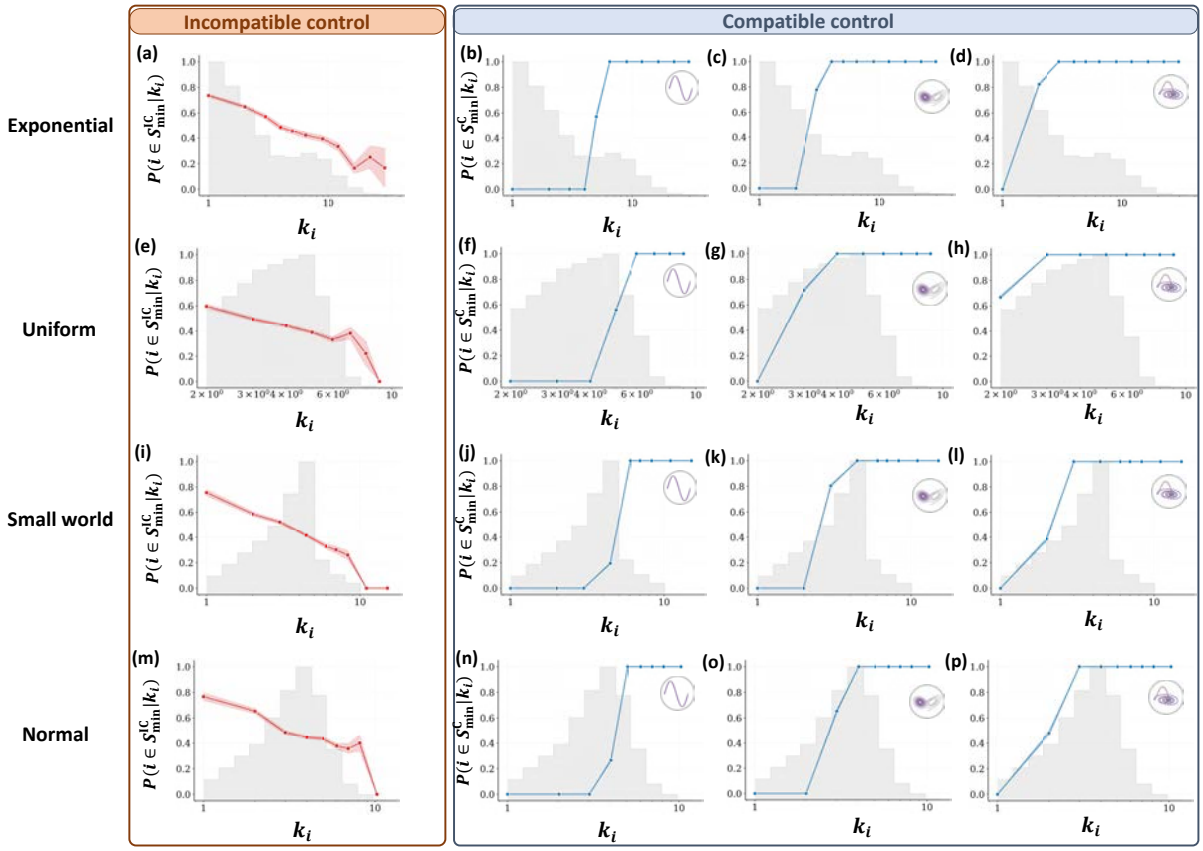

Figure 7: **Conditional probability versus node degree for different networks**. In the left panel (**a**, **e**, **i**, **m**), the conditional probability $P(i \in S^{\mathrm{IC}}_{\mathrm{min}} \mid k_i)$ versus k_i (red). The probability declines with k_i, corroborating that incompatible control favors low-degree nodes and tends to avoid hubs. The degree distribution $P(k)$ is also shown (grey). In the right panel (all the other subfigures), the conditional probability $P(i \in S^{\mathrm{C}}_{\mathrm{min}} \mid k_i)$ versus k_i for the three dynamics (blue). In contrast to incompatible control, here the probability shows a sharp preference for high-degree nodes.

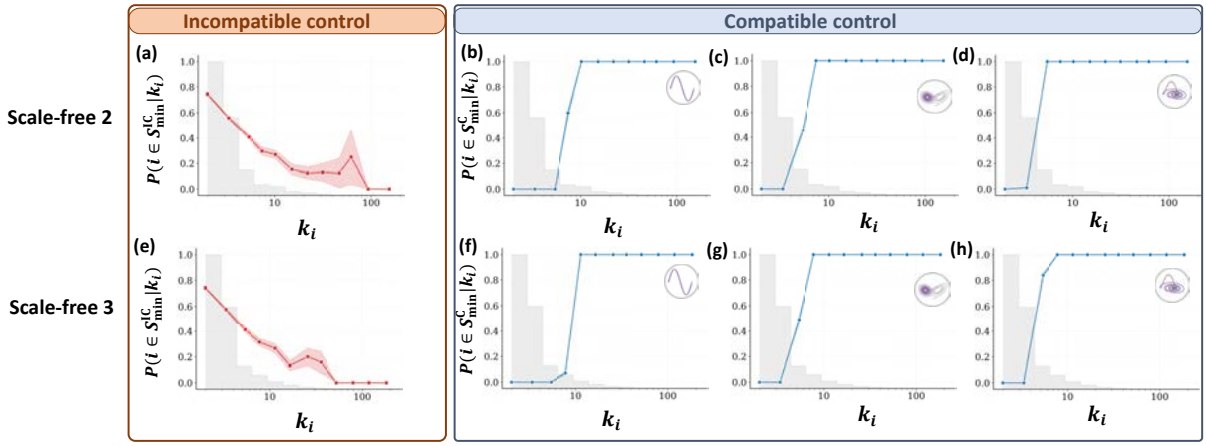

Figure 8: **Conditional probability versus node degree for different networks**. In the left panel (**a**, **e**), the conditional probability $P(i \in S^{\mathrm{IC}}_{\mathrm{min}} \mid k_i)$ versus k_i (red). The probability declines with k_i, corroborating that incompatible control favors low-degree nodes and tends to avoid hubs. The degree distribution $P(k)$ is also shown (grey). In the right panel (all the other subfigures), the conditional probability $P(i \in S^{\mathrm{C}}_{\mathrm{min}} \mid k_i)$ versus k_i for the three dynamics (blue). In contrast to incompatible control, here the probability shows a sharp preference for high-degree nodes.

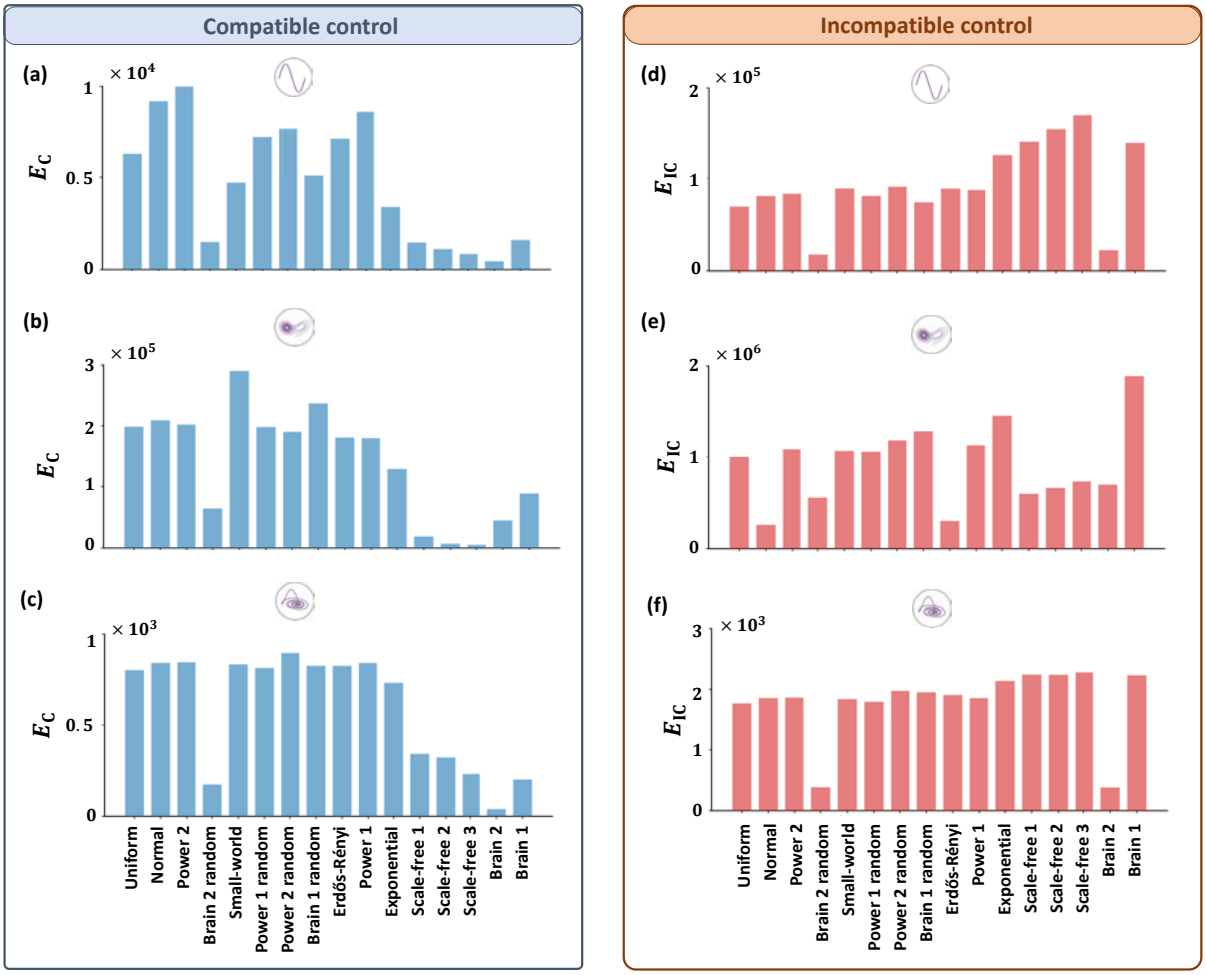

Figure 9: **Reaching energy under compatible and incompatible control.** **a**–**c** Reaching energy over the 0–20 s interval for all 16 networks shown in the main text. Networks are arranged from least to most heterogeneous for the case of compatible control. The three panels correspond to the Kuramoto, Lorenz, and Rössler dynamics, respectively. **d**–**f** Similar results for the case of incompatible control.

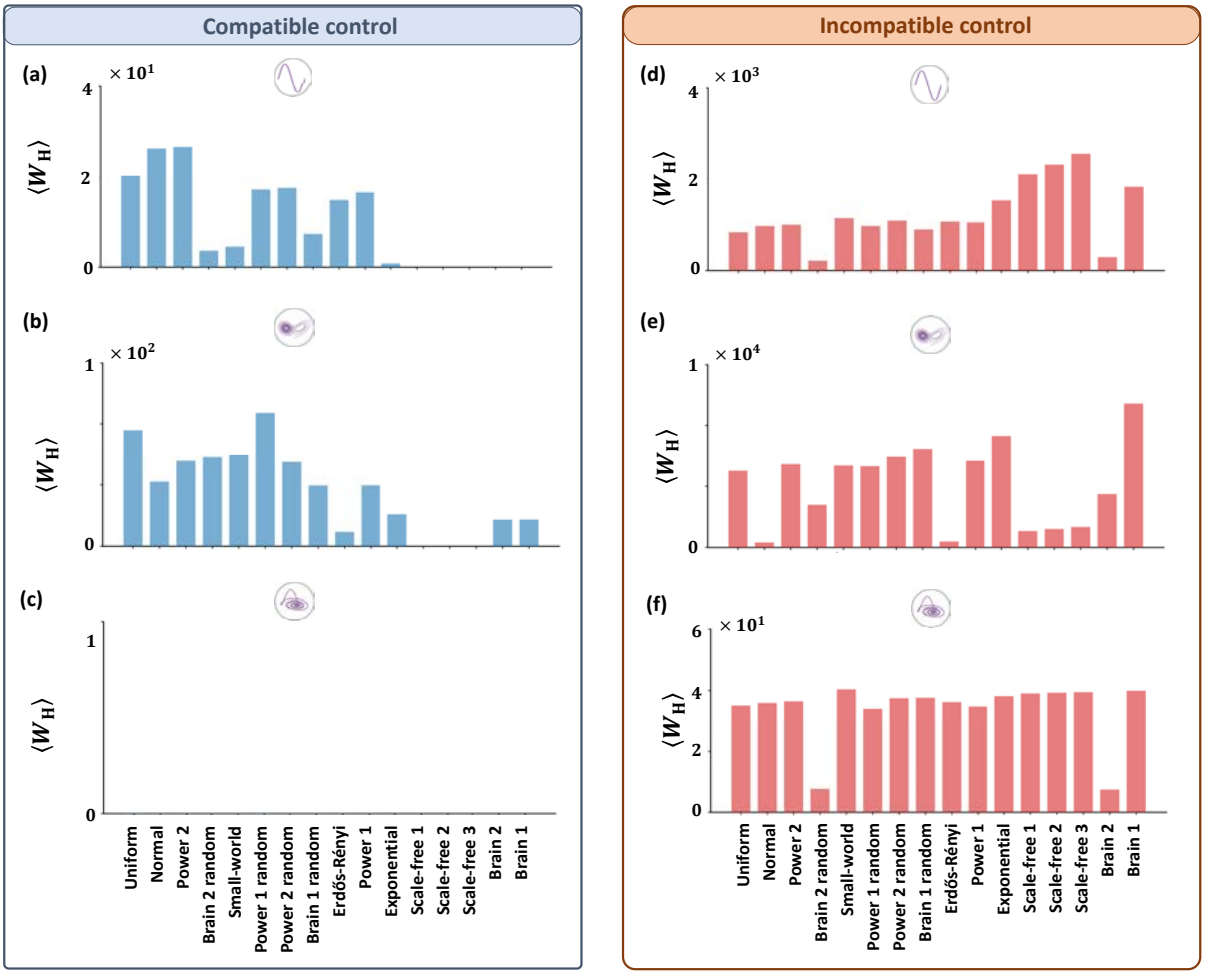

Figure 10: **Average holding power under compatible and incompatible control.** **a**–**c** Average holding power over the 20–25 s interval for all 16 networks shown in the main text. Networks are arranged from least to most heterogeneous for the case of compatible control. The three panels correspond to the Kuramoto, Lorenz, and Rössler dynamics, respectively. **d**–**f** Similar results for the case of incompatible control.

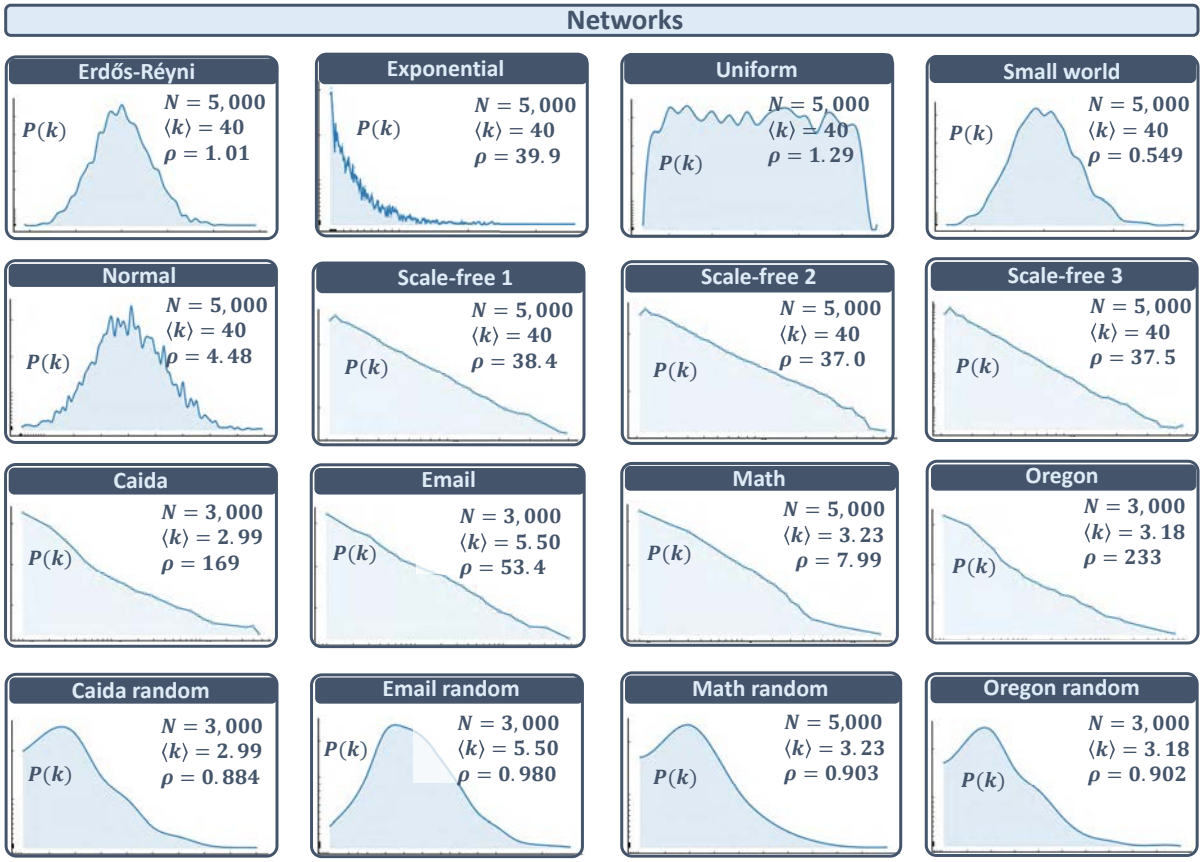

Figure 11: **Additional testing ground for functional robustness**. We consider 16 synthetic (more denser) and other real-world networks. For each network, we report its size N, average degree $\langle k \rangle$, and heterogeneity score ρ defined in (8) in the main text. The degree distribution $P(k)$ is also shown (blue).

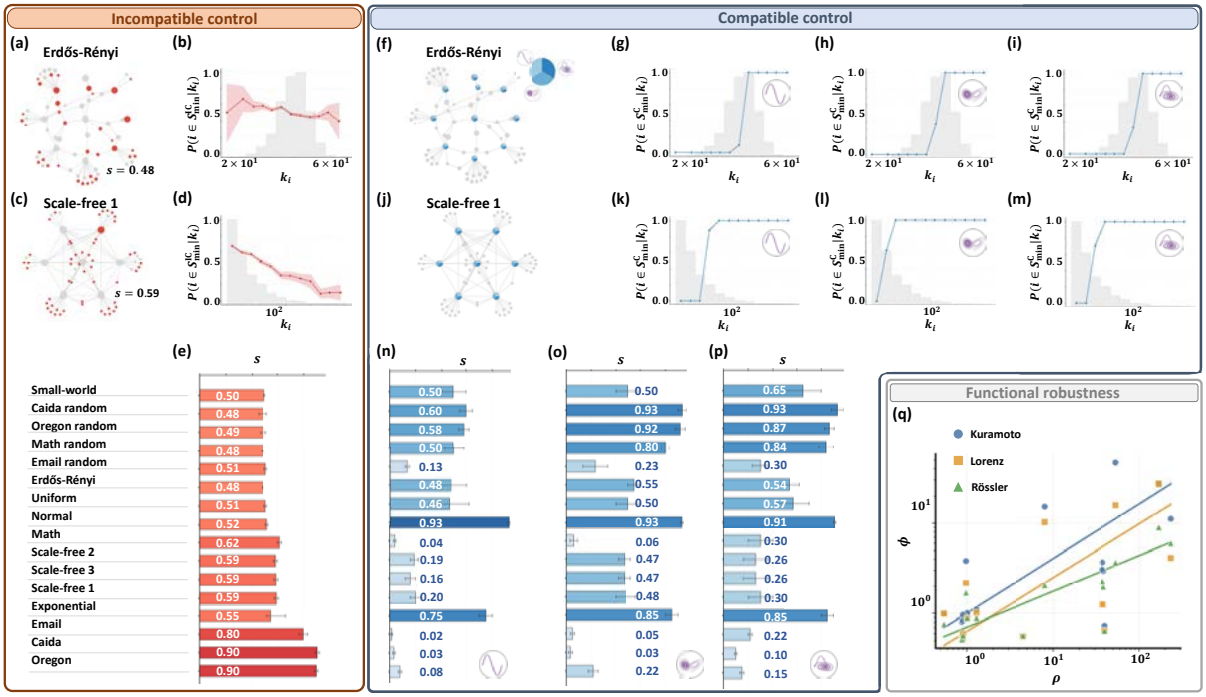

Figure 12: **The minimal driver node set under compatible and incompatible control**. **a** The minimal driver node set $S_{\min}^{\mathrm{IC}}$ obtained for the Erdős-Rényi network shown in Fig. 11 under incompatible control (red nodes; shown is a sample extracted from the full network). The fraction of nodes in $S_{\min}^{\mathrm{IC}}$ is $s = 0.48$, and their locations at the network periphery indicate a preference for the low-degree nodes. **b** The conditional probability $P(i \in S_{\min}^{\mathrm{IC}} \mid k_i)$ versus k_i (red). The probability declines with k_i, corroborating that incompatible control favors low-degree nodes and tends to avoid hubs. The degree distribution $P(k)$ is also shown (grey). **c** Similar results for the Scale-free 1 network shown in Fig. 11, this time with $s = 0.59$. **d** Here too, $S_{\min}^{\mathrm{IC}}$ systematically avoids the hubs. **e** The fraction of driver nodes s for all 16 networks shown in Fig. 11. Networks are arranged from least to most heterogeneous, showing that s increases with degree heterogeneity. This is consistent with the well-known result that heterogeneous networks are difficult to control, here reaffirmed specifically for the case of incompatible control. **f** $S_{\min}^{\mathrm{C}}$ for the Erdős-Rényi network under compatible control, as obtained for Kuramoto (light blue), Lorenz (blue), and Rössler (dark blue) dynamics. The driver nodes tend to condense near the center, indicating that $S_{\min}^{\mathrm{C}}$ is composed primarily of hubs. **g-i** The conditional probability $P(i \in S_{\min}^{\mathrm{C}} \mid k_i)$ versus k_i for the three dynamics (blue). In contrast to incompatible control, here the probability shows a sharp preference for high-degree nodes. **j-m** Similar results for Scale-free 1. **n** The fraction of driver nodes s under compatible control across all networks, shown here for Kuramoto dynamics. Here, heterogeneity facilitates control, and hence more heterogeneous networks (bottom) exhibit smaller s. **o-p** We observe a similar trend also for Lorenz and Rössler dynamics. **q** Functional robustness ϕ in (7) in the main text versus network heterogeneity ρ, as obtained from Kuramoto (blue), Lorenz (orange) and Rössler (green) dynamics, implemented on each of 16 networks. A clear trend emerges, whereby heterogeneous networks exhibit enhanced functional robustness. **These results provide further support for the conclusions drawn in the main text.**

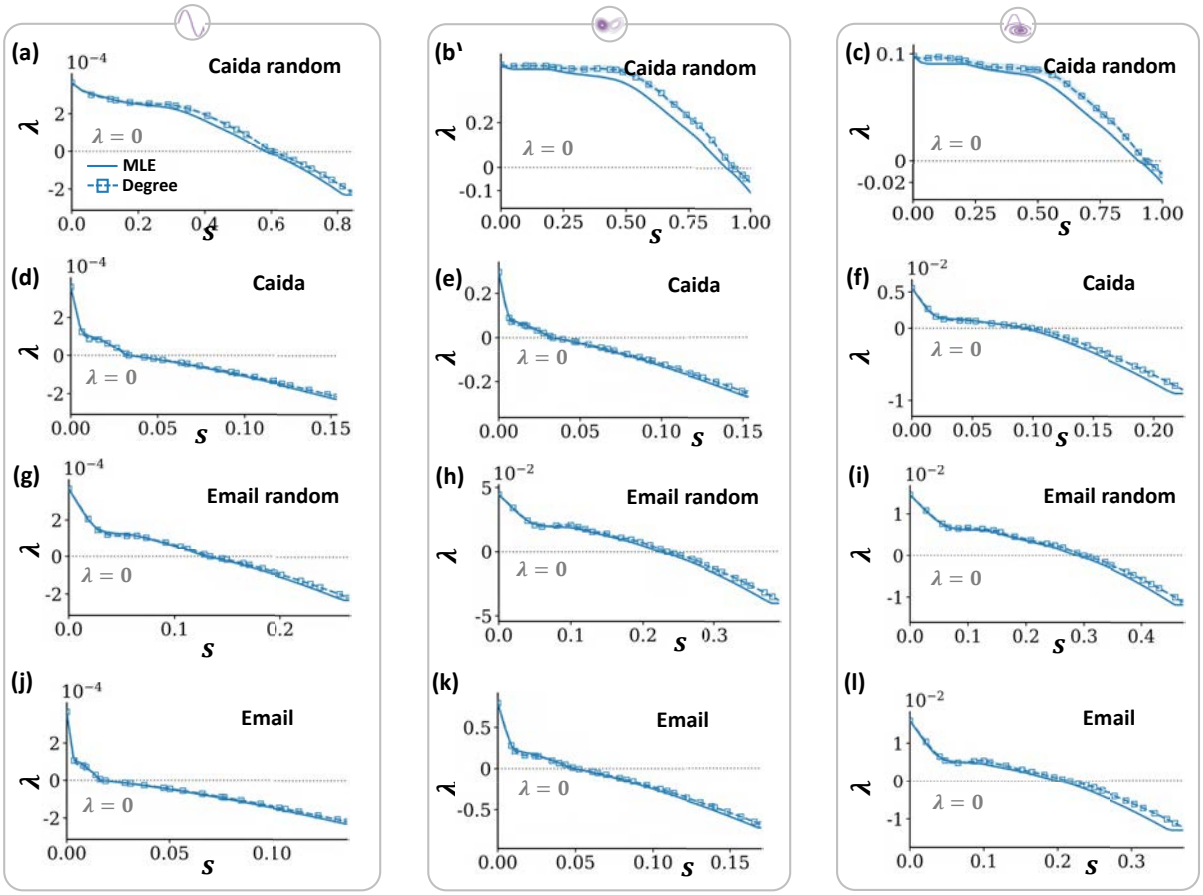

Figure 13: **Comparison between MLE-based algorithm and degree-based algorithm**. **a**, **d**, **g**, **i** λ versus the fraction of nodes s introduced into $S^{C}_{\min}$ for different networks in Fig. 11 with Kuramoto dynamics, as obtained from the MLE-based algorithm (solid line). As nodes are added, λ decreases until it crosses the $\lambda = 0$ line (grey dashed), at which point no additional nodes are required and the minimal set is complete. We also show the results from the degree-based approach (squares), which are nearly indistinguishable. Similar results for Lorenz (**b**, **e**, **h**, **k**) and Rössler dynamics (**c**, **f**, **i**, **l**).

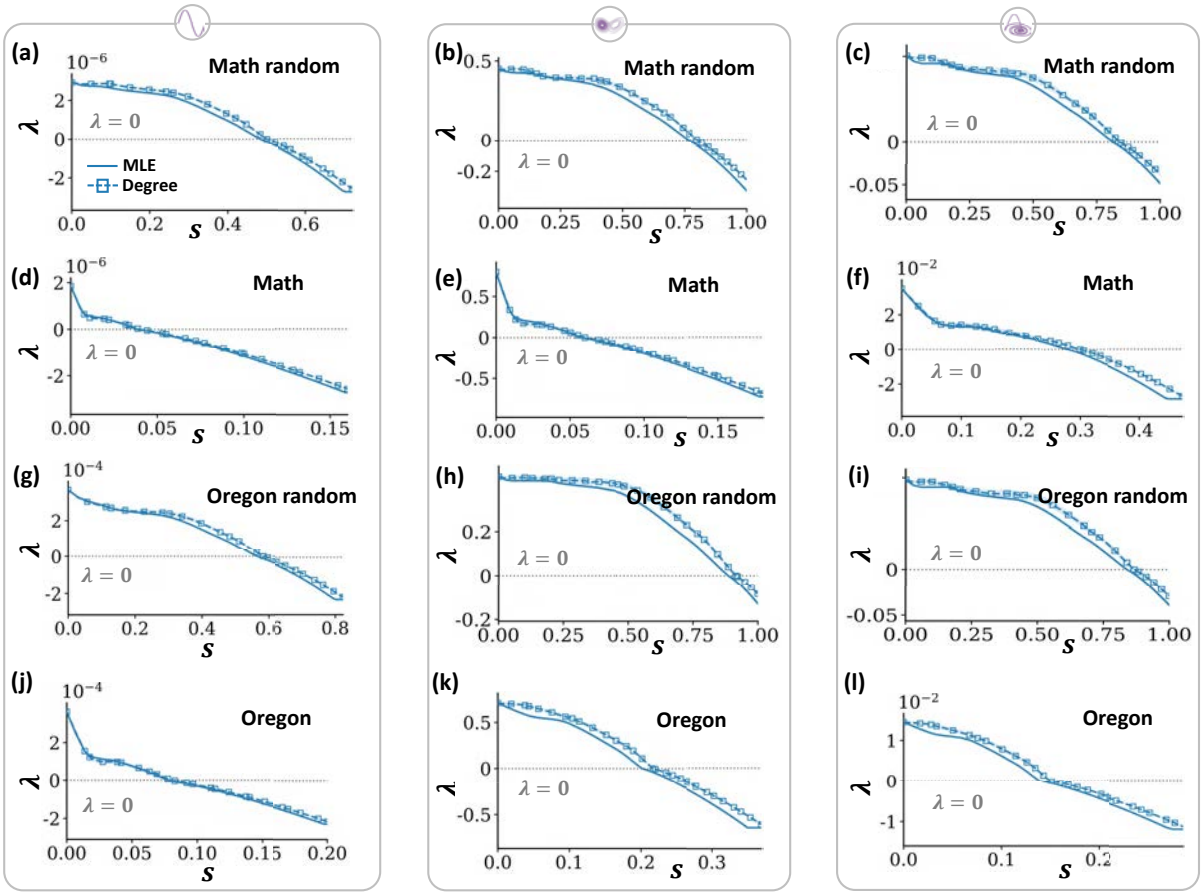

Figure 14: **Comparison between MLE-based algorithm and degree-based algorithm**. **a**, **d**, **g**, **i** λ versus the fraction of nodes s introduced into $S^{\mathrm{C}}_{\mathrm{min}}$ for different networks in Fig. 11 with Kuramoto dynamics, as obtained from the MLE-based algorithm (solid line). As nodes are added, λ decreases until it crosses the $\lambda = 0$ line (grey dashed), at which point no additional nodes are required and the minimal set is complete. We also show the results from the degree-based approach (squares), which are nearly indistinguishable. Similar results for Lorenz (**b**, **e**, **h**, **k**) and Rössler dynamics (**c**, **f**, **i**, **l**).

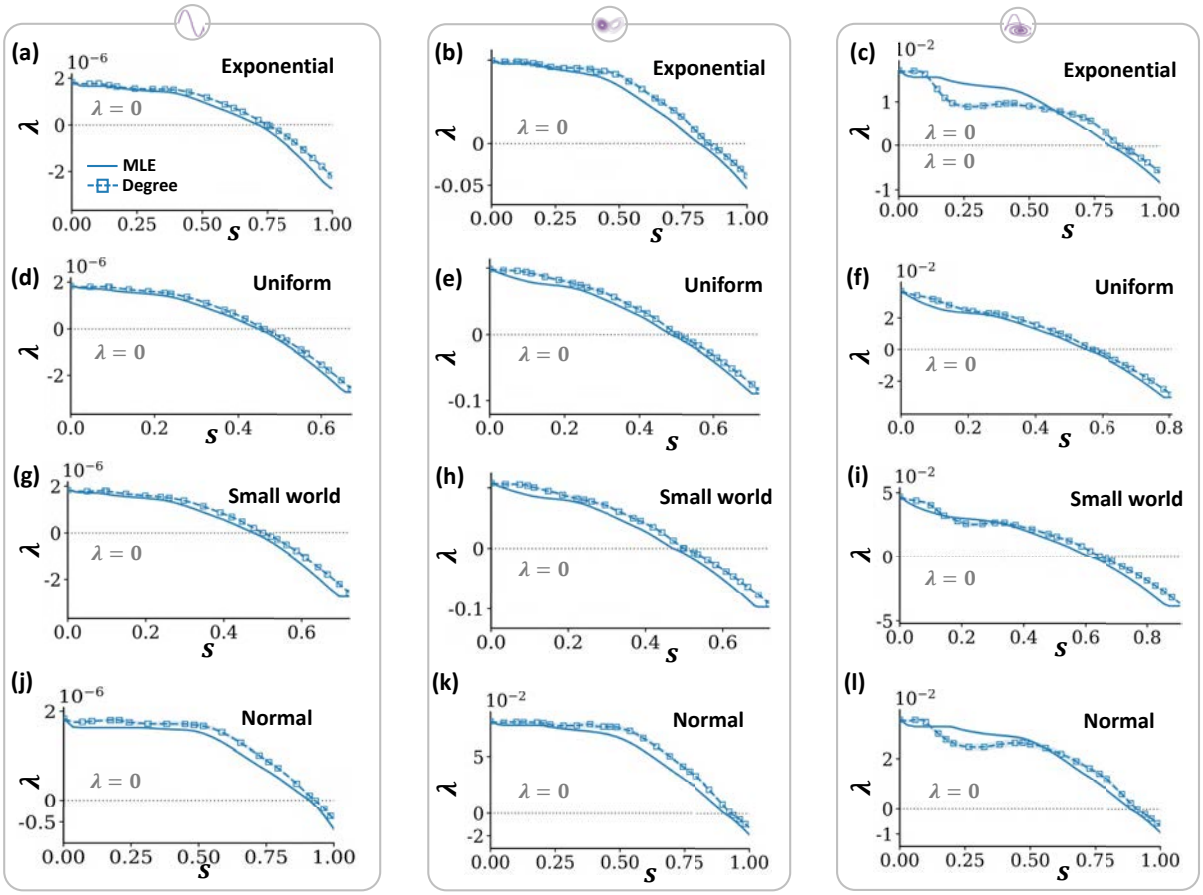

Figure 15: **Comparison between MLE-based algorithm and degree-based algorithm**. **a**, **d**, **g**, **i** λ versus the fraction of nodes s introduced into $S^{\mathrm{C}}_{\mathrm{min}}$ for different networks in Fig. 11 with Kuramoto dynamics, as obtained from the MLE-based algorithm (solid line). As nodes are added, λ decreases until it crosses the $\lambda = 0$ line (grey dashed), at which point no additional nodes are required and the minimal set is complete. We also show the results from the degree-based approach (squares), which are nearly indistinguishable. Similar results for Lorenz (**b**, **e**, **h**, **k**) and Rössler dynamics (**c**, **f**, **i**, **l**).

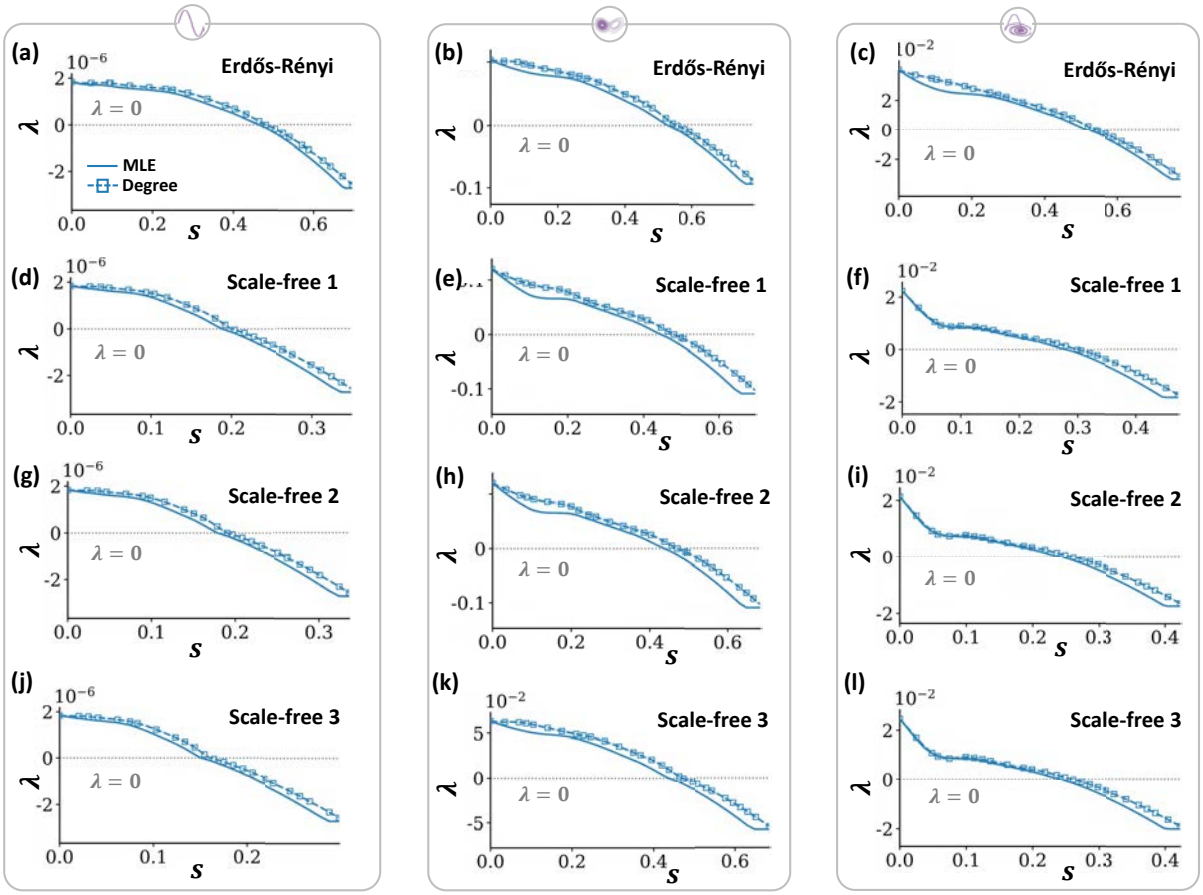

Figure 16: **Comparison between MLE-based algorithm and degree-based algorithm**. **a**, **d**, **g**, **i** λ versus the fraction of nodes s introduced into $S^{\mathrm{C}}_{\mathrm{min}}$ for different networks in Fig. 11 with Kuramoto dynamics, as obtained from the MLE-based algorithm (solid line). As nodes are added, λ decreases until it crosses the $\lambda = 0$ line (grey dashed), at which point no additional nodes are required and the minimal set is complete. We also show the results from the degree-based approach (squares), which are nearly indistinguishable. Similar results for Lorenz (**b**, **e**, **h**, **k**) and Rössler dynamics (**c**, **f**, **i**, **l**).

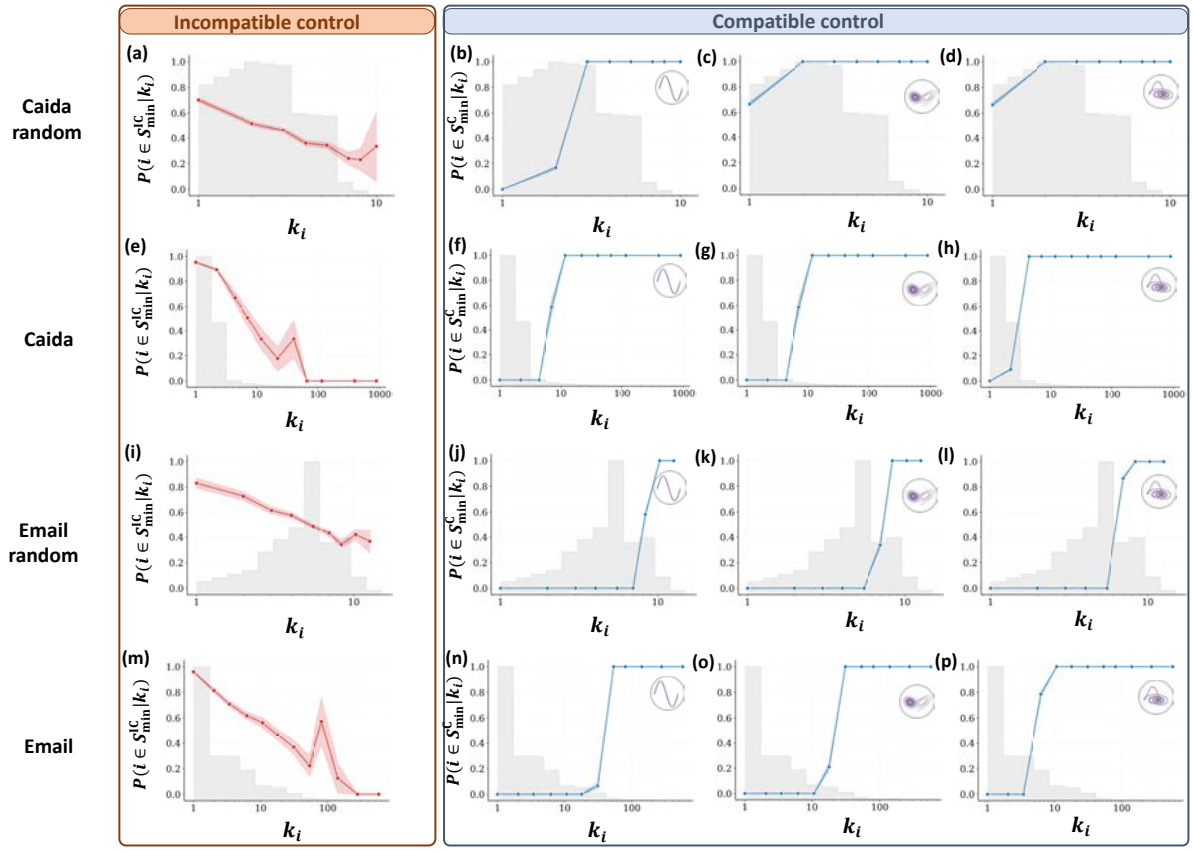

Figure 17: **Conditional probability versus node degree for different networks in Fig. 11**. In the left panel (**a**, **e**, **i**, **m**), the conditional probability $P(i \in S^{\mathrm{IC}}_{\mathrm{min}} \mid k_i)$ versus k_i (red). The probability declines with k_i, corroborating that incompatible control favors low-degree nodes and tends to avoid hubs. The degree distribution $P(k)$ is also shown (grey). In the right panel (all the other subfigures), the conditional probability $P(i \in S^{\mathrm{C}}_{\mathrm{min}} \mid k_i)$ versus k_i for the three dynamics (blue). In contrast to incompatible control, here the probability shows a sharp preference for high-degree nodes.

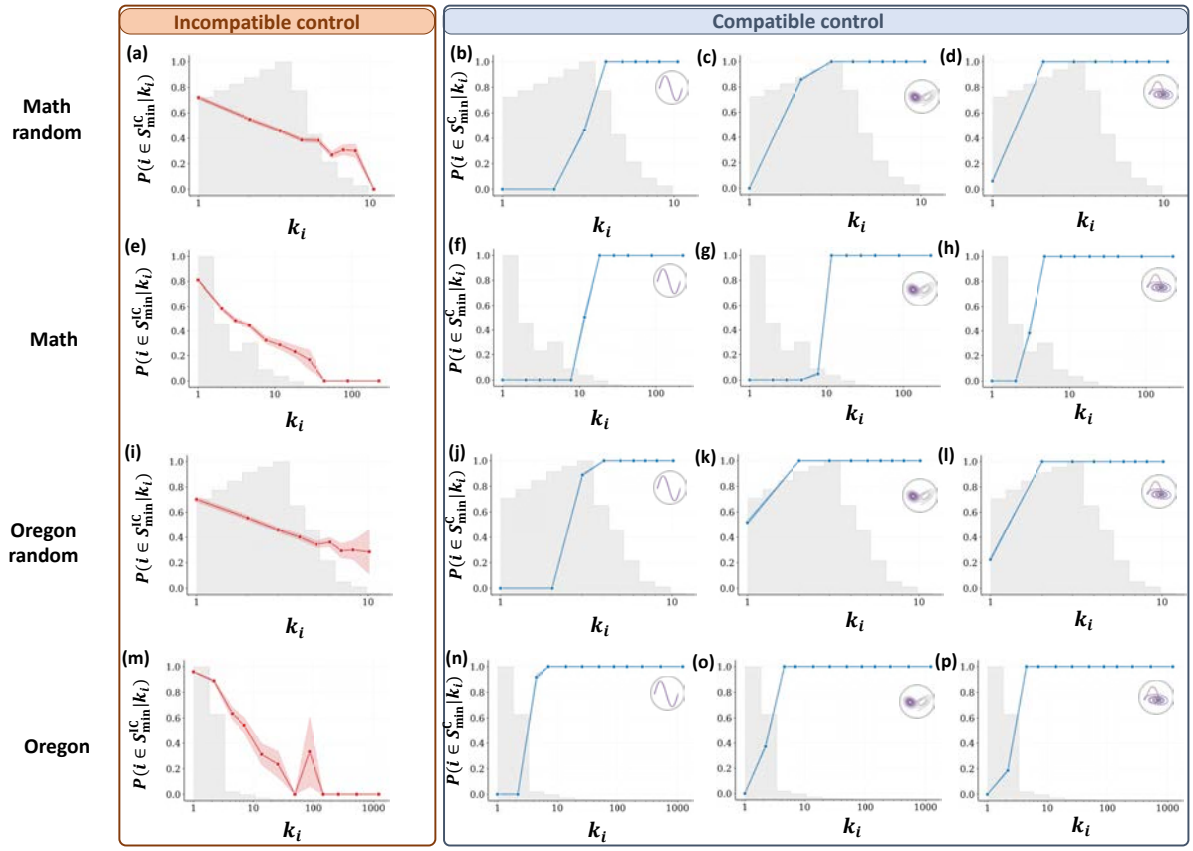

Figure 18: **Conditional probability versus node degree for different networks in Fig. 11**. In the left panel (**a**, **e**, **i**, **m**), the conditional probability $P(i \in S^{\mathrm{IC}}_{\mathrm{min}} \mid k_i)$ versus k_i (red). The probability declines with k_i, corroborating that incompatible control favors low-degree nodes and tends to avoid hubs. The degree distribution $P(k)$ is also shown (grey). In the right panel (all the other subfigures), the conditional probability $P(i \in S^{\mathrm{C}}_{\mathrm{min}} \mid k_i)$ versus k_i for the three dynamics (blue). In contrast to incompatible control, here the probability shows a sharp preference for high-degree nodes.

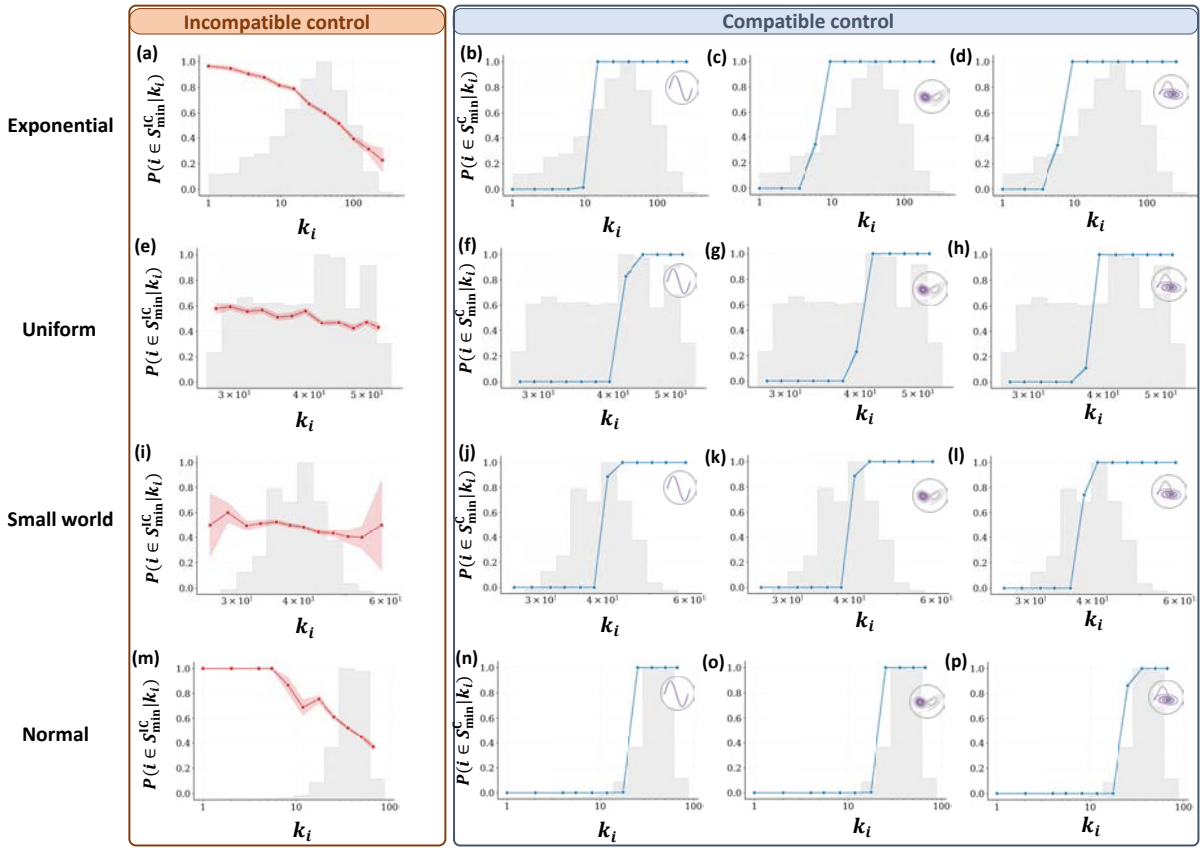

Figure 19: **Conditional probability versus node degree for different networks in Fig. 11**. In the left panel (**a**, **e**, **i**, **m**), the conditional probability $P(i \in S_{\min}^{\mathrm{IC}} \mid k_i)$ versus k_i (red). The probability declines with k_i, corroborating that incompatible control favors low-degree nodes and tends to avoid hubs. The degree distribution $P(k)$ is also shown (grey). In the right panel (all the other subfigures), the conditional probability $P(i \in S_{\min}^{\mathrm{C}} \mid k_i)$ versus k_i for the three dynamics (blue). In contrast to incompatible control, here the probability shows a sharp preference for high-degree nodes.

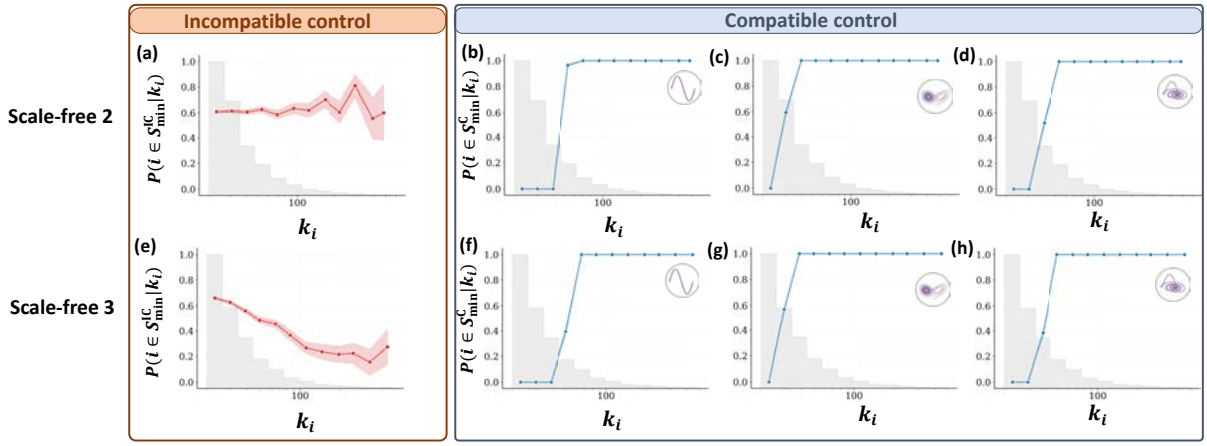

Figure 20: **Conditional probability versus node degree for different networks in Fig. 11**. In the left panel (**a**, **e**), the conditional probability $P(i \in S^{\mathrm{IC}}_{\mathrm{min}} \mid k_i)$ versus k_i (red). The probability declines with k_i, corroborating that incompatible control favors low-degree nodes and tends to avoid hubs. The degree distribution $P(k)$ is also shown (grey). In the right panel (all the other subfigures), the conditional probability $P(i \in S^{\mathrm{C}}_{\mathrm{min}} \mid k_i)$ versus k_i for the three dynamics (blue). In contrast to incompatible control, here the probability shows a sharp preference for high-degree nodes.

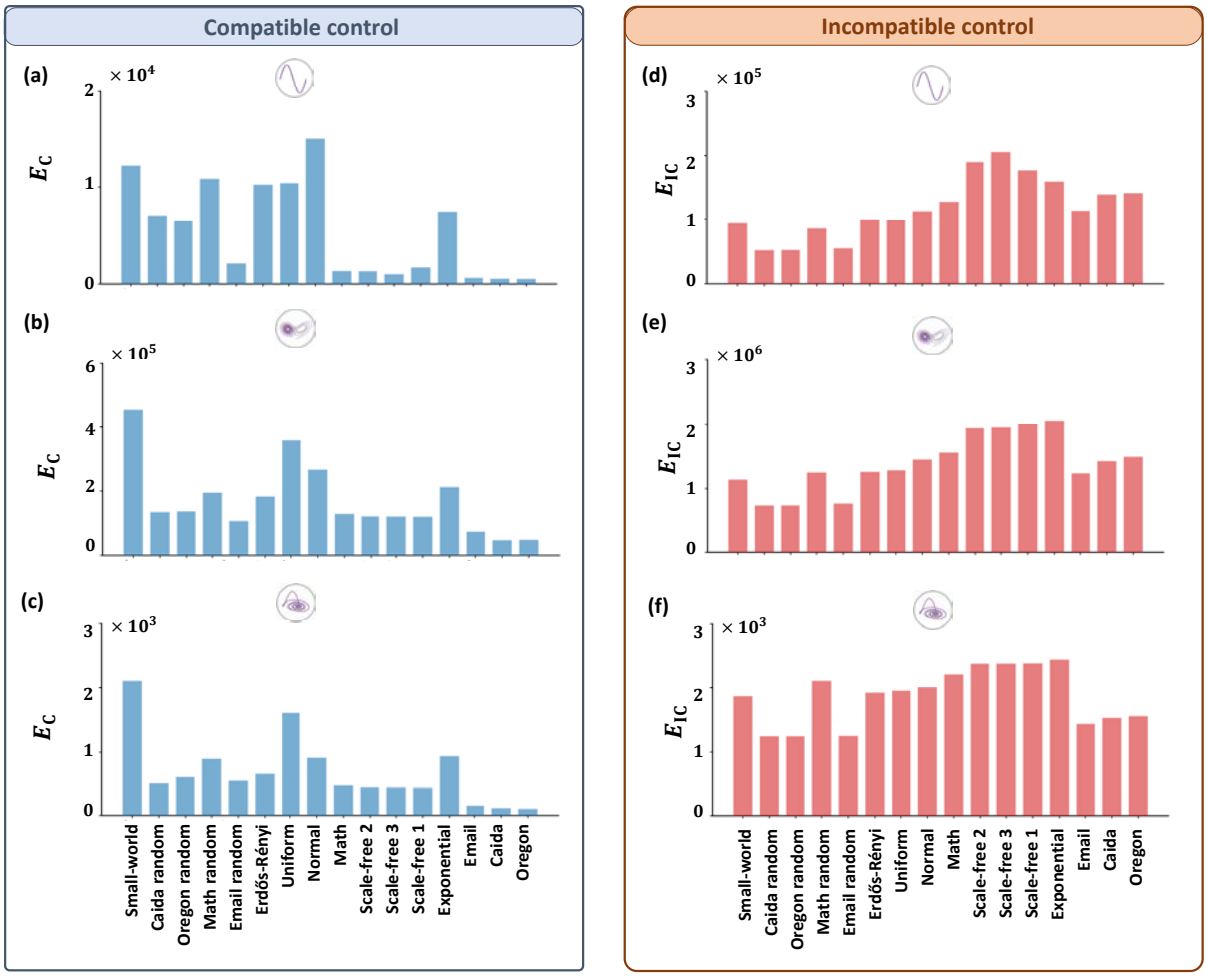

Figure 21: **Reaching energy under compatible and incompatible control.** **a**–**c** Reaching energy over the 0–20 s interval for all 16 networks shown in Fig. 11. Networks are arranged from least to most heterogeneous for the case of compatible control. The three panels correspond to the Kuramoto, Lorenz, and Rössler dynamics, respectively. **d**–**f** Similar results for the case of incompatible control.

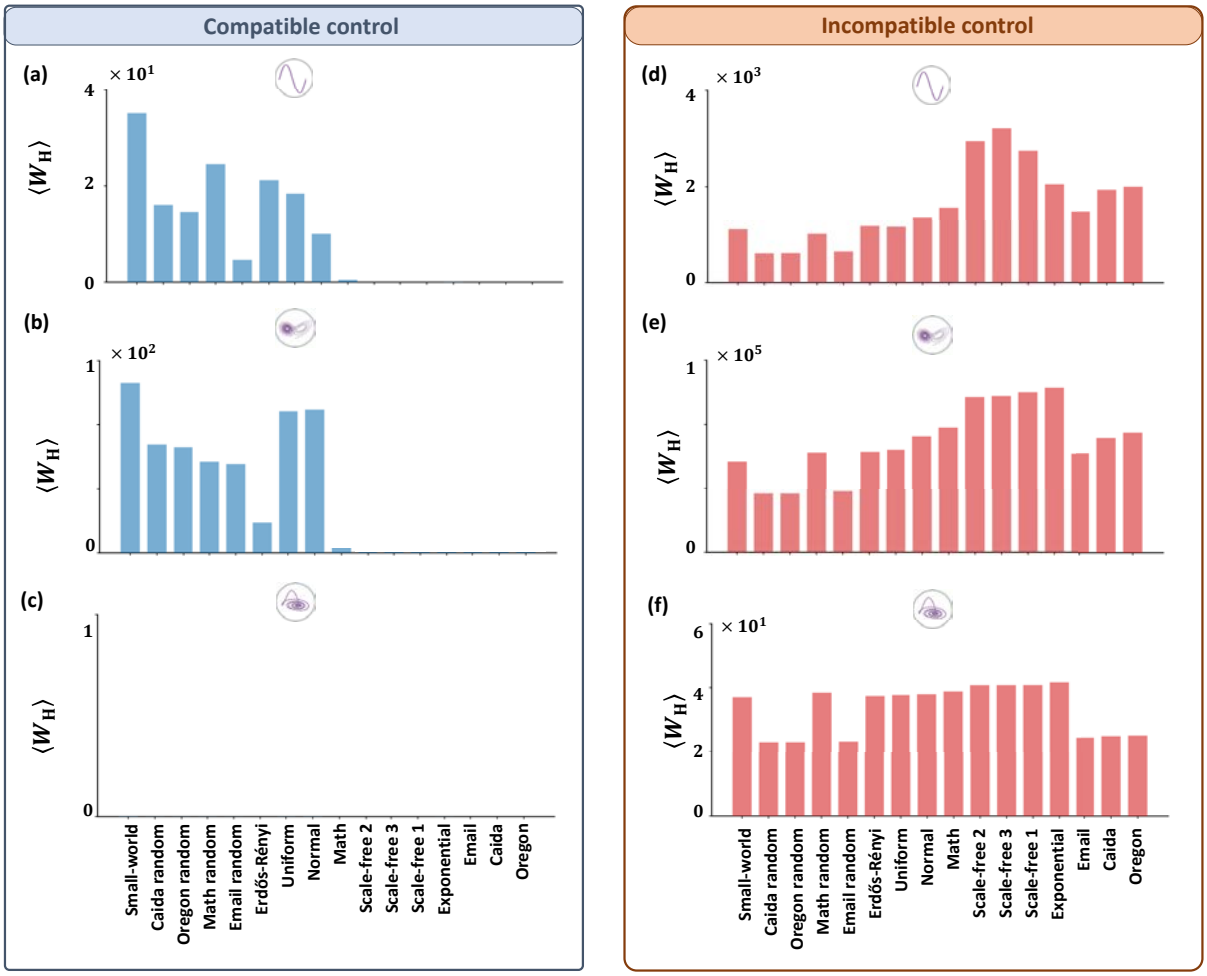

Figure 22: **Average holding power under compatible and incompatible control.** **a**–**c** Average holding power over the 20–25 s interval for all 16 networks shown in Fig. 11. Networks are arranged from least to most heterogeneous for the case of compatible control. The three panels correspond to the Kuramoto, Lorenz, and Rössler dynamics, respectively. **d**–**f** Similar results for the case of incompatible control.

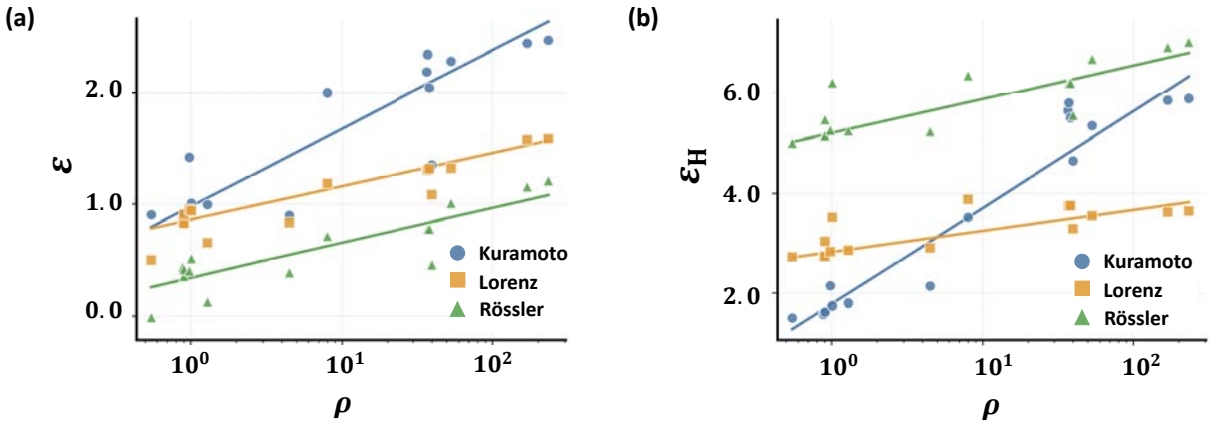

Figure 23: **Energy gap versus network heterogeneity**. **a** Reaching energy gap ε in (9) in the main text versus network heterogeneity ρ, as obtained from Kuramoto (blue), Lorenz (orange) and Rössler (green) dynamics, implemented on each of 16 networks shown in Fig. 11. **b** Holding energy gap ε_{H} in (19) versus network heterogeneity ρ. A clear trend emerges, where heterogeneous networks require more control energy when driven to incompatible states, but achieve compatible control more readily.

References

[1] Liu, Y. Y., Slotine, J. J., & Barabási, A. L. Controllability of complex networks. *Nature* **473**, 167–173 (2011).

[2] Whalen, A. J., Brennan, S. N., Sauer, T. D., & Schiff, S. J. Observability and controllability of nonlinear networks: The role of symmetry. *Phys. Rev. X* **5**, 011005 (2015).

[3] Leskovec, J., Kleinberg, J., & Faloutsos, C. Graph evolution: Densification and shrinking diameters. *ACM Trans. Knowl. Discov. Data (TKDD)* **1**, 2-es (2007).

[4] Leskovec, J., Kleinberg, J., & Faloutsos, C. Graphs over time: densification laws, shrinking diameters and possible explanations. *Proc. 11th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, 177–187 (2005).

[5] Klimt, B., & Yang, Y. The Enron corpus: A new dataset for email classification research. *Eur. Conf. Mach. Learn.*, 217–226 (2004).

[6] Palla, G., Barabási, A. L., & Vicsek, T. Quantifying social group evolution. *Nature* **446**, 664–667 (2007).