ARTICLE

OPEN

Check for updates

Data-driven control of complex networks

Giacomo Baggio[1], Danielle S. Bassett [2,3,4,5,6,7] & Fabio Pasqualetti [8✉]

Our ability to manipulate the behavior of complex networks depends on the design of efficient control algorithms and, critically, on the availability of an accurate and tractable model of the network dynamics. While the design of control algorithms for network systems has seen notable advances in the past few years, knowledge of the network dynamics is a ubiquitous assumption that is difficult to satisfy in practice. In this paper we overcome this limitation, and develop a data-driven framework to control a complex network optimally and without any knowledge of the network dynamics. Our optimal controls are constructed using a finite set of data, where the unknown network is stimulated with arbitrary and possibly random inputs. Although our controls are provably correct for networks with linear dynamics, we also characterize their performance against noisy data and in the presence of nonlinear dynamics, as they arise in power grid and brain networks.

---

[1] Department of Information Engineering, University of Padova, Padova, Italy. [2] Department of Bioengineering, University of Pennsylvania, Philadelphia, PA, USA. [3] Department of Physics and Astronomy, University of Pennsylvania, Philadelphia, PA, USA. [4] Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA, USA. [5] Department of Neurology, University of Pennsylvania, Philadelphia, PA, USA. [6] Department of Psychiatry, University of Pennsylvania, Philadelphia, PA, USA. [7] Santa Fe Institute, Santa Fe, NM, USA. [8] Department of Mechanical Engineering, University of California at Riverside, Riverside, CA, USA. ✉email: fabiopas@engr.ucr.edu

With the development of sensing, processing, and storing capabilities of modern sensors, massive volumes of information-rich data are now rapidly expanding in many physical and engineering domains, ranging from robotics[1] to biological[2,3] and economic sciences[4]. Data are often dynamically generated by complex interconnected processes, and encode key information about the structure and operation of these networked phenomena. Examples include temporal recordings of functional activity in the human brain[5], phasor measurements of currents and voltages in the power distribution grid[6], and streams of traffic data in urban transportation networks[7]. When first-principle models are not conceivable, costly, or difficult to obtain, this unprecedented availability of the data offers a great opportunity for scientists and practitioners to better understand, predict, and, ultimately, control the behavior of real-world complex networks.

Existing works on the controllability of complex networks have focused exclusively on a model-based setting[8–14], although, in practice, constructing accurate models of large-scale networks is a challenging, often unfeasible, task[15–17]. In fact, errors in the network model (i.e., missing or extra links, incorrect link weights) are unavoidable, especially if the network is identified from data[18,19] (see Fig. 1a). This uncertainty is particularly important for network controllability, since, as exemplified in Fig. 1b, c, the computation of model-based network controls tends to be unreliable and highly sensitive to model uncertainties, even for moderate size networks, if the network is controlled by few nodes[20,21]. It is therefore natural to ask whether network controls

can be learned directly from data, and, if so, how well these data-driven control policies perform.

Data-driven control of dynamical systems has attracted increasing interest over the last few years, triggered by recent advances and successes in machine learning and artificial intelligence[22,23]. The classic (indirect) approach to learn controls from data is to use a sequential system identification and control design procedure. That is, one first identifies a model of the system from the available data and then computes the desired controls using the estimated model[24]. However, identification algorithms are sometimes inaccurate and time-consuming, and several direct data-driven methods have been proposed to bypass the identification step[25]. These include, among others, (model-free) reinforcement learning[26,27], iterative learning control[28], adaptive and self-tuning control[29], and behavior-based methods[30,31].

The above techniques differ in the data-generation procedure, class of system dynamics considered, and control objectives. In classic reinforcement learning settings, data are generated online and updated under the guidance of a policy evaluator or reward simulator, which in many applications is represented by an offline-trained (deep) neural network[32]. Iterative learning control is used to refine and optimize repetitive control tasks: data are recorded online during the execution of a task repeated multiple times, and employed to improve tracking accuracy from trial to trial. In adaptive control, the structure of the controller is fixed and a few control parameters are optimized using data collected on the fly. A widely known example is the auto-tuning of PID
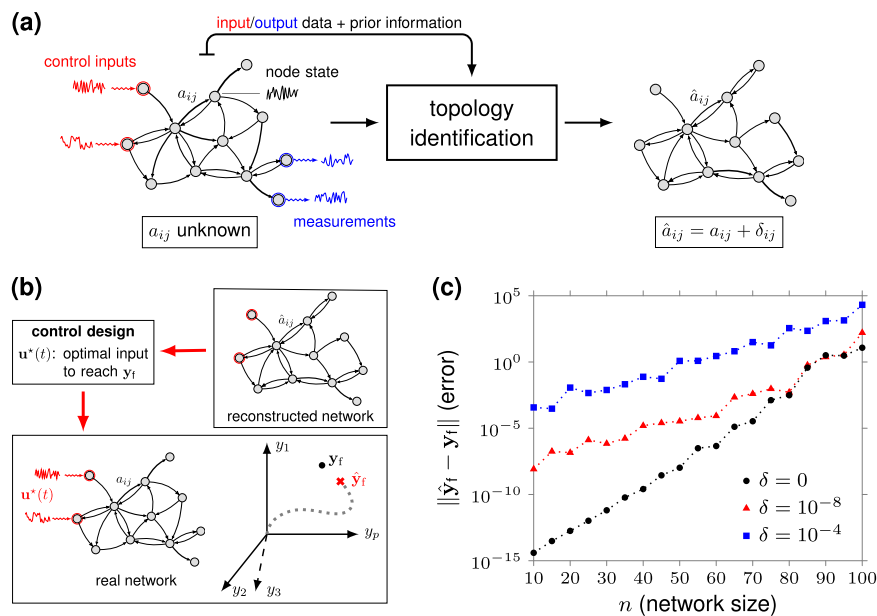

Fig. 1 The effect of model uncertainty in the computation of optimal network controls. Panel (**a**) shows a schematic of a classic network identification procedure. The reconstructed network is affected by estimation errors δ_{ij}. The symbol $\{a_{ij}\}$ denote the correct network weights, whereas $\{\hat{a}_{ij}\}$ the (incorrectly) reconstructed ones. Panel (**b**) illustrates the error in the final (output) state induced by an optimal control design based on the reconstructed network. The symbol $\mathbf{y}_f$ denote the desired final state of (a subset of) the network nodes, whereas $\hat{\mathbf{y}}_f$ the one reached by the optimal input $\mathbf{u}^\star(t)$. In panel (**c**), we consider minimum-energy controls designed from exact and incorrectly reconstructed linear dynamical networks, and compute the resulting error in the final state as the network size n varies. We consider connected Erdös–Rényi networks with edge probability $p_{edge} = \ln n/n + 0.1$, ten randomly selected control nodes, control horizon $T = 2n$, and a randomly chosen final state $\mathbf{x}_f \in \mathbb{R}^n$. We assume $\mathbf{y}_f = \mathbf{x}_f$, i.e., the nodes that we want to control coincide with all network nodes. To mimic errors in the network reconstruction process, we add to each edge of the network a disturbance modeled as an i. i.d. random variable uniformly distributed in $[-\delta, \delta]$, $\delta > 0$. Each curve represents the average of the (norm of the) error in the final state over 100 independent realizations of $\mathbf{x}_f$. To compute minimum-energy control inputs, we use the classic Gramian-based formula and standard LAPACK linear-algebra routines (see "Methods"). Notice that there is a nonzero error in the final state which grows with the size of the network even in the absence of uncertainty ($\delta = 0$). This is due to numerical errors in the computation of the minimum-energy control which are a consequence of the ill-conditioning of the Gramian[9,20].

controllers[33]. Behavior-based techniques exploit a trajectory-based (or behavioral) representation of the system, and data that typically consist of a single, noiseless, and sufficiently long input–output system trajectory[31]. Each of the above data-driven approaches has its own limitations and merits, which strongly depend on the intended application area. However, a common feature of all these approaches is that they are tailored to or have been employed for closed-loop control tasks, such as stabilization or tracking, and not for finite-time point-to-point control tasks.

In this paper, we address the problem of learning from data point-to-point optimal controls for complex dynamical networks. Precisely, following recent literature on the controllability of complex networks[34,35], we focus on control policies that optimally steer the state of (a subset of) network nodes from a given initial value to a desired final one within a finite-time horizon. To derive analytic, interpretable results that capture the role of the network structure, we consider networks governed by linear dynamics, quadratic cost functions, and data consisting of a set of control experiments recorded offline. Importantly, experimental data are not required to be optimal, and can even be generated through random control experiments. In this setting, we establish closed-form expressions of optimal data-driven control policies to reach the desired target state and, in the case of noiseless data, characterize the minimum number of experiments needed to exactly reconstruct optimal control inputs. Further, we introduce suboptimal yet computationally simple data-driven expressions and discuss certain numerical and computational advantages of using our data-driven approach when compared to classic model-based ones. Finally, we illustrate with different numerical studies how our framework can be applied to (i) induce prescribed patterns of synchronization in networks of oscillators, (ii) restore the correct operation of power-grid networks after a fault, and (iii) characterize the controllability properties of functional brain networks.

While the focus of this paper is on designing optimal control inputs, the expressions derived in this work could also serve as an alternative, computationally reliable, and efficient way of (i) analyzing the controllability properties of network systems and (ii) solving control-related network problems, such as the optimal selection of control and sensor nodes. In particular, as a by-product of our analysis, we show that (output) controllability can be assessed directly and simply from data. This constitutes an additional methodological contribution to the extensive literature on the model-based analysis of network controllability.

Results

Network dynamics and optimal point-to-point control. We consider networks governed by linear time-invariant dynamics

$$\mathbf{x}(t+1) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t),$$
$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t), \tag{1}$$

where $\mathbf{x}(t) \in \mathbb{R}^n$, $\mathbf{u}(t) \in \mathbb{R}^m$, and $\mathbf{y}(t) \in \mathbb{R}^p$ denote, respectively, the state, input, and output of the network at time t. The matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ describes the (directed and weighted) adjacency matrix of the network, and the matrices $\mathbf{B} \in \mathbb{R}^{n \times m}$ and $\mathbf{C} \in \mathbb{R}^{p \times n}$, respectively, are typically chosen to single out prescribed sets of input and output nodes of the network.

In this work, we are interested in solving point-to-point control problems; that is, designing open-loop control policies that steer the network output $\mathbf{y}(t)$ from an initial value $\mathbf{y}(0) = \mathbf{y}_0$ to a desired one $\mathbf{y}(T) = \mathbf{y}_f$ in a finite number of steps T. If $\mathbf{y}_f$ is output controllable in T steps (a standing assumption in this paper; we refer to Supplementary Note 1 for more details), then the latter problem admits a solution and, in fact, there are many ways to accomplish such a control task. Here, we assume that the network

is initially relaxed ($\mathbf{x}(0) = \mathbf{0}$), and we seek the control input $\mathbf{u}_{0:T-1}^\star = [\mathbf{u}^\star(T-1)^\mathsf{T} \cdots \mathbf{u}^\star(0)^\mathsf{T}]^\mathsf{T}$ that drives the output of the network to $\mathbf{y}_f$ in T steps and, at the same time, minimizes a prescribed quadratic combination of the control effort and locality of the controlled trajectories.

Mathematically, we study and solve the following constrained minimization problem:

$$\mathbf{u}_{0:T-1}^\star = \arg\min_{\mathbf{u}_{0:T-1}} \mathbf{y}_{1:T-1}^\mathsf{T}\, \mathbf{Q}\, \mathbf{y}_{1:T-1} + \mathbf{u}_{0:T-1}^\mathsf{T}\, \mathbf{R}\, \mathbf{u}_{0:T-1}$$
$$\text{s.t. Eq. (1) and } \mathbf{y}_T = \mathbf{y}_f, \tag{2}$$

where $\mathbf{Q} \succcurlyeq \mathbf{0}$ and $\mathbf{R} \succ \mathbf{0}$ are tunable (positive semidefinite and positive definite, respectively) matrices that penalize output deviation and input usage, respectively, $\mathbf{y}_{1:T-1} = [\mathbf{y}(1)^\mathsf{T}\ \mathbf{y}(2)^\mathsf{T} \cdots \mathbf{y}(T-1)^\mathsf{T}]^\mathsf{T}$, and $\mathbf{y}_T = \mathbf{y}(T)$. Problem (2) generalizes the classic (open-loop) linear–quadratic control framework by including the possibility of minimizing a linear function of the state (as opposed to the whole state) in addition to the control input. Further, we remark that increasing $\mathbf{R}$ in Eq. (2) leads to optimal control inputs that achieve the desired final state with increasingly smaller magnitudes. Similarly, the matrix $\mathbf{Q}$ in Eq. (2) weighs the norm of the output (state), so that increasing $\mathbf{Q}$ forces the optimization problem to generate inputs that limit the norm of the output (state), at the expenses of using a larger control input[36]. In particular, if $\mathbf{Q} = \mathbf{0}$ and $\mathbf{R} = \mathbf{I}$, then $\mathbf{u}_{0:T-1}^\star$ coincides with the minimum-energy control to reach $\mathbf{y}_f$ in T steps[9,37].

Equation (2) admits a closed-form solution whose computation requires the exact knowledge of the network matrix $\mathbf{A}$ and suffers from numerical instabilities ("Methods"). In the following section, we address this limitation by deriving model-free and reliable expressions of $\mathbf{u}_{0:T-1}^\star$ that solely rely on experimental data collected during the network operation.

Learning optimal controls from non-optimal data. We assume that the network matrix $\mathbf{A}$ is unknown and that N control experiments have been performed with the dynamical network in Eq. (1). The ith experiment consists of generating and applying the input sequence $\mathbf{u}_{0:T-1}^{(i)}$, and measuring the resulting output trajectory $\mathbf{y}_{0:T}^{(i)}$ (Fig. 2a). Here, as in ref. [38], we consider episodic experiments where the network state is reset to zero before running a new trial, and refer to Supplementary Note 5 for an extension to the nonepisodic setting and to the case of episodic experiments with nonzero initial state resets. We let $\mathbf{U}_{0:T-1}$, $\mathbf{Y}_{1:T-1}$, and $\mathbf{Y}_T$ denote the matrices containing, respectively, the experimental inputs, the output measurements in the time interval $[1, T-1]$, and the output measurements at time T. Namely,

$$\mathbf{U}_{0:T-1} = \left[\mathbf{u}_{0:T-1}^{(1)} \cdots \mathbf{u}_{0:T-1}^{(N)}\right],$$
$$\mathbf{Y}_{1:T-1} = \left[\mathbf{y}_{1:T-1}^{(1)} \cdots \mathbf{y}_{1:T-1}^{(N)}\right], \tag{3}$$
$$\mathbf{Y}_T = \left[\mathbf{y}_T^{(1)} \cdots \mathbf{y}_T^{(N)}\right].$$

An important aspect of our analysis is that we do not require the input experiments to be optimal, in the sense of Eq. (2), nor do we investigate the problem of experiment design, i.e., generating data that are informative for our problem. In our setting, data are given, and these are generated from arbitrary, possibly random, or carefully chosen experiments.

By relying on the data matrices in Eq. (3), we derive the following data-driven candidate solution to the minimization problem in Eq. (2) (Supplementary Note 2):

$$\hat{\mathbf{u}}_{0:T-1} = \mathbf{U}_{0:T-1}(\mathbf{I} - \mathbf{K}_{\mathbf{Y}_T}(\mathbf{L}\mathbf{K}_{\mathbf{Y}_T})^\dagger \mathbf{L})\mathbf{Y}_T^\dagger\, \mathbf{y}_f, \tag{4}$$
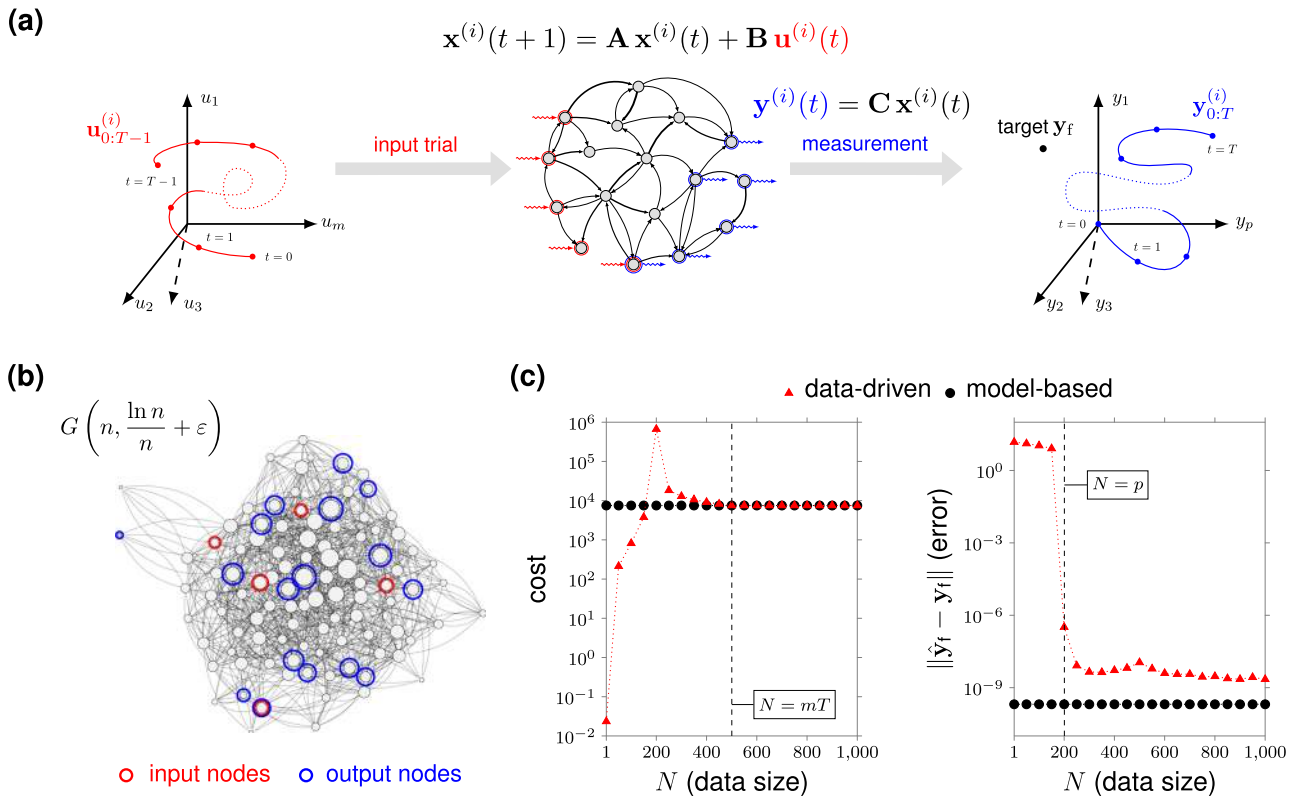
Fig. 2 Experimental setup and optimal data-driven network controls. Panel (**a**) illustrates the data-collection process. With reference to the *i*th control experiment, a *T*-step input sequence $\mathbf{u}_{0:T}^{(i)}$ excites the network dynamics in Eq. (1), and the time samples of the resulting output trajectory $\mathbf{y}_{0:T}^{(i)}$ are recorded. The input trajectory $\mathbf{u}_{0:T}^{(i)}$ may be generated randomly, so that the final output $\mathbf{y}_T^{(i)}$ does not normally coincide with the desired target output $\mathbf{y}_f$. Red nodes denote the control or input nodes (forming matrix **B**) and the blue nodes denote the measured or output nodes (forming matrix **C**). Panel (**b**) shows a realization of the Erdös–Rényi graph model $G(n, p_{\text{edge}})$ used in our examples, where n is the number of nodes, p_{edge} is the edge probability. We set the edge probability to $p_{\text{edge}} = \ln n/n + \varepsilon$, $\varepsilon = 0.05$, to ensure connectedness with high probability, and normalize the resulting adjacency matrix by $\sqrt{n}$. Panel (**c**) shows the value of the cost function (left) and the (norm of the) error in the final state (right) for the data-driven input (4) and the model-based control as a function of the number of data points. The symbol $\mathbf{y}_f$ denotes the desired final target and $\hat{\mathbf{y}}_f$ the output reached by the (model-based or data-driven) control input. We choose $\mathbf{Q} = \mathbf{R} = \mathbf{I}$, $n = 1000$, $T = 10$, $m = 50$, and $p = 200$, and consider Erdös–Rényi networks as in panel (**b**). For additional details, see "Methods".

where **L** is any matrix satisfying $\mathbf{L}^{\mathsf{T}}\mathbf{L} = \mathbf{Y}_{1:T-1}^{\mathsf{T}}\mathbf{Q}\mathbf{Y}_{1:T-1} + \mathbf{U}_{0:T-1}^{\mathsf{T}}\mathbf{R}\mathbf{U}_{0:T-1}$, $\mathbf{K}_{\mathbf{Y}_T}$ denotes a matrix whose columns form a basis of the kernel of $\mathbf{Y}_T$, and the superscript symbol $\cdot^{\dagger}$ stands for the Moore–Penrose pseudoinverse operation[39]. We remark that the dependence of Eq. (4) on the (unknown) network parameters **A**, **B**, and **C** is implicit and encoded in the collected data $\mathbf{U}_{0:T-1}$, $\mathbf{Y}_{1:T-1}$, and $\mathbf{Y}_T$. Further, we stress that $\hat{\mathbf{u}}_{0:T-1}$ in Eq. (4) does not, in general, coincide with the optimal solution $\mathbf{u}_{0:T-1}^{\star}$ in Eq. (2). However, if enough linearly independent data are collected, then $\hat{\mathbf{u}}_{0:T-1} = \mathbf{u}_{0:T-1}^{\star}$, as we illustrate next.

Minimum number of the data to learn optimal controls and a data-based controllability condition. Finite data suffice to exactly reconstruct the optimal control input via the data-driven expression in Eq. (4). The minimum number of data N_{min} required to accomplish such a task depends both on the target $\mathbf{y}_f$ and the available data matrices, which are in turn implicitly dependent on the (unknown) parameters **A**, **B**, and **C**. However, it is possible to establish a simple upper bound on N_{min}. Namely, if the input data matrix in Eq. (3) contains mT linearly independent experiments, that is, if $\mathbf{U}_{0:T-1}$ has full row rank, then $N_{\text{min}} \le mT$ (Supplementary Note 3). We stress that linear independence of the control experiments is a mild condition that is normally satisfied when the experiments are generated randomly. Further, if the number of independent trials is smaller than mT

but such that $\mathbf{y}_f$ belongs to the range space of $\mathbf{Y}_T$, then the data-driven control $\hat{\mathbf{u}}_{0:T-1}$ still correctly steers the network output to $\mathbf{y}_f$ in T steps, but with a cost that is typically larger than the optimal one (Supplementary Note 3). In this case, $\hat{\mathbf{u}}_{0:T-1}$ is a suboptimal solution to Eq. (2), which becomes optimal if the collected data are made of control experiments that are optimal as well. We stress that for a number $N \ge p$ of randomly chosen control experiments, any (output controllable) $\mathbf{y}_f$ normally belongs to the range space of $\mathbf{Y}_T$. In Fig. 2c, we illustrate the above observations for the class of Erdös–Rényi networks of Fig. 2b.

Finally, as a by-product of the above analysis, it follows that the (output) controllability of a network system can be checked directly and simply from the data. Specifically, if $N \ge mT$ linearly independent input experiments are collected with $T \ge n$, then the system is output controllable if and only if the columns of the output data matrix $\mathbf{Y}_T$ span the entire space $\mathbb{R}^p$; that is, if and only if $\mathbf{Y}_T$ has full row rank (Supplementary Note 4). For $p = n$, the latter condition can be used to assess the classic controllability of a network system from data.

Data-driven minimum-energy control. By letting $\mathbf{Q} = \mathbf{0}$ and $\mathbf{R} = \mathbf{I}$ in Eq. (4), we recover a data-driven expression for the T-step minimum-energy control to reach $\mathbf{y}_f$. We remark that the family of minimum-energy controls has been extensively employed to characterize the fundamental capabilities and limitations of
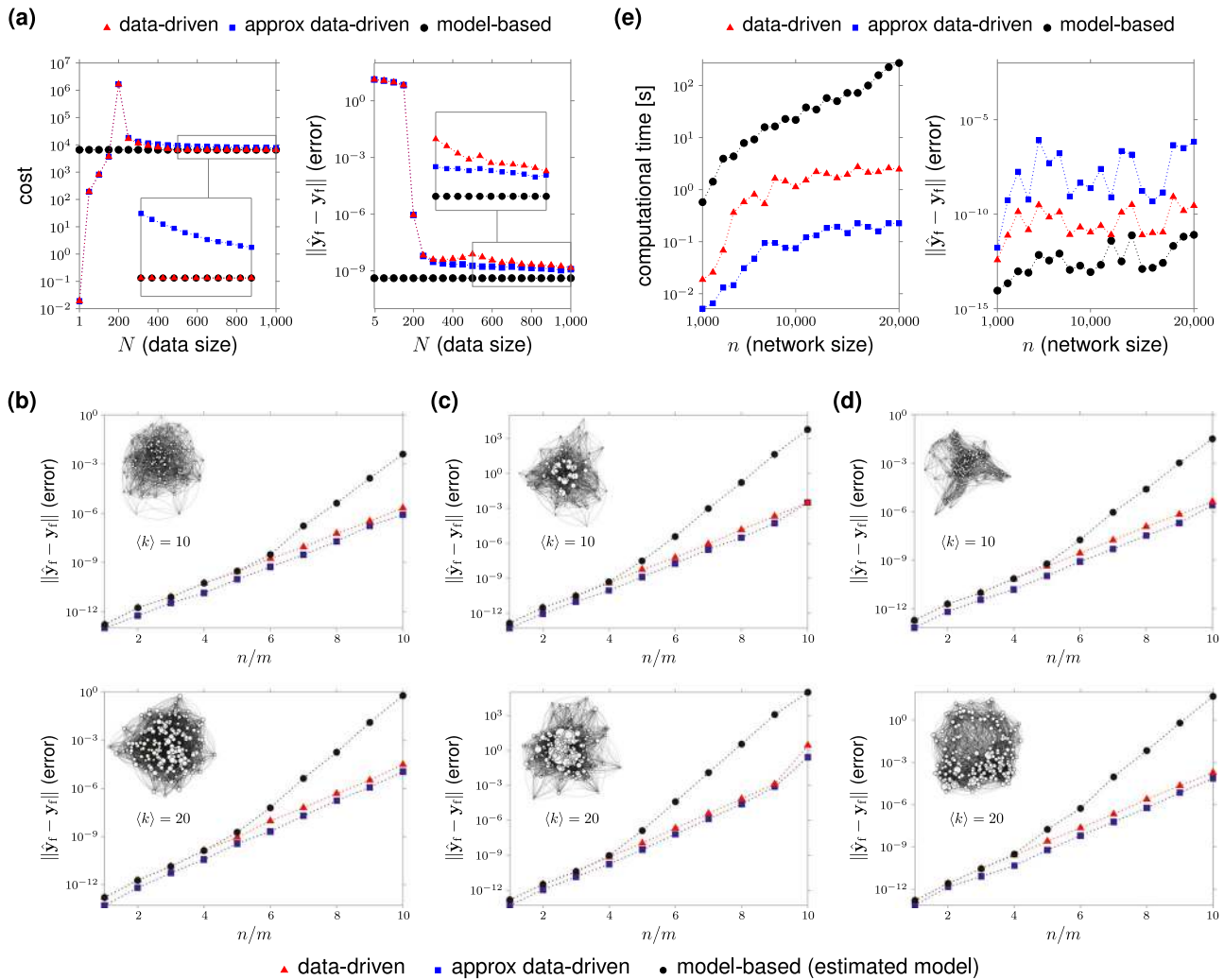
Fig. 3 Performance of minimum-energy data-driven network controls. Panel (**a**) shows the value of the cost function (left) and the (norm of the) error in the final state (right) for the minimum-energy data-driven controls (5) and (6), and the model-based one as a function of the data size *N*. We consider Erdös–Rényi networks as in Fig. 2b with $\varepsilon = 0.05$, and parameters $n = 1000$, $T = 10$, $m = 50$, $p = 200$. In panels (**b**–**d**), we assume **C = I** and compare the error in the final state generated by the data-driven minimum-energy controls (5) and (6) and model-based expression for a fixed number of control nodes $m = 100$ and increasing dimension $n \in [100, 1000]$. The model-based control has been computed by first estimating matrices **A** and **B** from data according to the subspace-based technique in "Methods", and then using the model-based control expression. In panel (**b**), we consider Erdös–Rényi networks with average degree $\langle k \rangle = 10$ (top) and $\langle k \rangle = 20$ (bottom). In panel (**c**), we consider Barabasi–Albert networks[71] with the initial number of nodes $m_0 = 20$ and average degree $\langle k \rangle = 10$ (top) and $\langle k \rangle = 20$ (bottom). In panel (**d**), we consider Watts–Strogatz networks[72] with rewiring probability $p_{\text{rew}} = 0.2$ and average degree $\langle k \rangle = 10$ (top) and $\langle k \rangle = 20$ (bottom). In all plots, we use a control horizon $T = 15$ and a number of the data $N = mT + 200$. To limit the influence of eigenvalues in the computation of optimal controls across different network models, we normalize matrix **A** by its norm $\|\mathbf{A}\|$. The curves represent the average of over 100 realizations of networks, data, control nodes, and final states. Panel (**e**), left, compares the time needed to compute the optimal controls via data-driven and model-based strategies as a function of the network size, for one realization of the Erdös–Rényi model of Fig. 2b and data. Panel (**e**), right, shows the errors in the final state. We use the following parameters: $\varepsilon = 0.05$, $m = \lfloor n/100 \rfloor$, $p = \lfloor n/50 \rfloor$, $T = 50$, and $N = mT + 100$. For additional details, see "Methods".

controlling networks, e.g., see[9,11,14]. After some algebraic manipulations, the data-driven minimum-energy control input can be compactly rewritten as (Supplementary Note 5)

$$\hat{\mathbf{u}}_{0:T-1} = \left(\mathbf{Y}_T \mathbf{U}_{0:T-1}^\dagger \right)^\dagger \mathbf{y}_{\mathrm{f}}. \qquad (5)$$

The latter expression relies on the final output measurements only (matrix $\mathbf{Y}_T$) and, thus, it does not exploit the full output data (matrix $\mathbf{Y}_{1:T-1}$). An alternative control expression is

$$\tilde{\mathbf{u}}_{0:T-1} = \mathbf{U}_{0:T-1} \mathbf{Y}_T^\dagger \mathbf{y}_{\mathrm{f}}. \qquad (6)$$

This is a simple, suboptimal data-based control sequence that correctly steers the network to $\mathbf{y}_{\mathrm{f}}$ in *T* steps, as long as $\mathbf{y}_{\mathrm{f}}$ belongs

to the range space of $\mathbf{Y}_T$ (a condition that is normally satisfied when *p* randomly generated data are available). Further, and more importantly, when the input data samples are drawn randomly and independently from a Gaussian distribution with zero mean and finite variance, Eq. (6) converges to the minimum-energy control in the limit of infinite data (Supplementary Note 6).

Figure 3a compares the performance (in terms of control effort and error in the final state) of the two data-driven expressions in Eqs. (5) and (6), and the model-based control as a function of the data size *N*. While the data-driven control in Eq. (5) becomes optimal for a finite number of data (precisely, for $N = mT$ independent data), the approximate expression in Eq. (6) tends to

the optimal control only asymptotically in the number of data (Fig. 3a, left). In both cases, the error in the final state goes to zero after collecting $N = p$ data (Fig. 3a, right). For the approximate control in Eq. (6), we also establish upper bounds on the size of the dataset to get a prescribed deviation from the optimal control, in the case of Gaussian input data. Our nonasymptotic analysis indicates that this deviation is proportional to the worst-case control energy required to reach a unit-norm target. This, in turn, implies that networks that are easy to control require fewer trials to attain a prescribed approximation error (Supplementary Note 6).

Numerical and computational benefits of data-driven controls. By relying on the same set of experimental data, in Fig. 3b–d, we compare the numerical accuracy, as measured by the error in the final state, of the data-driven controls in Eqs. (5) and (6) and the minimum-energy control computed via a standard two-step approach comprising a network identification step followed by model-based control design. First, we point out that if some nodes of the network are not accessible ($C \neq I$) and no prior information about the network structure is available, then it is impossible to exactly reconstruct the network matrix A using (any number of) data[40]. In contrast, the computation of minimum-energy inputs is always feasible via our data-driven expression, provided that enough independent data are collected. We thus focus on the case in which the state of all nodes can be measured (i.e., $C = I$). We first consider Erdös–Rényi networks with a fixed number of control nodes $m = 100$ and increasing dimension $n \in [100, 1000]$. To reconstruct the network matrices A and B, we employ the subspace-based identification technique described in "Methods". Although both the data-driven and the model-based controls yield a poor numerical accuracy for increasing values of n/m (which is due to the fact that the energy to control a network typically grows exponentially with the ratio n/m[9,11,20,21]) the model-based input exhibits a faster growth of the error when compared to the data-driven ones for sufficiently large values of n/m (Fig. 3b). We find similar tradeoffs for other random network models, namely scale-free (Barabási–Albert model) and small-world (Watts–Strogatz model), as illustrated in Figs. 3c and d, respectively. This poor performance of the standard approach is somehow expected because, independently of the network identification procedure, the standard two-step approach requires a number of operations larger than those required by the data-driven approach, resulting in a potentially higher sensitivity to round-off errors. Also, we empirically observe that the gain in numerical accuracy offered by data-driven control inputs is more significant for dense networks (see also Supplementary Fig. 1). Finally, it is worth noting that the approximate data-driven control given in Eq. (6), even though suboptimal, yields the best accuracy. This is particularly interesting since, for a finite number of the data, Eq. (6) does not have a model-based counterpart.

A further advantage in using data-driven controls over model-based ones arises when dealing with massive networks featuring a small fraction of input and output nodes. Specifically, in Fig. 3e we plot the time needed to numerically compute the data-driven and model-based controls as a function of the size of the network. We focus on Erdös–Rényi networks as in Fig. 2b of dimension $n \geq 1000$ with $\lfloor n/100 \rfloor$ input and output nodes and a control horizon $T = 50$. The model-based control input requires the computation of the first $T - 1$ powers of A ("Methods"). The computation of the data-driven expressions in Eqs. (5) and (6) involves, instead, linear-algebraic operations on two matrices ($U_{0:T-1}$ and Y_T) that are typically smaller than A when n is very large (precisely, when $T < n/m$ and $N < n$). Thus, the computation of the control input via the data-driven approach is normally faster

than the classic model-based computation (Fig. 3e, left). In particular, the data-driven control given in Eq. (6), although suboptimal, yields the most favorable performance due to its particularly simple expression. Finally, we note that the error in the final state committed by the data-driven controls is always upper bounded by 10^{-5} and thus it has a negligible effect on the control accuracy (Fig. 3e, right).

Data-driven controls with noisy data. The analysis so far has focused on noiseless data. A natural question is how the data-driven controls behave in the case of noisy data. If the noise is unknown but small in magnitude, then the established data-driven expressions will deviate slightly from the correct values (Supplementary Note 7). However, if some prior information on the noise is known, this information can be exploited to return more accurate control expressions. A particularly relevant case is when data are corrupted by additive i.i.d. noise with zero mean and known variance. Namely, the available data read as

$$\begin{aligned} U_{0:T-1} &= \bar{U}_{0:T-1} + \Delta_U, \\ Y_{1:T-1} &= \bar{Y}_{1:T-1} + \Delta_Y, \\ Y_T &= \bar{Y}_T + \Delta_{Y_T}, \end{aligned} \qquad (7)$$

where $\bar{U}_{0:T-1}$, $\bar{Y}_{1:T-1}$, and $\bar{Y}_T$ denote the ground truth values, and Δ_U, Δ_Y, and Δ_{Y_T} are random matrices with i.i.d. entries with zero mean and variance σ_U^2, σ_Y^2, and $\sigma_{Y_T}^2$, respectively. In this setting, it can be shown that the data-driven control in Eq. (4) and the data-driven minimum-energy controls in Eqs. (5) and (6) are typically not consistent; that is, they do not converge to the true control inputs as the data size tends to infinity (see Supplementary Note 7 for a concrete example). However, by suitably modifying these expressions, it is possible to recover asymptotically correct data-driven formulas (Supplementary Note 7). The key idea is to add correction terms that compensate for the noise variance arising from the pseudoinverse operations. In particular, the asymptotically correct version of the data-driven controls in Eqs. (5) and (6) read, respectively, as

$$\hat{u}_{0:T-1}^{(c)} = (Y_T U_{0:T-1}^\mathsf{T} (U_{0:T-1} U_{0:T-1}^\mathsf{T} - N\sigma_U^2 I)^\dagger)^\dagger y_f, \qquad (8)$$

$$\tilde{u}_{0:T-1}^{(c)} = U_{0:T-1} Y_T^\mathsf{T} (Y_T Y_T^\mathsf{T} - N\sigma_{Y_T}^2 I)^\dagger y_f, \qquad (9)$$

where we used the fact that $X^\dagger = X^\mathsf{T}(XX^\mathsf{T})^\dagger$ for any matrix X[39], and $N\sigma_U^2 I$ and $N\sigma_{Y_T}^2 I$ represent the noise-dependent correction terms. Note, in particular, that if the noise corrupts the output data Y_T only, then Eq. (8) coincides with the original data-driven control in Eq. (5), so that no correction is needed. Similarly, if the noise corrupts the input data U_T only, then Eq. (9) coincides with the data-driven control in Eq. (6).

Data-driven pattern control of synchronized activity in Kuramoto networks. The problem of inducing desired patterns of synchronized activity in networks of oscillators has several applications in many natural and technological networks[41,42]. For instance, in the clinical treatment of neurological disorders[43–46] and in the administration and dispatch of power in distribution networks[47–49]. For these reasons, several methods have been investigated in the literature for the control of synchronized patterns of activity[46,49–51]. Here, we show how the data-driven framework proposed in this paper can be employed to provide a solution to this problem.

To this end, we consider a simple yet insightful example, that is, a ring network of n Kuramoto oscillators. The dynamics of the
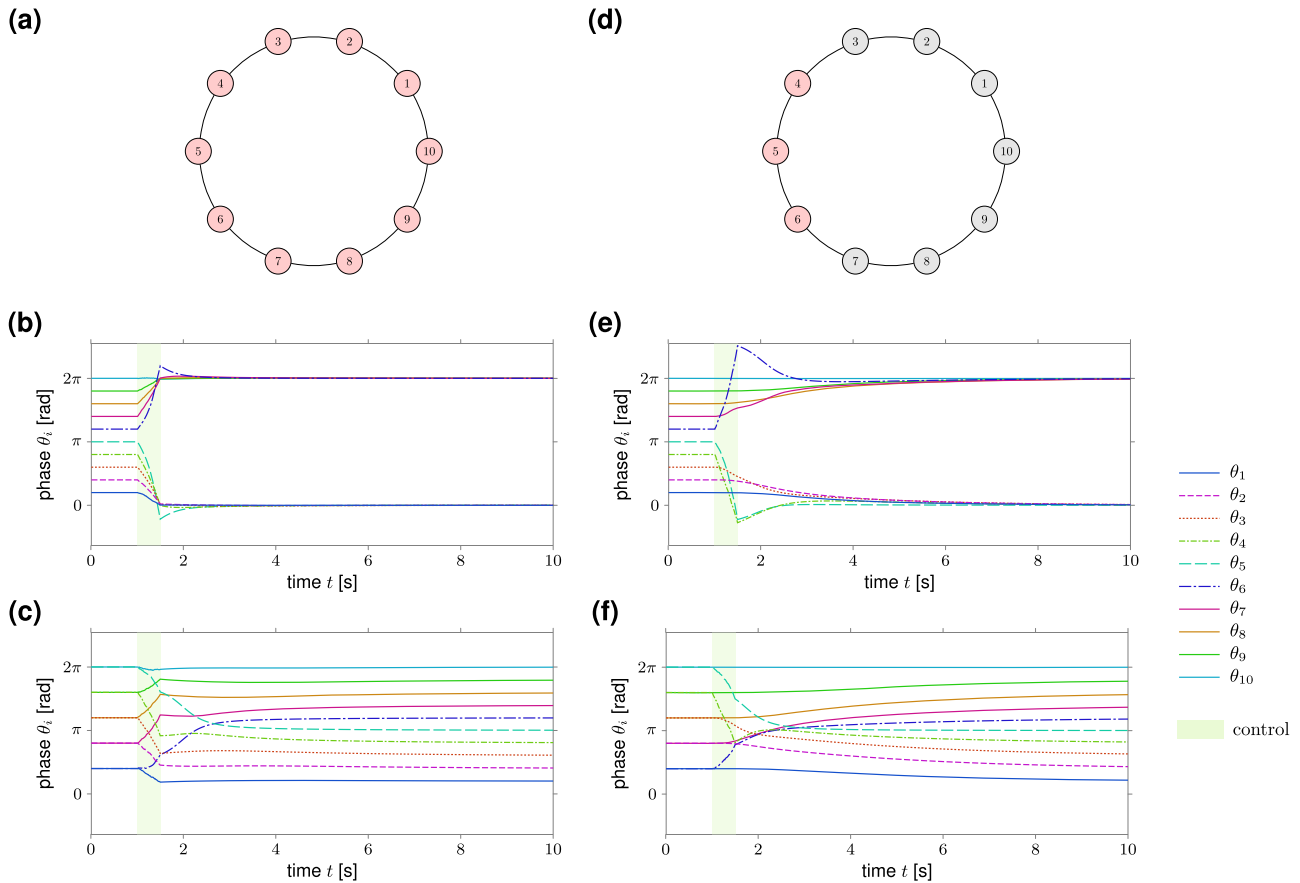
Fig. 4 Data-driven control of synchronized patterns in a ring of Kuramoto oscillators. We consider a ring network of $n = 10$ Kuramoto oscillators for two different configurations of control nodes (red nodes), namely $m = 10$ (**a**) and $m = 3$ (**d**). For both configurations, we apply the data-driven control (Eq. (4) of the main text) to steer the phases from the splay state $\{\bar{\theta}_{1,i}(t)\}$ to the synchronous state (**b**, **e**) and from the splay state $\{\bar{\theta}_{2,i}(t)\}$ to $\{\bar{\theta}_{1,i}(t)\}$ (**c**, **f**). The green region denotes the application of the control. We choose as parameters $T = 50$ samples of the discretized dynamics (corresponding to a control horizon of 0.5 s), $\mathbf{Q} = 5\mathbf{I}$, $\mathbf{R} = \mathbf{I}$, and $N = 1000$ data obtained by perturbing the initial equilibrium with i.i.d. Gaussian inputs with zero mean and standard deviation 0.1.

phases of the oscillators are

$$\dot{\theta}_i(t) = \omega_i + \sin(\theta_{i-1}(t) - \theta_i(t)) + \sin(\theta_{i+1}(t) - \theta_i(t)), \quad i = 1, \dots, n,$$
(10)

where ω_i is the natural frequency of the ith oscillator and the index i is periodic mod n. We consider the case where $\omega_i = \omega$ for all i. In this case, the network always has a stable synchronous state given by $\bar{\theta}_i(t) = \omega t$ for all i. However, as n grows other stable equilibria arise, namely[52]:

$$\bar{\theta}_{q,i}(t) = \omega t + \frac{2\pi q i}{n} + c, \quad i = 1, \dots, n,$$
(11)

where c is an arbitrary constant and q is the winding number which takes integer values. These equilibria correspond to phases linearly spaced on the unit circle and are commonly referred to as splay states. To simulate the dynamics in Eq. (10) and generate the data, we discretize the Kuramoto dynamics via the forward Euler method with discretization step 0.01. In Fig. 4, we consider a network of $n = 10$ oscillators with $\omega = 0$. In the top plots, we assume that we have access to all nodes of the network ($m = 10$) and apply an external control input to steer the network: (i) from the splay state $\{\bar{\theta}_{1,i}(t)\}$ to the synchronous state $\{\bar{\theta}_i(t)\}$ (Fig. 4b), and (ii) from the splay state $\{\bar{\theta}_{2,i}(t)\}$ to the splay state $\{\bar{\theta}_{1,i}(t)\}$ (Fig. 4c). The control input has been computed using the data-driven expression given in Eq. (4) with $T = 50$ samples

(corresponding to a control horizon of 0.5 s), and parameters $\mathbf{Q} = 5\mathbf{I}$, $\mathbf{R} = \mathbf{I}$. Further, we subtract to each output data sample in matrix $\mathbf{Y}_{1:T-1}$ the value of the final equilibrium point, so that choosing a sufficiently large $\mathbf{Q}$ favor trajectories with a small deviation from the equilibrium. We remark that the choice of $\mathbf{Q}$ is particularly important when applying our expressions to nonlinear networks. In this case, choosing a large $\mathbf{Q}$ often improves the applicability and effectiveness of our methods, since a nonlinear system approximately behaves as a linear one in a sufficiently small neighborhood of equilibrium (see also Supplementary Fig. 2). Data have been generated through $N = 1000$ control experiments obtained by perturbing the initial equilibrium with i.i.d. Gaussian external inputs with zero mean and standard deviation 0.1. In the right plots of Fig. 4, we repeat the same experiments using $m = 3$ control nodes (red nodes in Fig. 4d). In both scenarios, the control input does not exactly drive the network to the desired target state (because of the nonlinearity of the dynamics) but to a state close to it. Nevertheless, the final state falls within the basin of attraction of the desired target equilibrium so that the network reaches asymptotically the desired pattern of synchronization. Finally, we point out that the above-described procedure can, in principle, be applied to more complex (possibly random) network topologies. In such cases, however, it is typically more challenging to determine the initial and final equilibrium configurations that specify the considered point-to-point control problem[53].
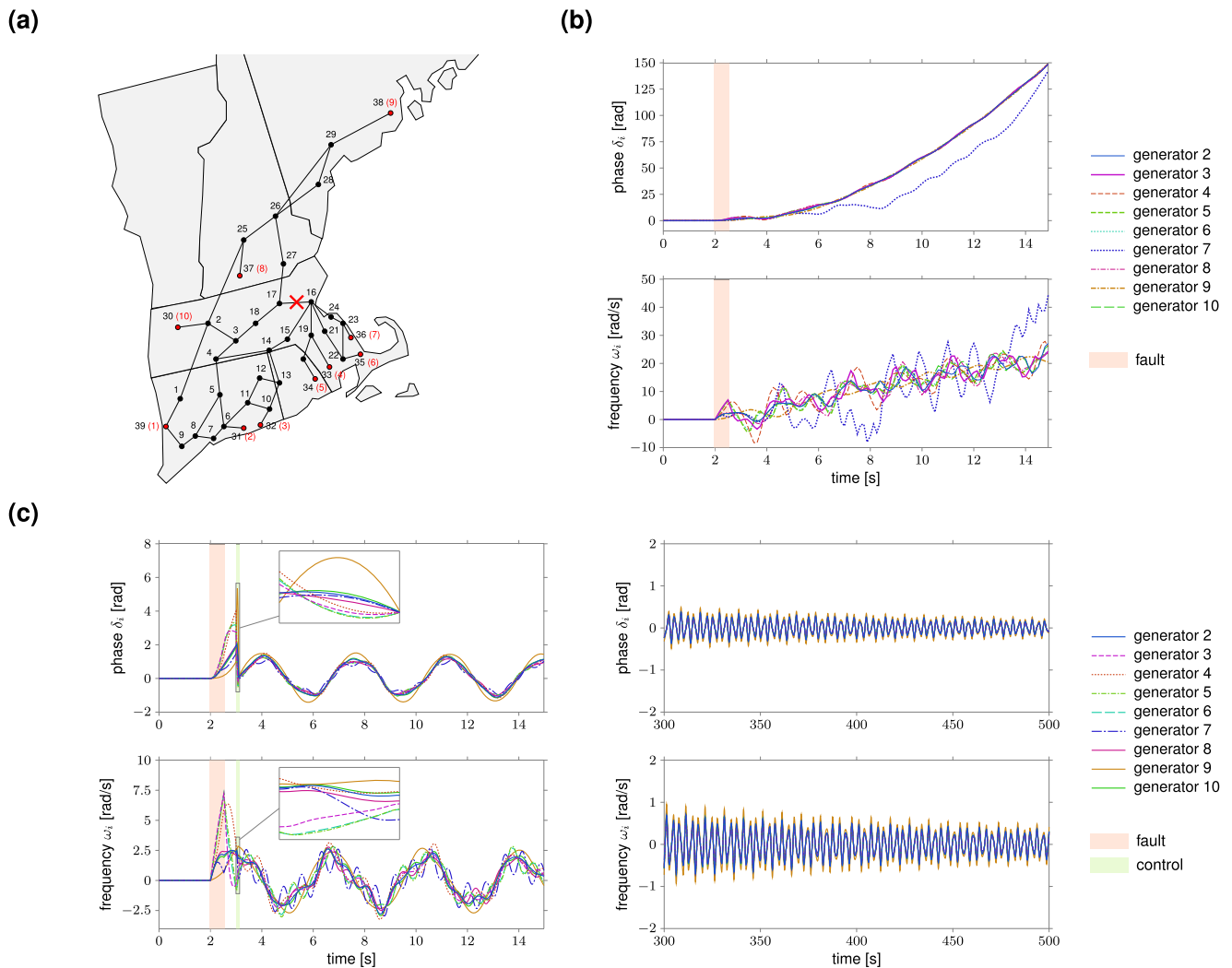
Fig. 5 Data-driven fault recovery in the New England power-grid network. Panel (**a**) depicts the 39-node New England power-grid network (see ref. [70], Appendix A). The black nodes {1, ..., 29} represent load nodes, while the red nodes {30, ..., 39} are power generators. The generators are labeled according to the numbers in the red brackets. The red cross denotes the location of the fault. Panel (**b**) plots the behavior of the phases and frequencies of generators {2, ..., 10} after the occurrence of the fault. The onset time of the fault is $t = 2$ s and the fault duration is 0.5 s (red area in the plots). At time $t = 2.5$ s the fault is cleared. The phase and frequency of generator 1 (not shown) are fixed to a constant (see "Methods"). The left plots of the panel (**c**) show the behavior of the phases and frequencies of generators {2, ..., 10} after the application of the data-driven control input (4). The duration of the control action is 0.1 s (green area in the plots) which corresponds to a control horizon $T = 400$ for the discretized network dynamics with sampling period $T_s = 2.5 \times 10^{-4}$ s. For the computation of the control input, we employ $N = 4000$ experimental data collected offline by perturbing the state of the generators locally around its steady-state value (see "Methods"). We use weighting matrices $\mathbf{R} = \mathbf{I}$ and $\mathbf{Q} = \varepsilon \mathbf{I}$, with $\varepsilon = 0.01$. The insets illustrate the behavior of the phases and frequencies during the control. The right plots of the panel (**c**) show the asymptotic behavior of the phases and frequencies of generators {2, ..., 10} after the application of the data-driven control (4).

Data-driven fault recovery in power-grid networks. We address the problem of restoring the normal operation of a power-grid network after the occurrence of a fault that desynchronizes part of the grid. If not mitigated in a timely manner, such desynchronization instabilities may trigger cascading failures that can ultimately cause major blackouts and grid disruptions[54–56]. In our case study, we consider a line fault in the New England power-grid network comprising 39 nodes (29 load nodes and 10 generator nodes), as depicted in Fig. 5a, and we compute an optimal point-to-point control from data to recover the correct operation of the grid. A similar problem is solved in ref. [55] using a more sophisticated control strategy which requires knowledge of the network dynamics. As in refs. [54,55], we assume that the phase δ_i and the (angular) frequency ω_i of each generator i obey the swing

equation dynamics with the parameters given in ref. [54] (except for generator 1 whose phase and frequency are fixed to a constant, cf. "Methods"). Initially, each generator operates at a locally stable steady-state condition determined by the power-flow equations. At time $t = 2$ s, a three-phase fault occurs in the transmission line connecting nodes 16 and 17. After 0.5 s, the fault is cleared; however, the generators have lost synchrony and deviate from their steady-state values (Fig. 5b). To recover the normal behavior of the grid, 0.5 s after the clearance of the fault, we apply a short, optimal control input to the frequency of the generators to steer the state (phase and frequency) of the generators back to its steady-state value. The input is computed from data via Eq. (4) using $N = 4000$ input/state experiments collected by locally perturbing the state of the generators around its normal operation
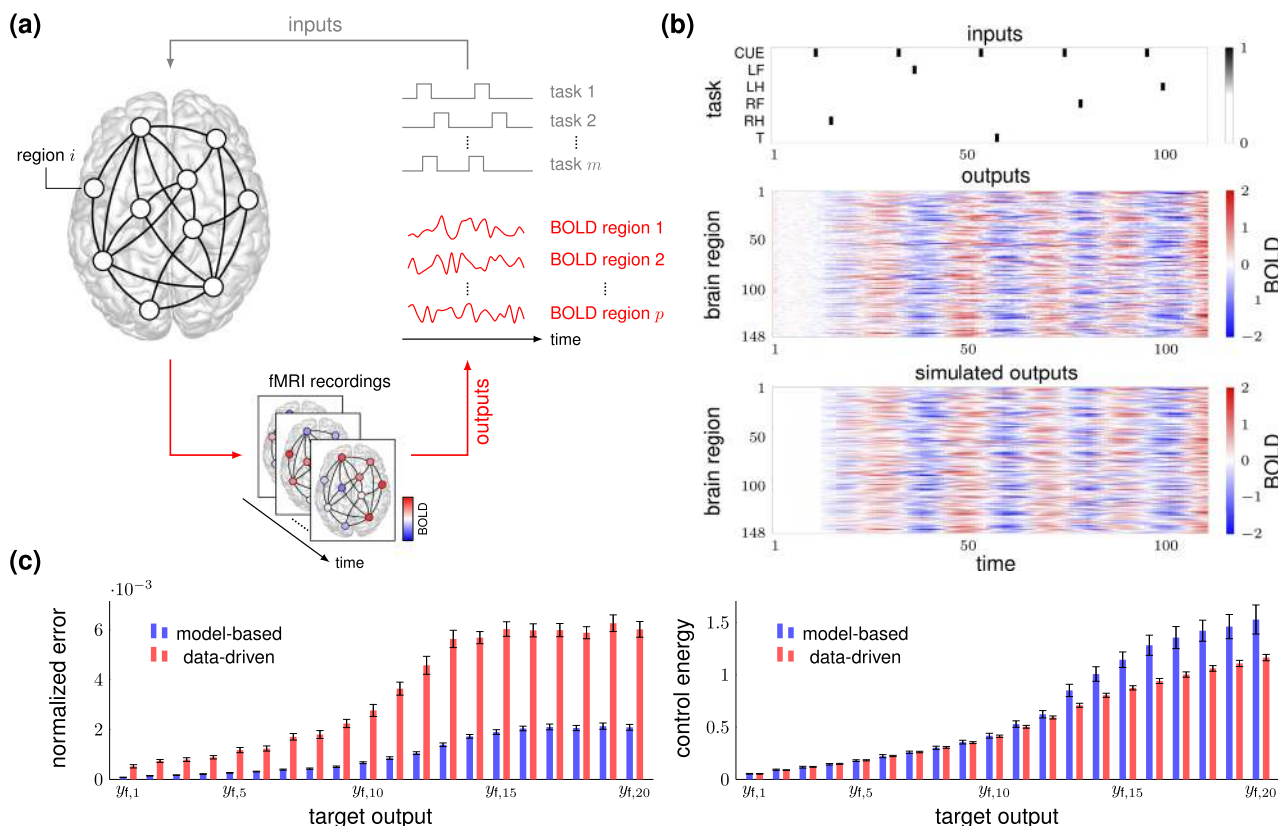
Fig. 6 Data-driven control of functional brain networks. Panel (**a**) provides a schematic of the experimental setup. A set of external stimuli represented by m different task commands induce brain activity. Functional magnetic resonance (fMRI) blood oxygen level-dependent (BOLD) signals are measured and recorded at different times and converted into p time series, one for each brain region. The top and center heatmaps of the panel (**b**) show the inputs and outputs, respectively, for the first 110 measurements of one subject of the HCP dataset. The inputs are divided into $m = 6$ channels corresponding to different task conditions, i.e., CUE (a visual cue preceding the occurrence of other task conditions), LF (squeeze left toe), LH (tap left fingers), RF (squeeze right toe), RH (tap right finger), and T (move tongue). As in ref. [58], each input is a binary 0–1 signal taking the value 1 when the corresponding task condition is issued and 0 otherwise. The outputs represent the BOLD signals of the $p = 148$ brain regions obtained from and enumerated according to the Destrieux 2009 atlas[59]. The bottom heatmap of the panel (**b**) displays the simulated outputs obtained by exciting the approximate low-dimensional linear model of ref. [58] with the input sequence of the top plot. In panel (**c**), we compare the performance of the data-driven and model-based strategy, assuming that the dynamics obey the above-mentioned approximate linear model. We set the control horizon to $T = 100$ and generate the data matrices by sliding a time window of size T across the data samples. The target state $\mathbf{y}_{f,i}$ is the eigenvector associated with the i-th eigenvalue of the empirical Gramian matrix $\hat{\mathcal{W}}_T = \hat{\mathcal{C}}_T^{\mathsf{T}} \hat{\mathcal{C}}_T$, where $\hat{\mathcal{C}}_T = \mathbf{Y}_T \mathbf{U}_{0:T-1}^{\dagger}$. The left plot shows the error to reach the targets $\{\mathbf{y}_{f,i}\}_{i=1}^{20}$ using the data-driven minimum-energy input in Eq. (5) and the model-based one. The right plot shows the norm of the two inputs. The colored bars denote the mean over 100 unrelated subjects and the error bars are the 95% confidence intervals around the mean.

point using the real, nonlinear swing dynamics (see also "Methods"). We consider data sampled with period $T_s = 2.5 \times 10^{-4}$ s, and set the control horizon to $T = 400$ time samples (corresponding to 0.1 s), $\mathbf{R} = \mathbf{I}$, and $\mathbf{Q} = \varepsilon \mathbf{I}$ with $\varepsilon = 0.01$ to enforce locality of the controlled trajectories. As shown in Fig. 5c, the data-driven input drives the state of the generators to a point close enough to the starting synchronous solution (left, inset) so as to asymptotically recover the correct operation of the grid (right). We remark that, because of the nonlinearity of the dynamics, the data-driven control input is not able to exactly steer the state the network back to the original synchronous state, but to a point close to it. The latter point, however, falls within the basin of attraction of the synchronous solution. Thus, the control input is able to correctly steer the network to the desired synchronous state, although not in finite time. Notably, as previously discussed, the computation of the control input requires only pre-collected data, is numerically efficient, and optimal (for the linearized dynamics). More generally, this numerical study shows that the data-driven strategy in Eq. (4) could represent a simple, viable, and computationally efficient approach to control complex nonlinear networks around an operating point.

Controlling functional brain networks via fMRI snapshots. We investigate the problem of generating prescribed patterns of activity in functional brain networks directly from task-based functional magnetic resonance imaging (task-fMRI) time series. Specifically, we examine a dataset of task-based fMRI experiments related to motor activity extracted from the Human Connectome Project (HCP)[57] (see Fig. 6a). In these experiments, participants are presented with visual cues that ask them to execute specific motor tasks; namely, tap their left or right fingers, squeeze their left or right toes, and move their tongue. We consider a set of $m = 6$ input channels associated with different task-related stimuli; that is, the motor tasks' stimuli and the visual cue preceding them. As in ref. [58], we encode the input signals as binary time series taking the value of 1 when the corresponding task-related stimulus occurs and 0 otherwise. The output signals consist of minimally pre-processed blood oxygen-level-dependent (BOLD) time series associated with the fMRI measurements at different regions of the brain (see also "Methods"). In our numerical study, we parcellated the brain into $p = 148$ brain regions (74 regions per hemisphere) according to the Destrieux 2009 atlas[59]. Further, as a baseline for comparison, we approximate the dynamics of the

functional network with a low-dimensional ($n = 20$) linear model computed via the approach described in ref. [58], which has been shown to accurately capture the underlying network dynamics. In fact, although it is widely acknowledged that brain dynamics are nonlinear, linear models can provide a reasonable approximation of the actual nonlinear neural trajectories in certain operating conditions[60,61].

In Fig. 6b, we plot the inputs (top) and outputs (center) of one subject for the first sequence of five motor tasks. The bottom plot of the same figure shows the outputs obtained by approximating the network dynamics with the above-mentioned linear model. In Fig. 6c, we compare the performance of the minimum-energy data-driven control in Eq. (5) with the model-based one, assuming that the network obeys the dynamics of the approximate linear model. We choose a control horizon $T = 100$, form the data matrices in Eq. (3) by sliding a window of fixed size T over the available fMRI data, and consider a set of 20 orthogonal targets $\{\mathbf{y}_{f,i}\}_{i=1}^{20}$ corresponding to eigenvectors of the estimated T-step controllability Gramian (see "Methods" for further details). The top plot of Fig. 6c reports the error (normalized by the output dimension) in the final state of the two strategies, while the bottom plot shows the corresponding control energy (that is, the norm of the control input). In the plots, the targets are ordered from the most ($\mathbf{y}_{f,1}$) to the least ($\mathbf{y}_{f,20}$) controllable. The data-driven and the model-based inputs exhibit an almost identical behavior with reference to the most controllable targets. As we shift towards the least controllable targets, the data-driven strategy yields larger errors but, at the same time, requires less energy to be implemented, thus being potentially more feasible in practice. Importantly, since the underlying brain dynamics are not known, errors in the final state are computed using the identified linear dynamical model. It is thus expected that data-driven inputs yield larger errors in the final state than model-based inputs, although these errors may not correspond to control inaccuracies when applying the data-driven inputs to the actual brain dynamics. Ultimately, our numerical study suggests that the data-driven framework could represent a viable alternative to the classic model-based approach[12,46,62] to infer controllability properties of brain networks, and (by suitably modulating the reconstructed inputs) enforce desired functional configurations in a non-invasive manner and without requiring real-time measurements.

Discussion

In this paper, we present a framework to control complex dynamical networks from data generated by non-optimal (and possibly random) experiments. We show that optimal point-to-point controls to reach the desired target state, including the widely used minimum-energy control input, can be determined exactly from data. We provide closed-form and approximate data-based expressions of these control inputs and characterize the minimum number of samples needed to compute them. Further, we show by means of numerical simulations that data-driven inputs present some numerical advantages with respect to classic model-based approaches, and can be used to analyze and manipulate the controllability properties of real networks.

The data-driven expressions derived in this paper are not only theoretically intriguing and practically relevant but they may also provide an alternative set of tools to investigate how different network properties, such as dimension, heterogeneity, and structure, affect controllability. These questions, which are currently being asked in a model-based setting, may find an easier answer in a data-driven framework due to the simplified mathematical expressions of optimal controls. More generally, our framework and results suggest that many network control

problems may be solved by simply relying on experimental data, thus promoting a new, exciting, and practical line of research in the field of complex networks. Because of the abundance of data in modern applications and the computationally appealing properties of data-driven controls, we expect that this new line of research will benefit a broad range of research communities, spanning from engineering to biology, which employs control-theoretic methods and tools to comprehend and manipulate complex networked phenomena.

Some limitations of this study should also be acknowledged and discussed. First, in our work we consider networks governed by linear dynamics. On the one hand, this is a restrictive assumption since many real-world networks are inherently nonlinear. On the other hand, linear models are used successfully to approximate the behavior of nonlinear dynamical networks around desired operating points and capture more explicitly the impact of the network topology. Second, in our numerical studies, we employed routines that are commonly used in engineering and scientific computation. Using higher precision routines can alleviate and possibly alter our numerical results. However, since we use the same routines to compare model-based and data-driven methods, we believe that the comparisons in the paper remain qualitatively valid (although possibly with different values) even when using routines with higher precision. Third, in many cases, a closed-loop control strategy is preferable to a point-to-point one, especially if the control objective is to stabilize an equilibrium when external disturbances corrupt the dynamics. However, we stress that point-to-point controls, in addition to being able to steer the network to arbitrary configurations, are extensively used to characterize the fundamental control properties and limitations in networks of dynamical nodes. For instance, the expressions we provide for point-to-point control can also lead to novel, data-based methods to study the energetic limitations of controlling complex networks[9], select sensors and actuators for optimized estimation and control[63], and design optimized network structures[64]. Notably, model-based solutions to these control-related problems have been fruitfully applied to shed light on the behavior and operation of real (nonlinear) networks[12,60,61,65]. Finally, although we provide data-driven expressions that compensate for the effect of noise in the limit of infinite data, we do not provide nonasymptotic guarantees on the reconstruction error. Overcoming these limitations represent a compelling direction of future work, which can strengthen the relevance and applicability of our data-driven control framework, and ultimately lead to viable control methods for complex networks.

Methods

Model-based expressions of optimal controls. The model-based solution to the problem in Eq. (2) can be written in a batch form as

$$\mathbf{u}_{0:T-1}^{\star} = (\mathbf{I} - \mathbf{K}_{\mathcal{C}_T}(\mathbf{M}\mathbf{K}_{\mathcal{C}_T})^{\dagger}\mathbf{M})\mathcal{C}_T^{\dagger}\mathbf{y}_f, \tag{12}$$

where $\mathcal{C}_T = [\mathbf{CB}\ \mathbf{CAB} \cdots \mathbf{CA}^{T-1}\mathbf{B}]$ is the T-step output controllability matrix of the dynamical network in Eq. (1), $\mathbf{K}_{\mathcal{C}_T}$ denotes a basis of the kernel of $\mathcal{C}_T$, and $\mathbf{M}$ is any matrix satisfying $\mathbf{M}^{\mathsf{T}}\mathbf{M} = \mathcal{H}_T^{\mathsf{T}}\mathbf{Q}\mathcal{H}_T + \mathbf{R}$, with

$$\mathcal{H}_T = \begin{bmatrix} \mathbf{0} & \cdots & \cdots & \mathbf{0} & \mathbf{CB} \\ \vdots & \cdots & \mathbf{0} & \mathbf{CB} & \mathbf{CAB} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{CB} & \mathbf{CAB} & \cdots & \mathbf{CA}^{T-2}\mathbf{B} \end{bmatrix}, \tag{13}$$

and 0 entries denoting $p \times m$ zero matrices. If $\mathbf{Q} = \mathbf{0}$ and $\mathbf{R} = \mathbf{I}$ (minimum-energy control input), Eq. (12) simplifies to $\mathbf{u}_{0:T-1}^{\star} = \mathcal{C}_T^{\dagger}\mathbf{y}_f$. Alternatively, if the network is output controllable, the minimum-energy input can be compactly written as

$$\mathbf{u}^{\star}(t) = \mathbf{B}^{\mathsf{T}}\mathbf{A}^{T-t-1}\mathbf{C}^{\mathsf{T}}\mathcal{W}_T^{-1}\mathbf{y}_f, \quad t = 0, 1, 2, \ldots, T-1. \tag{14}$$

where $\mathcal{W}_T$ denotes the T-step output controllability Gramian of the dynamical

network in Eq. (1)

$$\mathcal{W}_T = \mathcal{C}_T \mathcal{C}_T^{\mathsf{T}} = \sum_{t=0}^{T-1} \mathbf{C}\mathbf{A}^t \mathbf{B}\mathbf{B}^{\mathsf{T}}(\mathbf{A}^{\mathsf{T}})^t \mathbf{C}^{\mathsf{T}}, \tag{15}$$

which is invertible if and only if the network is output controllable. Equation (14) is the classic (Gramian-based) expression of the minimum-energy control input[37]. It is well-known that this expression is numerically unstable, even for moderate size systems[20].

Subspace-based system identification. Given the data matrices $\mathbf{U}_{0:T-1}$ and $\mathbf{Y}_T$ as defined in Eq. (3) and assuming that $\mathbf{C} = \mathbf{I}$, a simple deterministic subspace-based procedure[66, Ch. 6] to estimate the matrices $\mathbf{A}$ and $\mathbf{B}$ from the available data consist of the following two steps:

1. Compute an estimate of the T-step controllability matrix of the network as the solution to the minimization problem

$$\hat{\mathcal{C}}_T = \arg\min_{\mathcal{C}_T} \|\mathbf{Y}_T - \mathcal{C}_T \mathbf{U}_{0:T-1}\|_{\mathrm{F}}^2, \tag{16}$$

where $\|\cdot\|_{\mathrm{F}}$ denotes the Frobenius norm of a matrix. The solution to the problem in Eq. (16) has the form $\hat{\mathcal{C}}_T = \mathbf{Y}_T \mathbf{U}_{0:T-1}^{\dagger}$.

2. In view of the definition of the controllability matrix, obtain an estimate of the matrix $\mathbf{B}$ by extracting the first m columns of $\hat{\mathcal{C}}_T$. Namely, $\hat{\mathbf{B}} = [\hat{\mathcal{C}}_T]_{:,1:m}$, where $[\mathbf{X}]_{:,i:j}$ indicates the sub-matrix of $\mathbf{X}$ obtained from keeping the entries from the ith to jth columns and all of its rows. An estimate of matrix $\mathbf{A}$ can be obtained as the solution to the least-squares problem

$$\hat{\mathbf{A}} = \arg\min_{\mathbf{A}} \left\| [\hat{\mathcal{C}}_T]_{:,m+1:mT} - \mathbf{A}\, [\hat{\mathcal{C}}_T]_{:,1:(T-1)m} \right\|_{\mathrm{F}}^2, \tag{17}$$

which yields the matrix $\hat{\mathbf{A}} = [\hat{\mathcal{C}}_T]_{:,m+1:mT}[\hat{\mathcal{C}}_T]_{:,1:(T-1)m}^{\dagger}$.

If the data are noiseless, the system is controllable in $T-1$ steps, and $\mathbf{U}_{0:T-1}$ has full row rank, then this procedure provably returns correct estimates of $\mathbf{A}$ and $\mathbf{B}$[66].

Power-grid network dynamics, parameters, and data generation. The short-term electromechanical behavior of generators $\{2, \ldots, 10\}$ of the New England power-grid network are modeled by the swing equations[67]:

$$\dot{\delta}_i = \omega_i,$$

$$\frac{H_i}{\pi f_b}\dot{\omega}_i = -D_i\omega_i + P_{\mathrm{m}i} - G_{ii}E_i^2 + \sum_{j=1, j\neq i}^{10} E_iE_j(G_{ij}\cos(\delta_i - \delta_j) + B_{ij}\sin(\delta_i - \delta_j)). \tag{18}$$

where δ_i is the angular position or phase of the rotor in generator i with respect to generator 1, and where ω_i is the deviation of the rotor speed or frequency in generator i relative to the nominal angular frequency $2\pi f_b$. Generator 1 is assumed to be connected to an infinite bus and has constant phase and frequency. The parameters H_i and D_i are the inertia constant and damping coefficient, respectively, of generator i. The parameter G_{ii} is the internal conductance of generator i, and $G_{ij} + iB_{ij}$ (where i is the imaginary unit) is the transfer impedance between generators i and j. The parameter $P_{\mathrm{m}i}$ denotes the mechanical input power of generator i and E_i denotes the internal voltage of generator i. The values of parameters f_b, H_i, D_i, G_{ij}, B_{ij}, and $P_{\mathrm{m}i}$ in the non-faulty and faulty configuration are taken from ref. [54], while the voltages E_i and initial conditions $(\delta_i(0), \omega_i(0) = 0)$ are fixed using a power-flow computation. In our numerical study, we discretize the dynamics in Eq. (18) using a forward Euler method with sampling time $T_s = 2.5 \times 10^{-4}$ s. Data are generated by applying a Gaussian i.i.d. perturbation with zero mean and variance 0.01 to each frequency ω_i of the swing dynamics in Eq. (18). The initial condition of each experiment is computed by adding a Gaussian i.i.d. perturbation with zero mean and variance 0.01 to the steady-state values of δ_i and ω_i of the swing dynamics in Eq. (18).

Task-fMRI dataset, pre-processing pipeline, and identification setup. The motor task-fMRI data used in our numerical study are extracted from the HCP S1200 release[57,68]. The details for data acquisition and experiment design can be found in ref. [68]. The BOLD measurements have been pre-processed according to the minimal pipeline described in ref. [69], and, as in ref. [58], filtered with a band-pass filter to attenuate the frequencies outside the 0.06–0.12 Hz band. Specifically, we use an order 50 FIR-type filter using Matlab® equiripple method, so as to achieve a 20 dB attenuation outside the passband. The initial stop and pass frequencies considered were $f_{s,1} = 0.04$ Hz, $f_{p,1} = 0.06$ Hz, and the final stop and pass frequencies were $f_{p,2} = 0.12$ Hz and $f_{s,2} = 0.15$ Hz, respectively. Further, as common practice, the effect of the physiological signals (cardiac, respiratory, and head motion signals) is removed from the BOLD measurements by means of standard regression procedure[58]. The data matrices in Eq. (3) are generated via a sliding window of fixed length $T = 100$ with initial time in the interval $[-90, 10]$. We assume that the inputs and states are zero for times less than or equal to 10, i.e., the instant at which the first task condition is issued. We approximate the input–output dynamics with a linear model with state dimension $n = 20$ computed using input–output data in the interval $[0, 150]$ and the subspace-based

identification procedure detailed in ref. [58]. In particular, we use (Hankel) output data matrices with columns consisting of $s = 3$ output samples and a regularization term $\gamma\mathbf{I}$, $\gamma = 5$, in the regression procedure for the estimation of matrix $\mathbf{B}$. When the estimated network matrix $\mathbf{A}$ has unstable eigenvalues, we stabilize $\mathbf{A}$ by diving it by $\rho(\mathbf{A}) + 0.01$, where $\rho(\mathbf{A})$ denotes the spectral radius of $\mathbf{A}$.

Additional computational and experimental details. All numerical simulations have been performed via standard linear-algebra LAPACK routines available as built-in functions in Matlab® R2019b, running on a 2.6 GHz Intel Core i5 processor with 8 GB of RAM. In particular, for the computation of pseudoinverses, we use the singular value decomposition method (command pinv in Matlab®) with a threshold of 10^{-8}. In the numerical simulations of Figs. 2 and 3, if not otherwise stated, the entries of the final state $\mathbf{y}_f$ and those of the input data matrix $\mathbf{U}_{0:T-1}$ are standard normal i.i.d. variables, the input/output nodes are randomly selected with the only constraint that the resulting system is output controllable, and the curves represent the average over 500 independent realizations of networks, data, input/output nodes, and final states. We ensure output controllability by choosing networks that are connected and by choosing sets of input/output nodes that yield the smallest singular value of the resulting output controllability matrix no smaller than 10^{-10}.

Data availability
The New England power-grid interconnection scheme can be found in Appendix A of the reference textbook ref. [70] and the grid parameters in the faulty and non-faulty configurations are described in ref. [54]. The HCP data used in our study are part of the 1200 Subjects Release (S1200) and are publicly available on the ConnectomeDB database (https://db.humanconnectome.org). These data are also available in the public GitHub repository: https://github.com/baggiogi/data_driven_control.

Code availability
The code used in this study is freely available in the public GitHub repository: https://github.com/baggiogi/data_driven_control.

References
1. Levine, S., Pastor, P., Krizhevsky, A., Ibarz, J. & Quillen, D. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *Int. J. Robot. Res.* **37**, 421–436 (2018).
2. Marx, V. Biology: the big challenges of big data. *Nature* **498**, 255–260 (2013).
3. Sejnowski, T. J., Churchland, P. S. & Movshon, J. A. Putting big data to good use in neuroscience. *Nat. Neurosci.* **17**, 1440 (2014).
4. Einav, L. & Levin, J. Economics in the age of big data. *Science* **346**, 1243089 (2014).
5. Turk-Browne, N. B. Functional interactions as big data in the human brain. *Science* **342**, 580–584 (2013).
6. Bose, A. Smart transmission grid applications and their supporting infrastructure. *IEEE Trans. Smart Grid* **1**, 11–19 (2010).
7. Lv, Y., Duan, Y., Kang, W., Li, Z. & Wang, F.-Y. Traffic flow prediction with big data: a deep learning approach. *IEEE Trans. Intell. Transportation Syst.* **16**, 865–873 (2014).
8. Liu, Y. Y., Slotine, J. J. & Barabási, A. L. Controllability of complex networks. *Nature* **473**, 167–173 (2011).
9. Pasqualetti, F., Zampieri, S. & Bullo, F. Controllability metrics, limitations and algorithms for complex networks. *IEEE Trans. Control Netw. Syst.* **1**, 40–52 (2014).
10. Bof, N., Baggio, G. & Zampieri, S. On the role of network centrality in the controllability of complex networks. *IEEE Trans. Control Netw. Syst.* **4**, 643–653 (2017).
11. Yan, G. et al. Spectrum of controlling and observing complex networks. *Nat. Phys.* **11**, 779–786 (2015).
12. Gu, S. et al. Controllability of structural brain networks. *Nat. Commun.* **6**, 1–10 (2015).
13. Liu, Y.-Y. & Barabási, A.-L. Control principles of complex systems. *Rev. Mod. Phys.* **88**, 035006 (2016).
14. Lindmark, G. & Altafini, C. Minimum energy control for complex networks. *Sci. Rep.* **8**, 3188–3202 (2018).
15. Gonçalves, J. & Warnick, S. Necessary and sufficient conditions for dynamical structure reconstruction of lti networks. *IEEE Trans. Autom. Control* **53**, 1670–1674 (2008).
16. Shandilya, S. G. & Timme, M. Inferring network topology from complex dynamics. *N. J. Phys.* **13**, 013004 (2011).

17. Angulo, M. T., Moreno, J. A., Lippner, G., Barabási, A.-L. & Liu, Y.-Y. Fundamental limitations of network reconstruction from temporal data. *J. R. Soc. Interface* **14**, 20160966 (2017).

18. Achlioptas, D., Clauset, A., Kempe, D. & Moore, C. On the bias of traceroute sampling: or, power-law degree distributions in regular graphs. *J. ACM* **56**, 1–28 (2009).

19. Handcock, M. S. & Gile, K. J. Modeling social networks from sampled data. *Ann. Appl. Stat.* **4**, 5 (2010).

20. Sun, J. & Motter, A. E. Controllability transition and nonlocality in network control. *Phys. Rev. Lett.* **110**, 208701 (2013).

21. Wang, L.-Z., Chen, Y.-Z., Wang, W.-X. & Lai, Y.-C. Physical controllability of complex networks. *Sci. Rep.* **7**, 40198 (2017).

22. Levine, S., Finn, C., Darrell, T. & Abbeel, P. End-to-end training of deep visuomotor policies. *J. Mach. Learn. Res.* **17**, 1334–1373 (2016).

23. Silver, D. et al. Mastering the game of Go without human knowledge. *Nature* **550**, 354 (2017).

24. Gevers, M. Identification for control: from the early achievements to the revival of experiment design. *Eur. J. Control* **11**, 1–18 (2005).

25. Brunton, S. L. & Kutz, J. N. *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control* (Cambridge University Press, 2019).

26. Lewis, F. L., Vrabie, D. & Vamvoudakis, K. G. Reinforcement learning and feedback control: using natural decision methods to design optimal adaptive controllers. *IEEE Control Syst. Mag.* **32**, 76–105 (2012).

27. Recht, B. A tour of reinforcement learning: the view from continuous control. *Ann. Rev. Control, Robotics, Auton. Syst.* **2**, 253–279 (2018).

28. Bristow, D. A., Tharayil, M. & Alleyne, A. G. A survey of iterative learning control. *IEEE control Syst. Mag.* **26**, 96–114 (2006).

29. Åström, K. J. & Wittenmark, B. On self tuning regulators. *Automatica* **9**, 185–199 (1973).

30. Markovsky, I. & Rapisarda, P. Data-driven simulation and control. *Int. J. Control* **81**, 1946–1959 (2008).

31. Persis, C. D. & Tesi, P. Formulas for data-driven control: stabilization, optimality and robustness. *IEEE Trans. Autom. Control* **65**, 909–924 (2020).

32. Bertsekas, D. P. & Tsitsiklis, J. N. *Neuro-dynamic Programming*, Vol. 5 (Athena Scientific Belmont, 1996).

33. Åström, K. J. & Hägglund, T. *PID Controllers: Theory, Design, and Tuning*, Vol. 2 (Instrument Society of America Research Triangle Park, 1995).

34. Gao, J., Liu, Y.-Y., D'Souza, R. M. & Barabási, A. L. Target control of complex networks. *Nat. Commun.* **5**, 5415 (2014).

35. Klickstein, I., Shirin, A. & Sorrentino, F. Energy scaling of targeted optimal control of complex networks. *Nat. Commun.* **8**, 15145 (2017).

36. Franklin, G. F., Powell, J. D. & Emami-Naeini, A. *Feedback Control of Dynamic Systems* (Prentice Hall Upper Saddle River, 1997).

37. Kailath, T. *Linear Systems* (Prentice-Hall, 1980).

38. Dean, S., Mania, H., Matni, N., Recht, B. & Tu, S. On the sample complexity of the linear quadratic regulator. *Found. Comput. Math.* **20**, 633–679 (2019).

39. Ben-Israel, A. & Greville, T. N. E. In *Generalized Inverses: Theory and Applications*, Vol. 15 of *CMS Books in Mathematics*, 2nd edn. (Springer-Verlag New York, 2003).

40. Paré, P. E., Chetty, V. & Warnick, S. On the necessity of full-state measurement for state-space network reconstruction. in *2013 IEEE Global Conference on Signal and Information Processing*, 803–806 (IEEE, 2013).

41. Osipov, G. V., Kurths, J. & Zhou, C. *Synchronization in Oscillatory Networks* (Springer, 2007).

42. Tang, Y., Qian, F., Gao, H. & Kurths, J. Synchronization in complex networks and its application-a survey of recent advances and challenges. *Annu. Rev. Control* **38**, 184–198 (2014).

43. Lakatos, P. et al. An oscillatory hierarchy controlling neuronal excitability and stimulus processing in the auditory cortex. *J. Neurophysiol.* **94**, 1904–1911 (2005).

44. Schnitzler, A. & Gross, J. Normal and pathological oscillatory communication in the brain. *Nat. Rev. Neurosci.* **6**, 285 (2005).

45. Womelsdorf, T. et al. Modulation of neuronal interactions through neuronal synchronization. *Science* **316**, 1609–1612 (2007).

46. Menara, T., Baggio, G., Bassett, D. S. & Pasqualetti, F. A framework to control functional connectivity in the human brain. In *IEEE Conference on Decision and Control*, 4697–4704 (Nice, 2019).

47. Dörfler, F., Chertkov, M. & Bullo, F. Synchronization in complex oscillator networks and smart grids. *Proc. Natl Acad. Sci. USA* **110**, 2005–2010 (2013).

48. Dörfler, F. & Bullo, F. Synchronization in complex networks of phase oscillators: a survey. *Automatica* **50**, 1539–1564 (2014).

49. Skardal, P. S. & Arenas, A. Control of coupled oscillator networks with application to microgrid technologies. *Sci. Adv.* **1**, e1500339 (2015).

50. Lehnert, J., Hövel, P., Selivanov, A., Fradkov, A. & Schöll, E. Controlling cluster synchronization by adapting the topology. *Phys. Rev. E* **90**, 042914 (2014).

51. Hart, J. D., Zhang, Y., Roy, R. & Motter, A. E. Topological control of synchronization patterns: trading symmetry for stability. *Phys. Rev. Lett.* **122**, 058301 (2019).

52. Wiley, D. A., Strogatz, S. H. & Girvan, M. The size of the sync basin. *Chaos: Interdisc. J. Nonlinear Sci.* **16**, 015103 (2006).

53. Dhagash, M., Daleo, N. S., Dörfler, F. & Hauenstein, J. D. Algebraic geometrization of the Kuramoto model: equilibria and stability analysis. *Chaos: Interdiscip. J. Nonlinear Sci.* **25**, 053103 (2015).

54. Susuki, Y., Mezić, I. & Hikihara, T. Coherent swing instability of power grids. *J. Nonlinear Sci.* **21**, 403–439 (2011).

55. Cornelius, S. P., Kath, W. L. & Motter, A. E. Realistic control of network dynamics. *Nat. Commun.* **4**, 1–9 (2013).

56. Simpson-Porco, J. W., Dörfler, F. & Bullo, F. Voltage collapse in complex power grids. *Nat. Commun.* **7**, 1–8 (2016).

57. Van Essen, D. C. et al. The WU-Minn human connectome project: an overview. *Neuroimage* **80**, 62–79 (2013).

58. Becker, C. O., Bassett, D. S. & Preciado, V. M. Large-scale dynamic modeling of task-fMRI signals via subspace system identification. *J. Neural Eng.* **15**, 066016 (2018).

59. Destrieux, C., Fischl, B., Dale, A. & Halgren, E. Automatic parcellation of human cortical gyri and sulci using standard anatomical nomenclature. *Neuroimage* **53**, 1–15 (2010).

60. Yan, G. et al. Network control principles predict neuron function in the *Caenorhabditis elegans* connectome. *Nature* **550**, 519 (2017).

61. Kim, J. Z. & Bassett, D. S. Linear dynamics and control of brain networks. in *Neural Engineering*, 497–518 (Springer, 2020).

62. Deng, S. & Gu, S. Controllability analysis of functional brain networks. Preprint at http://arxiv.org/abs/2003.08278 (2020).

63. Summers, T. H., Cortesi, F. L. & Lygeros, J. On submodularity and controllability in complex dynamical networks. *IEEE Trans. Control Netw. Syst.* **3**, 91–101 (2016).

64. Zhao, S. & Pasqualetti, F. Networks with diagonal controllability gramians: analysis, graphical conditions, and design algorithms. *Automatica* **102**, 10–18 (2019).

65. Vinayagam, A. et al. Controllability analysis of the directed human protein interaction network identifies disease genes and drug targets. *Proc. Natl Acad. Sci. USA* **113**, 4976–4981 (2016).

66. Katayama, T. Subspace methods for system identification. In *Communications and Control Engineering* (Springer-Verlag London, 2005).

67. Kundur, P. *Power System Stability and Control* (McGraw-Hill, 1994).

68. WU-Minn, HCP 1200 subjects data release reference manual. https://www.humanconnectome.org (2017).

69. Glasser, M. F. et al. The minimal preprocessing pipelines for the human connectome project. *Neuroimage* **80**, 105–124 (2013).

70. Pai, M. A. *Energy Function Analysis for Power System Stability* (Kluwer Academic Publishers, 1989).

71. Barabási, A. L. & Albert, R. Emergence of scaling in random networks. *Science* **286**, 509–512 (1999).

72. Watts, D. J. & Strogatz, S. H. Collective dynamics of 'small-world' networks. *Nature* **393**, 440–442 (1998).

Author contributions

G.B., D.S.B. and F.P. contributed to the conceptual and theoretical aspects of the study, wrote the paper and the Supplementary Information. G.B. carried out the numerical studies and prepared the figures.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41467-021-21554-0.

Correspondence and requests for materials should be addressed to F.P.

Peer review information *Nature Communications* thanks Isaac Klickstein, and the other, anonymous, reviewer(s) for their contribution to the peer review of this work. Peer reviewer reports are available.

Reprints and permission information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.