SCIENTIFIC REPORTS

Optimizing Dynamical Network Structure for Pinning Control

Yasin Orouskhani[1], Mahdi Jalili[2] & Xinghuo Yu[2]

Controlling dynamics of a network from any initial state to a final desired state has many applications in different disciplines from engineering to biology and social sciences. In this work, we optimize the network structure for pinning control. The problem is formulated as four optimization tasks: *i*) optimizing the locations of driver nodes, *ii*) optimizing the feedback gains, *iii*) optimizing simultaneously the locations of driver nodes and feedback gains, and *iv*) optimizing the connection weights. A newly developed population-based optimization technique (cat swarm optimization) is used as the optimization method. In order to verify the methods, we use both real-world networks, and model scale-free and small-world networks. Extensive simulation results show that the optimal placement of driver nodes significantly outperforms heuristic methods including placing drivers based on various centrality measures (degree, betweenness, closeness and clustering coefficient). The pinning controllability is further improved by optimizing the feedback gains. We also show that one can significantly improve the controllability by optimizing the connection weights.

Networked structures are found everywhere in our daily life; there exist a network system where a kind of information exchange is happening[1,2]. Networks are comprised of two or more nodes that are connected through directed/undirected and weighted/unweighted edges. Examples of everyday life complex network systems include the World Wide Web, the Internet, power grids, transportation and water distribution networks. Complex networks have different definitions, but in the most valid definition we call a network as complex when it has non-trivial structure[3]. A series of research works in network science are based on considering specific dynamics on the nodes and study collective dynamics arising from complex interaction networks[4–6]. It has been shown that if dynamical systems interact over a network and some conditions are met, a collective behavior (synchronization or consensus) emergences.

Controlling dynamics of a network from any initial state to a final desired state has many applications in different fields, which has been heavily studied in network science community[7–11]. For example, modern power grids with many generators and consumers are required to be controlled to have desirable performance. The ease by which the whole network can be pinned to a reference state is denoted as its pinning controllability[12–14]. This is often performed by choosing some of the nodes as drivers to which the control signal is fed. The master stability function formalism, developed to study linear stability of the synchronization manifold in coupled identical dynamical systems[15], is a proper tool to study pinning controllability. This allows use spectral properties of augmented Laplacian matrix, which includes the information on the control signal towards analyzing pinning controllability. Based on such information, the eigenratio of the augmented Laplacian, i.e., the largest eigenvalue divided by the smallest one, was proposed as a metric to quantify the pinning controllability of dynamical networks, i.e., the smaller the eigenratio the better the controllability[12,16].

For some applications it might be desirable to have a network with high levels of pinning control. One of the important issues in optimal pinning control is to find the best driver nodes. Early works in this field selected the drivers randomly and showed that having only a small number of drivers can successfully control the whole dynamics into a reference state[12]. It was shown that controlling central nodes (e.g., those with high degree or betweenness centrality measure) is more effective than randomly selecting the drivers. Zhuo *et al.* considered the speed of synchronization to the reference trajectory, which is related to the smallest eigenvalue of the augmented Laplacian matrix[17]. Using linear matrix inequality approach, they introduced an optimization method to obtain the optimal drivers and their associated feedback gains to maximize this eigenvalue. They showed that by driving the nodes with high gains, the speed of pinning control is significantly improved as compared to the cases when nodes with high degree or betweenness centrality are chosen as drivers[17].

[1]Department of Computer Engineering, Sharif University of Technology, Tehran, Iran. [2]Department of Electrical and Computer Engineering, School of Engineering, RMIT University, Melbourne, Australia. Correspondence and requests for materials should be addressed to M.J. (email: mahdi.jalili@rmit.edu.au)

We have recently proposed an optimization method to find optimal drivers and shown that it significantly outperforms heuristic approaches[16]. In this paper, we extended the previous work by further optimizing the network structure. Not only the locations of optimal driver nodes are found through an optimization technique, but also the feedback gains are optimized. We showed that optimizing the feedback gains have significant impact on the controllability. We further showed that the connection weights can also be optimized using the same optimization techniques, and the networks with optimal connection weighted have much higher controllability than unweighted ones.

Pinning Controllability: Background Information

Controlling a complex network system has many applications ranging from engineering systems (such as power grids) to biological systems. One type of controlling a complex network is pinning control in which the dynamics of the nodes are pinned to a specific state. The ease by which a network can be controlled is often interpreted as its pinning controllability[18,19]. Let us consider a network with N nodes, in which a dynamical system seats at each node. These individual dynamical systems interact through the edges of the network and the equations of the motion read:

$$\frac{d\boldsymbol{x}_i}{dt} = F(\boldsymbol{x}_i) - \sigma\sum_{j=1}^{N}l_{ij}H\boldsymbol{x}_j; \;\; i = 1, 2, \ldots, N,$$

(1)

where $x_i \in R^d$ are d-dimensional state vectors, $F:R^d \rightarrow R^d$ defines the individual dynamics of each node, l_{ij} is the element (i, j) of Laplacian matrix which describes the network topology and can be obtained from the adjacency matrix. In this work, we limit ourselves to simple (unweighted and undirected) networks, and thus entries of the adjacency matrix $A = (a_{ij})$ are either 1, if there is a link between i and j, and 0 otherwise. In order to construct the Laplacian, one has to obtain $l_{ij} = -a_{ij}$ for $i \neq j$, and $l_{ii} = k_i$, where k_i is degree of node i. σ is a unified coupling strength and H is a matrix indicating from which dimensions the dynamical systems are coupled to each other; for example, if the individual dynamical systems have three states that are connected through their first state, H is a 3-by-3 matrix whose $(1, 1)$ entry is one and others are zero.

As we mentioned before, the goal of pinning control is to lead the nodes into a reference state. Let us consider a time-varying reference state as

$$\frac{d\boldsymbol{s}(t)}{dt} = F(\boldsymbol{s}(t)).$$

(2)

In order to pin the nodes of network to the reference trajectory $s(t)$, often some of the nodes are chosen as drivers and the control signal is fed to them. Applying the control signal to the network, the dynamical equations read

$$\frac{dx_i}{dt} = F(x_i) - \sigma\sum_{j=1}^{N}l_{ij}Hx_j + \sigma\beta_i g_i H(s - x_i); \;\; i = 1, 2, \ldots, N,$$

(3)

where g_i is the feedback control gain, and β_i indicates the driver nodes such that $\beta_i = 1$, if node i is a driver node and $\beta_i = 0$ otherwise. If certain conditions are met, the driver nodes will be able to force the network to follow the reference trajectory, and a local synchronization to this manifold is obtained provided that the initial conditions are close enough, i.e., there exists $\varepsilon > 0$ such that for the initial conditions with distance less than ε, one has:

$$\left\|\boldsymbol{x}_i(t) - \boldsymbol{s}(t)\right\| \xrightarrow[t\rightarrow\infty]{} 0 \;\; \forall \;\;\; i = 1, \ldots N.$$

(4)

Complete synchronization to $s(t)$ can only be attained when the individual dynamical systems are identical and some conditions are met. In general, four causes have roles in determining whether synchronization can be obtained: i) the dynamics of the individual systems, expressed by $F(\cdot)$ in equation (1), ii) the network structure, represented by the connection graph described by A or L, iii) the type and strength of the interaction between the individual dynamical systems, and iv) the driver nodes and their associated feedback control gains. In studying pinning controllability of dynamical networks, the main problem is to study the stability of the solution $\boldsymbol{x}_1(t) = \boldsymbol{x}_2(t) = \ldots = \boldsymbol{x}_N(t) = \boldsymbol{s}(t)$. To this end, the master stability function mechanism, which has been originally proposed for synchronization of dynamical networks[15], can be efficiently used. The master stability function approach gives necessary conditions for the local stability of the above synchronized solution. This approach is based on studying the variational equations; each dynamical system is considered to have small enough perturbation $\boldsymbol{\zeta}_i$ from the synchronous state; $\boldsymbol{x}_i(t) = \boldsymbol{s}(t) + \boldsymbol{\zeta}_i$. The variational equations are

$$\dot{\boldsymbol{\zeta}}_i = DF(\boldsymbol{s})\boldsymbol{\zeta}_i - \sigma\sum_{\substack{j=1\\j\neq i}}^{N}c_{ij}H\boldsymbol{\zeta}_j; \;\; i = 1, 2, \ldots, N,$$

(5)

where D stands for Jacobian and C is the augmented Laplacian matrix obtained as

$$C = \{c_{ij}\} = \begin{pmatrix} l_{11} + g_1\beta_1 & l_{12} & \cdots & l_{1N} \\ l_{21} & l_{11} + g_2\beta_2 & \cdots & l_{2N} \\ \vdots & \vdots & \cdots & \vdots \\ l_{N1} & l_{N2} & \cdots & l_{NN} + g_N\beta_N \end{pmatrix}.$$

(6)

C has the information on the drivers (their locations and associated feedback gain) in its diagonal elements. Here, we consider undirected networks; thus C is a symmetric matrix ($C = C^T$), and it can be transformed into $C = \Gamma\Omega\Gamma^T$, where Γ is the orthogonal matrix with columns as the eigenvectors of C and Ω is a diagonal matrix of corresponding real eigenvalues. Let define $\zeta = (\zeta_1, \zeta_2, \ldots, \zeta_N) = \eta\Gamma^T$, where $\eta = (\eta_1, \eta_2, \ldots, \eta_N)$. Then, equation (5) is equivalent to

$$\dot{\eta}_i = DF\eta_i - \sigma\lambda_i DH\eta_i; \quad i = 1, 2, \ldots, N,$$

(7)

where λ_i are the eigenvalues of C, ordered as $0 < \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_N$[20]. Since C is a symmetric matrix, all its eigenvalues take real values. A necessary condition for the stability of the synchronization manifold is that all the Lyapunov exponents of equation (7) are negative. The largest Lyapunov exponent of the variational equation (7), $\Lambda(a)$ where $a = \sigma\lambda$, called master stability function, accounts for the linear stability of the synchronization solution, i.e., if $\Lambda(a) < 0$, the synchronized state is stable. For many systems, the master stability function is negative only within an interval (a_1, a_2) where and hence, the network synchronizes to the reference trajectory in such an interval[15]. Requiring all coupling strengths to lie within this interval, i.e. $a_1 < \sigma\lambda_1 \leq \ldots \leq \sigma\lambda_N < a_2$, one concludes that if the network locally asymptotically synchronizes to the reference trajectory, we have

$$R = \frac{\lambda_N}{\lambda_1} < \frac{a_2}{a_1},$$

(8)

for the corresponding graph.

The left hand side of the above equation depends only the network structure and information on the drivers, whereas the right hand side depends on the individual dynamical systems and the projection matrix. R is accounted for pinning controllability of the network; i.e., for network N_1 to be more pinning controllable than network N_2, N_1 should have smaller R as compared to N_2 (provided that they have the same number of nodes and connection cost). Therefore, in order to improve the controllability one has to minimize R.

Pinning Controllability: Optimization

Considering R as an indicator of network controllability, it depends not only on the network topology, but also on the number of driver nodes, their locations in the network and feedback gains. This way only topology of the network is important in determining its pinning control properties, and thus, controllability optimization will be performed through structural optimization of networks. In this research, we consider different scenarios to optimize R.

Optimizing the locations of driver nodes (Task 1).

One of the key issues in optimizing the network structure to have high levels of pinning control over the network is the location of driver nodes. Often, there is a limited budget for control, and thus not all nodes can be driven. Indeed, it is desired to have the control done with as small number of drivers as possible. In early works, the locations of drivers were considered to be random[12], where considering only a few drivers could successfully pin the whole network to the reference trajectory. A better way is to drive nodes with high centrality values, e.g., high-degree nodes[12], where it has been shown that by selecting the high-degree nodes as drivers, better control performance can be achieved as compared to the case when drivers are randomly chosen. We have recently shown that one can further optimize the pinning control performance by finding the optimal locations for drivers[16]. It was shown that properties of optimal drivers significantly depend on the network type; for heterogeneous networks, optimal drivers have characteristics close to hub nodes, whereas they have properties similar to network average in homogeneous networks.

The problem of finding the best drivers can be defined as a constrained optimization problem with specific cost function. In this part, let suppose that all drivers have the same feedback gain. When the drivers are chosen based on their centrality values, first the nodes are sorted based on their centrality scores (e.g., degree or betweenness centrality), and then $N_d (N_d \ll N)$ nodes with the highest centrality is considered to be drivers. Here we compare the optimization method with four heuristics including choosing the drivers based on their degree, betweenness centrality, closeness or clustering coefficient. As the feedback gains are considered to be the same, the only parameters that should be determined in the optimization process are β_i's that are 1 when the node is driver and 0 otherwise. Furthermore, the total number of driver nodes is N_d. Thus, the constrained optimization problem to be solved can be formulated as

$$\min_{\beta_i} R(\boldsymbol{\beta}) = \frac{\lambda_N(\boldsymbol{\beta})}{\lambda_1(\boldsymbol{\beta})}$$

$$subject\ to \sum_{N}^{i=1} \beta_i = N_d\ and\ \beta_i \in \{0, 1\}$$

(9)

Network	#Nodes	#Edges	Avg. Deg.	Avg. Betweenness	Avg. Closeness	Avg. Clustering
IEEE$_{14}$	14	20	2.85	8.92	0.0332	0.36
IEEE$_{30}$	30	41	2.73	33.43	0.0107	0.23
IEEE$_{57}$	57	78	2.73	110.70	0.0037	0.12
IEEE$_{118}$	118	179	3.03	310.55	0.0014	0.16
IEEE$_{300}$	300	408	2.73	1305.8	0.00035	0.08
France	146	223	3.05	406.58	0.0011	0.27
Spain	98	175	3.57	189.96	0.0022	0.31
Iran	105	142	2.7	305.84	0.0015	0.1
USA	4941	6594	2.6	44433	0.00001	0.08

Table 1. IEEE Benchmarks and Power Grids Details.

Parameter	Value
SMP	5
SPC	True
SRD	20%
CDC	80%
MR	2%
Population Size	250
Realization	20
Max-Iteration	500

Table 2. Parameter Settings for CSO method.

where $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_N)$ are the parameters to be optimized. For this task, all drivers have the same feedback gain of $k_i = 10$.

Optimizing the feedback gains (Task 2). As mentioned, not only the locations of drivers are important in determining pinning control properties of a dynamical network, but also the feedback gains play a major role. In this section, we suppose that the locations of optimal drivers are fixed, and determine the optimal value for the feedback gains. Therefore, the only free parameters for the optimization process are g_i's which are considered to be in a certain interval. Thus, the optimization problem is

$$\min_{g_i} R(g) = \frac{\lambda_N(\boldsymbol{g})}{\lambda_1(\boldsymbol{g})}$$
$$subject\ to\ \alpha \leq g_i \leq \gamma \tag{10}$$

where $\boldsymbol{g} = (g_1, g_2, \ldots, g_{N_d})$ are the parameters that should be optimized. In this manuscript, the feedback gains are limited to the interval [0, 100], which means that optimal feedback gains will be in this range.

Optimizing the locations of driver nodes and feedback gains (Task 3). In this section we do not fix the locations of driver nodes but let the optimization process simultaneously finds the optimal locations and their associated feedback gain. Fixing the number of drivers at N_d, β_i's and g_i's are chosen as free parameters that should be determined by the optimization process. Therefore, the optimization problem can be formulated as

$$\min_{\beta_i, g_i} R(\boldsymbol{g}, \boldsymbol{\beta}) = \frac{\lambda_N(\boldsymbol{g}, \boldsymbol{\beta})}{\lambda_1(\boldsymbol{g}, \boldsymbol{\beta})}$$
$$subject\ to\ \sum_{N}^{i=1} \beta_i = N_d\ and\ \beta_i \in \{0, 1\}\ and\ \alpha \leq g_i \leq \gamma \tag{11}$$

where $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_N)$ and $\boldsymbol{g} = (g_1, g_2, \ldots, g_{N_d})$ are the parameters that should be optimized. Similar to Task 2, in Task 3 also the feedback gains are limited to [0, 100].

Optimizing link weights for better pinning control (Task 4). In all the above problems, the underlying connection graph is an undirected and unweighted network. However, many real-world networks are weighted, and it has been shown that synchronization properties of dynamical networks can be improved by proper weighting the links[21,22]. Here we investigated whether the link weights can also be optimized for better control. To this end, we used the networks obtained in Task 1 (with optimized locations of drivers), and optimized the connection weights using the optimization technique. Let us denote the weight of the edge between nodes i and j by w_{ij}. The optimization problem can be formulated as
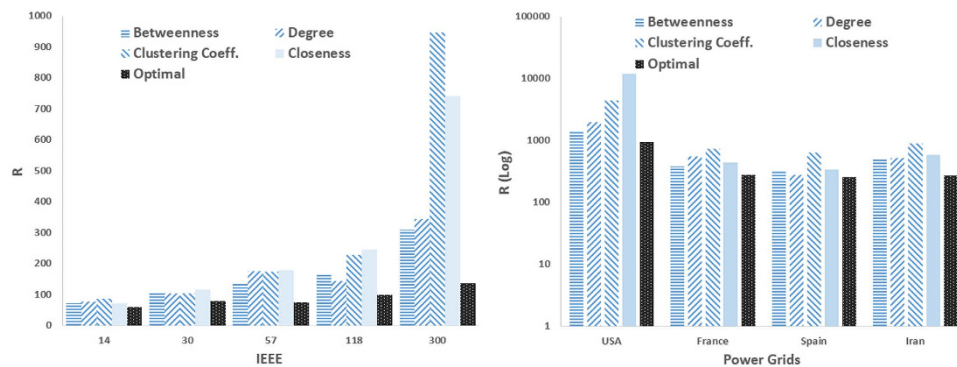
Figure 1. Optimal controllability (the less R the better the controllability of the network) in (left) IEEE benchmark and (right) and real power grid networks. The outcome of the optimization process (Optimal) is compared to the cases where the driver nodes are chosen based on (high) centrality values in terms of degree, betweenness, closeness, and clustering coefficient.
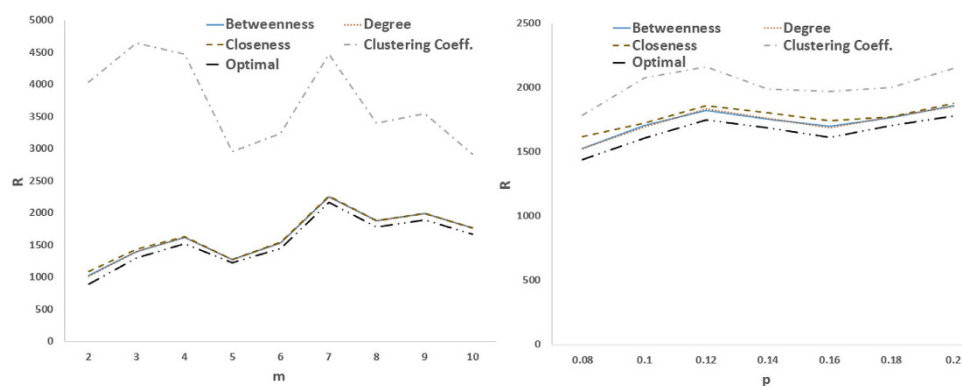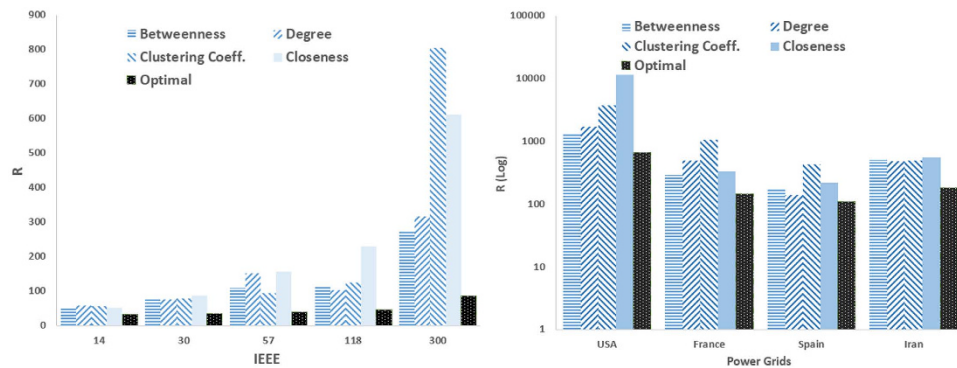

Figure 2. Optimal controllability as compared to heuristic methods; (left) R as a function of *m* in preferential attachment scale-free networks, and (right) R as a function of rewiring probability *P* in small-world networks. The graphs show averages over 20 realizations.
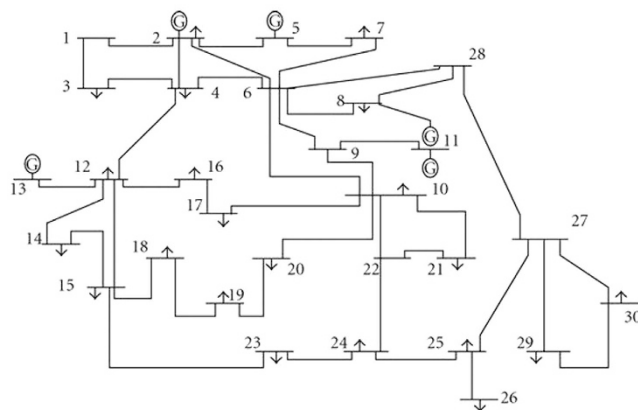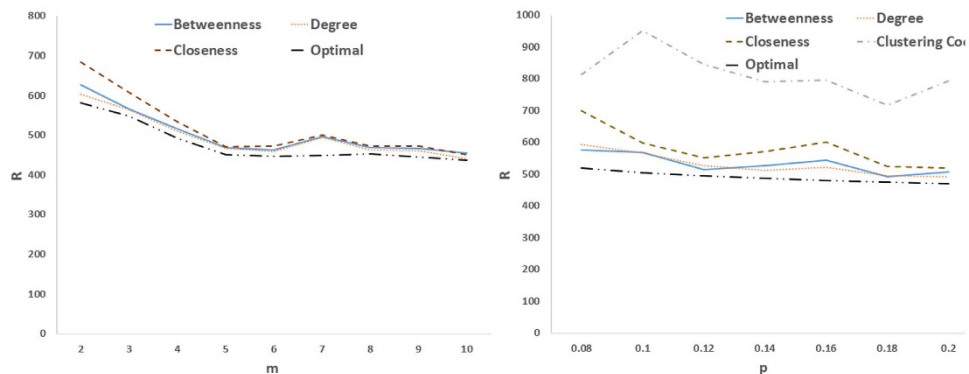

Figure 3. Schematic view of 30-node IEEE benchmark network.

$$\min_{w_{ij}} R(\boldsymbol{w}) = \frac{\lambda_N(\boldsymbol{w})}{\lambda_1(\boldsymbol{w})}$$

$$subject\ to\ \zeta \leq w_{ij} \leq \eta \tag{12}$$

where $\boldsymbol{w}$ is a vector of edges' weights that should be optimized. For this task, all drivers have the same feedback gain of $g_i = 10$.

Figure 4. R in benchmark and real power networks when the feedback gains are optimized. The location of drivers is those of Fig. 1.
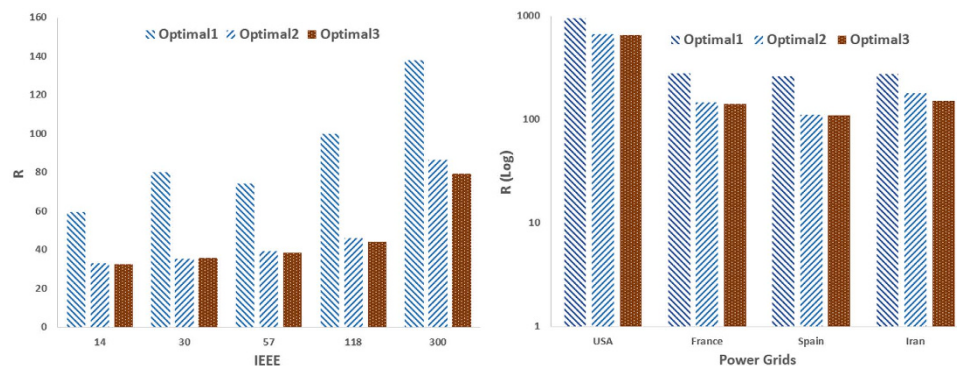

Figure 5. (Left) R as a function of *m* in preferential attachment scale-free networks, and (right) R as a function of rewiring probability *P* in small-world networks. The feedback gains are optimized through the optimization process and the location of drivers is those of Fig. 2. The graphs show averages over 20 realizations.

Optimization technique. The above optimization problems are not convex, and thus efficient tools available for optimization of convex problems cannot be used for them. A good approach to handle such optimization tasks is to use population-based techniques. In this paper, we used Cat Swarm Optimization (CSO), which has been shown to be an efficient population-based optimization technique[23]. CSO is an evolutionary algorithm designed to solve problems with discrete or continuous search space. This algorithm is composed of two modes: Seeking and Tracing. For modeling the behavior of cats in both the resting and alert states, we use the seeking mode. This mode is a time for thinking and deciding about the next move, and has four main parameters: seeking memory pool (SMP), seeking range of the selected dimension (SRD), counts of dimension to change (CDC) and self-position consideration (SPC). Seeking mode first makes copies of each cat (as counts of SMP by considering SPC), and then selects some copies and randomly mutates position of each cat (i.e., it selects CDC dimensions of each cat, and then mutates by $(1 + SRD)$ or $(1 - SRD)$ randomly)[23]. Tracing mode is responsible for moving cats to new points based on position and velocity equations considering the position of the best-performing cat. In order to integrate the two modes into the algorithm, we define a mixture ratio (MR) which indicates the rate of mixing of seeking and tracing modes. This parameter decides how many cats will be moved into the seeking mode process (the remaining cats will be moved into the tracing mode). In order to solve the first problem (i.e., finding the location of optimal driver nodes), discrete version of CSO[24] was used, while continuous version was used to solve the second problem (i.e., finding the optimal feedback weights while the location of drivers is fixed). In order to simultaneously optimize the location of optimal drivers and their associated feedback gain, co-evolutionary technique that was introduced in[25] was used to develop an appropriate optimization method. It is worth mentioning that other population-based optimization techniques (such as genetic algorithms, particle swarm optimization, simulated annealing and differential evolution) can also be used to solve these optimization problems; however our experiments showed that CSO resulted in better performance compared to them.

Results and Discussion
Datasets.
The above optimization techniques are applied to both model and real networks. Pining control has potential applications in power grids in which there are generator and consumer nodes. Therefore, we used a number of benchmark and real power networks in this work including IEEE benchmark networks with various nodes, high voltage grid of Spain, France[26] and Iran[27] and transmission network of northern USA[28]. In these
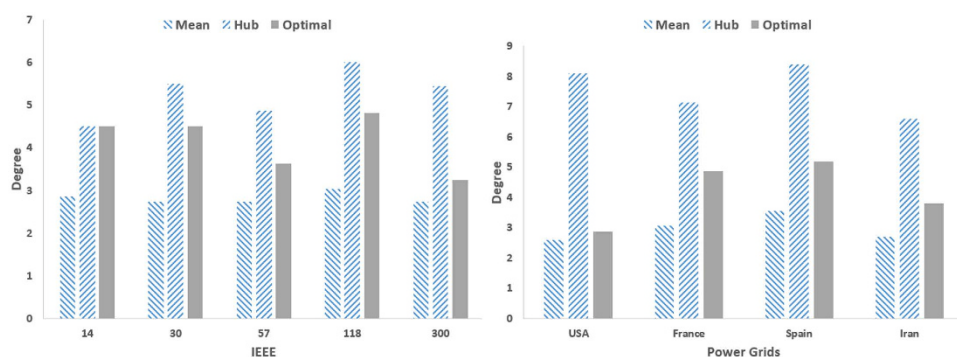
Figure 6. R in benchmark and real power networks when the locations and feedback gains are simultaneously optimized. The figures show the cases when only the location of drivers is optimized (Optimal 1), when only the feedback gains are optimized provided that the optimal drivers are already found (Optimal 2), and when the location of drivers and feedback gains are simultaneously optimized through a 2-dimesional optimization process (Optimal 3).
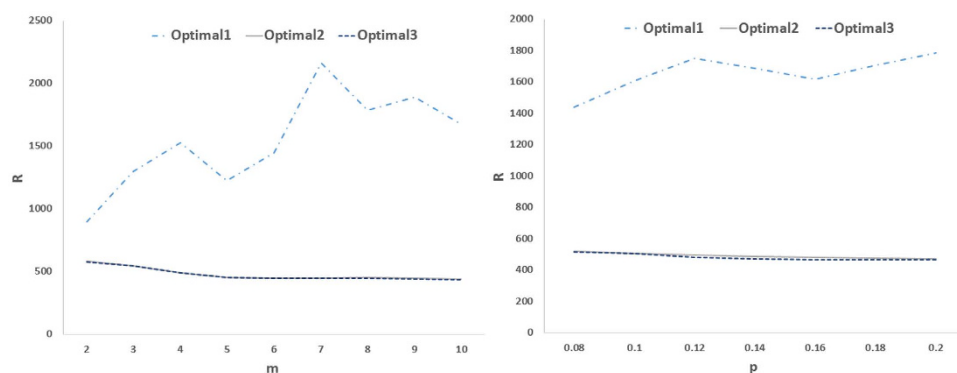

Figure 7. (Left) R as a function of *m* in preferential attachment scale-free networks, and (right) R as a function of rewiring probability *P* in small-world networks. Other designations are as Fig. 5 and graphs show averages over 20 realizations.

Figure 8. Average degree of the network (Mean), that of hub nodes (Hub), and that of optimal drivers (Optimal) in IEEE benchmark and real power networks.

networks, a node can be a generator, transformer or substation and an edge represents a power supply line. Table 1 shows details of these networks. In IEEE benchmark networks, 14% of the nodes were considered to be drivers, while in other power grids, 5% of nodes were selected as drivers.

The experiments were also carried out on model networks. As model networks, we used preferential attachment scale-free networks, as proposed by Barabasi and Albert[29], and small-world networks, as proposed by Watts and Strogatz[28]. Scale-free networks were constructed using the original preferential attachment algorithm
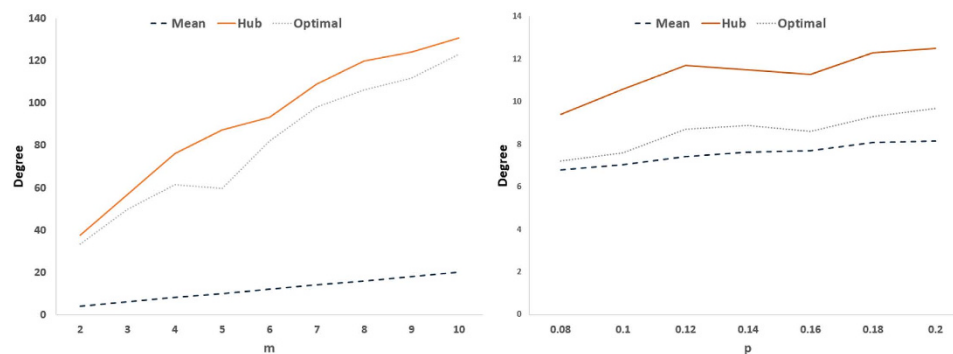
Figure 9. (Left) Average degree of the network (Mean), that of hub nodes (Hub), and that of optimal drivers (Optimal) in scale-free networks, and (right) average degrees as a function of *P* in small-world networks.

| Network	Common Nodes between tasks 1 and 3	Mean feedback gains difference between tasks 2 and 3	Network	Common Nodes between tasks 1 and 3	Mean feedback gains difference between tasks 2 and 3
BA ($m = 2$)	90%	2.3%	WS ($P = 0.08$)	60%	1.1%
BA ($m = 3$)	100%	0.6%	WS ($P = 0.1$)	80%	0.9
BA ($m = 4$)	90%	2.9%	WS ($P = 0.12$)	70%	1.4%
BA ($m = 5$)	100%	2.8%	WS ($P = 0.14$)	70%	1.6%
BA ($m = 6$)	100%	1.6%	WS ($P = 0.16$)	75%	0.7%
BA ($m = 7$)	100%	0.9%	WS ($P = 0.18$)	75%	2.4%
BA ($m = 8$)	90%	1%	WS ($P = 0.2$)	80%	1.2%
BA ($m = 9$)	90%	1.5%			
BA ($m = 10$)	90%	0.3%			

Table 3. The percentage of common optimal driver nodes in the outcome of Task 1 (when only the location of drivers is optimized and Task 3 (when both location of drivers and feedback gains are simultaneously optimized), and the absolute difference (in percentage) between the feedback gains in Task 2 (when the feedback gains are optimized for the drivers found in Task 1) and Task 3. The networks are Barabasi-Albert (BA) scale-free with different m and Watts-Strogatz (WS) small-world with different rewiring probability P.

introduced in[29], which is as follows. First a small set of fully connected network is considered (here a network with 3 nodes). Then, at each step, a new node is added to the network and connected to the old nodes with a probability that is proportional to their degree, i.e., the higher the degree of a node, the higher the chance of being connected to new nodes. Each new node makes *m* connections. Networks constructed in this way will have heterogeneous degree distribution; however, not all real-world networks show such properties. In order to construct networks with homogeneous properties, we considered small-world networks and used the algorithm proposed in[28]. First, a ring graph with each node connected to its *m*-nearest neighbor is considered. Then, each edge is rewired with probability *P*. For a range of *P*, the resulting network will have short average path length, comparable to corresponding random networks, while its clustering coefficient is still high (similar to the original ring graph), that is much higher than corresponding random networks. We chose 5% of the nodes as drivers and fixed the network size as 1000 for first three tasks and 200 for the fourth one. The experiments were carried out for different values of *m* in scale-free networks and different values of rewiring probability *P* in small-world networks (with $m = 6$).

Implementation details and parameters of the optimization method. Table 2 shows the values of the parameters of CSO method. These parameters are selected based on the recommendation from previous works[30]. The parameter "Max-Iteration" indicates the maximum number of iteration in our implementation. The optimization procedure is stopped after 500 iterations if a steady-state solution is not yet obtained. However, our experience showed that on average the optimization was terminated after 250 iterations. In order to avoid the results from random effects, we perform our experiments 20 times.

Simulation Results. Figures 1 and 2 compare the proposed optimization method with heuristic methods (i.e., selecting the driver nodes based on their centrality scores including degree, betweenness closeness and clustering coefficient) when only the location of drivers is optimized. Figure 1 shows the result of this comparison for real networks i.e., IEEE benchmarks and real power grids. As can be seen, the optimization strategy outperforms heuristic methods by providing the best controllability, i.e., the least *R*. Among the heuristic methods, those based on closeness and clustering coefficient have the poorest performance, while degree- and
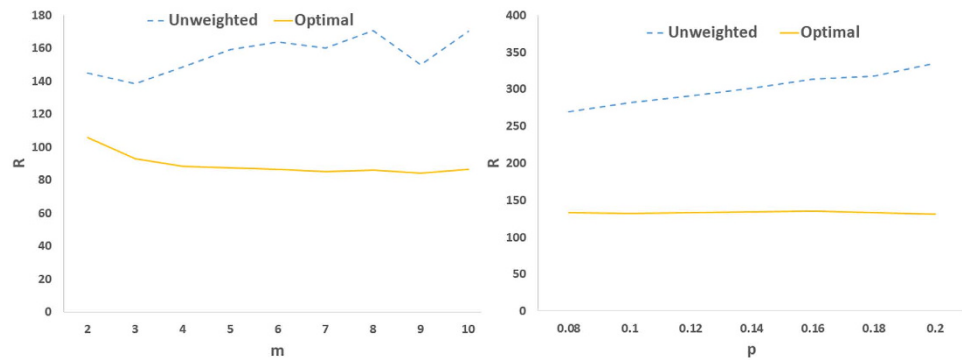
Figure 10. (Left) R as a function of _m_ in preferential attachment scale-free networks, and (right) R as a function of rewiring probability _P_ in small-world networks. The weights of existing edges are optimized through the optimization process. The graphs show averages over 20 realizations.
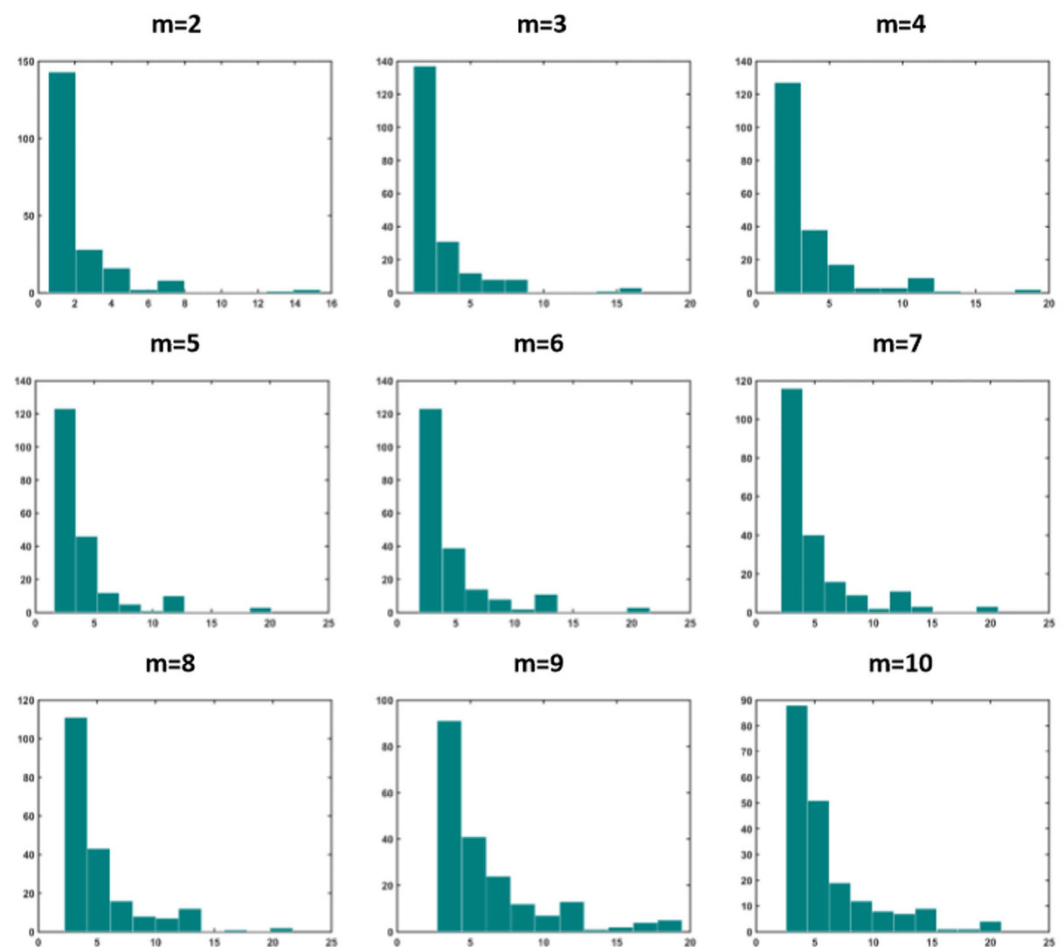
Figure 11. Distribution of nodal strength values in Barabasi-Albert for different values of parameter _m_. The graphs show averages over 20 realizations.

betweenness-based methods are the second-top performer (after the optimal one) interchanging. The optimal one is again the best-performer in model scale-free and small-world networks (Fig. 2), while the one based on clustering coefficient has the worst performance.

In order to better discuss the results, we consider the 30-node IEEE benchmark network as a sample network (Fig. 3), and study properties of the optimal driver nodes. In this network, we optimize the locations and feedback gains of 4 driver nodes. In this network, the average degree of nodes is 2.7 and that of 4 hub nodes is 5.5. The proposed optimization method (Task 1) selects nodes 2, 10, 15, 27 (average degree of 4.5) as optimal drivers. Note
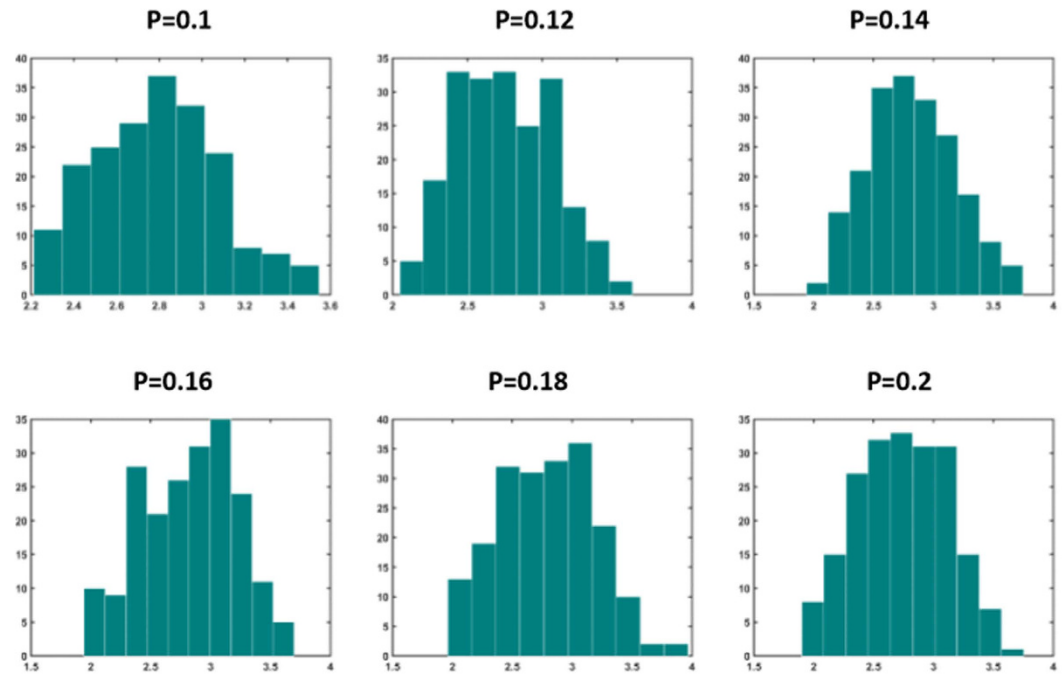
Figure 12. Distribution of nodal strength values in Watts-Strogatz networks for different values of rewiring probability *P*. The graphs show averages over 20 realizations.

that only nodes 2 and 10 are among the top-4 hub nodes. It is seen that sometimes taking a node with smaller degree than a hub node results in better controllability; for example, in our example node 27 with degree of 4 is better driver node than node 6 with degree 6.

In task 2, we aimed at optimizing the feedback gains for already existing driver nodes with fixed locations, and the results are shown in Figs 4 and 5 for real and model networks, respectively. It is seen that optimizing the feedback gains have significant influence on the controllability of the networks. Comparing Figs 1 and 4 reveals that optimizing the feedback gains made almost 30% improvement in the pinning controllability of both IEEE benchmarks and real power grid networks. Considering our example network (30-node IEEE benchmark network), the optimal drivers are nodes 2, 10, 15 and 27 with optimized feedback gains as 5.93, 4.06, 6.49 and 6.38, respectively. The improvement of pinning controllability in model networks is much better; by optimizing the feedback gains, the eigenratio of scale-free and small-world networks decreased almost 60% and 70%, respectively.

Next we consider optimizing the eigenratio as a 2-dimesional optimization problem in which both the location of drivers and feedback gains are simultaneously optimized. This allows search the solution space deeper and finds better results. Figures 6 and 7 show the results for real and model networks, respectively. It is seen that optimizing the location of drivers and feedback gains at the same time, can do the job better as compared to other cases that leads to network structures with better controllability. Another observation about the optimal controllability of model networks is that as the average degree of scale-free networks increased (i.e., *m* increased), the optimal controllability was improved. In small-world networks, as the rewiring probability increased (i.e., more shortcuts were created), communicability among the nodes improved, and as a result, the optimal controllability was enhanced.

In order to further study details of optimal driver nodes, we compare their average degree and feedback gain with those of hub nodes as well as the whole network. For each case, only the top-N_d hub nodes are considered. Figures 8 and 9 compare the average degrees for real and model networks, respectively. The real networks do not have the same behavior. For example, in 14-node IEEE network, the optimal drivers are indeed the hub nodes. As the networks become larger, the average degree of optimal drivers gets closer to the average degree of the network, indicating that more hubs are missing in the set of drivers. In the largest real network (USA grid), the optimal drivers have average degree much closer to the network average than the hub nodes. Therefore, one cannot simply search for optimal drivers among the hub, and sophisticated method (like the optimization strategy proposed in this work) should be used. Model networks have also different behavior. While, the average degree of optimal drivers is close to that of hubs in scale-free networks, it is much closer to the network average in Watts-Strogatz networks. Indeed, as the network becomes more homogeneous, less hubs are selected in the optimal drivers set.

We compare the outcomes of Tasks 1, 2 and 3. Table 3 summarizes the comparisons between the outcomes of these tasks in the considered Barabasi-Albert and Watts-Strogatz networks with varying parameters. The table shows the percentage of common nodes between the outcomes of tasks 1 and 3, and also the normalized difference between the feedback gains obtained in tasks 2 and 3. As is seen, tasks 1 and 3 result in almost the same set of optimal drivers in scale-free networks for which the match is at least 90%; however it varies between 60% to 80% in small-world networks. This is mainly due to higher heterogeneity in scale-free networks. As shown in[16], the set of optimal drivers in scale-free networks has a mean degree close to that of hub nodes (much higher than the
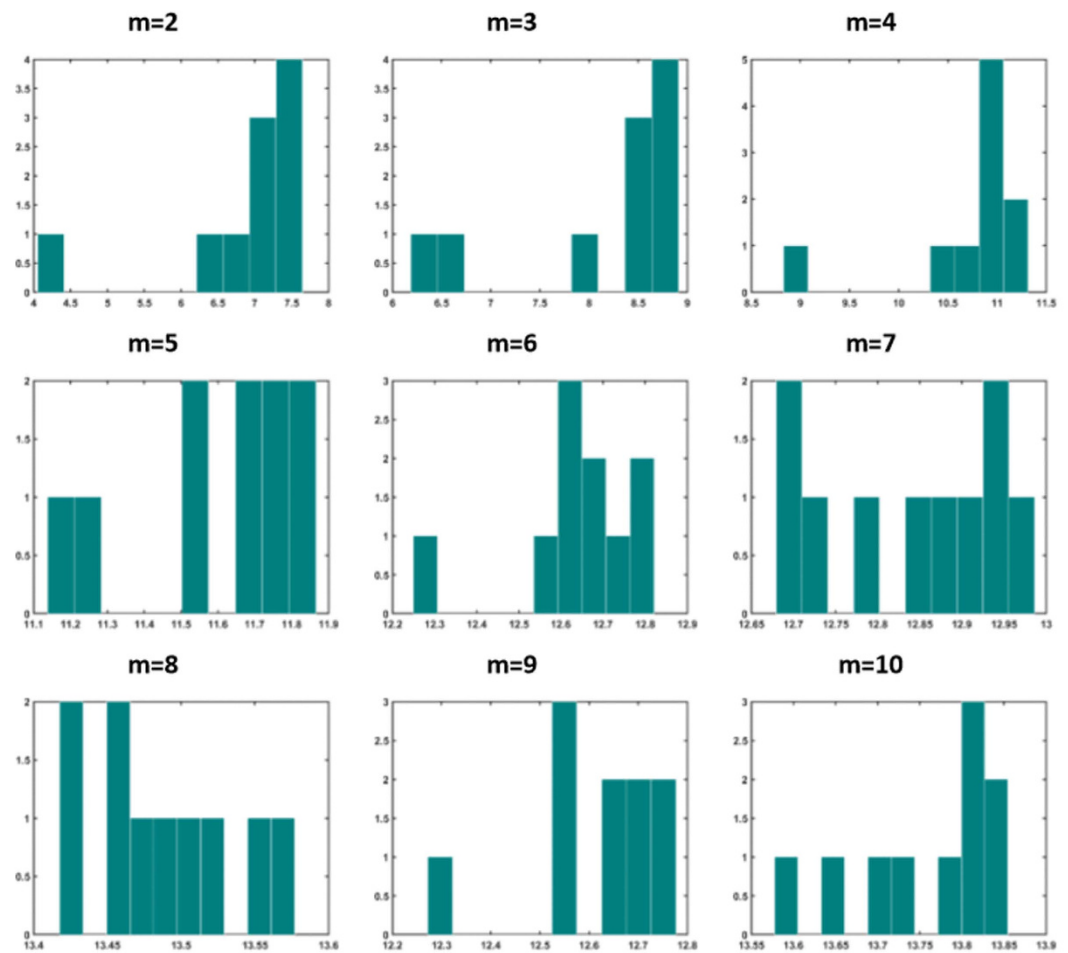
Figure 13. Distribution of strength values of driver nodes in Barabasi-Albert for different values of parameter *m*. The graphs show averages over 20 realizations.
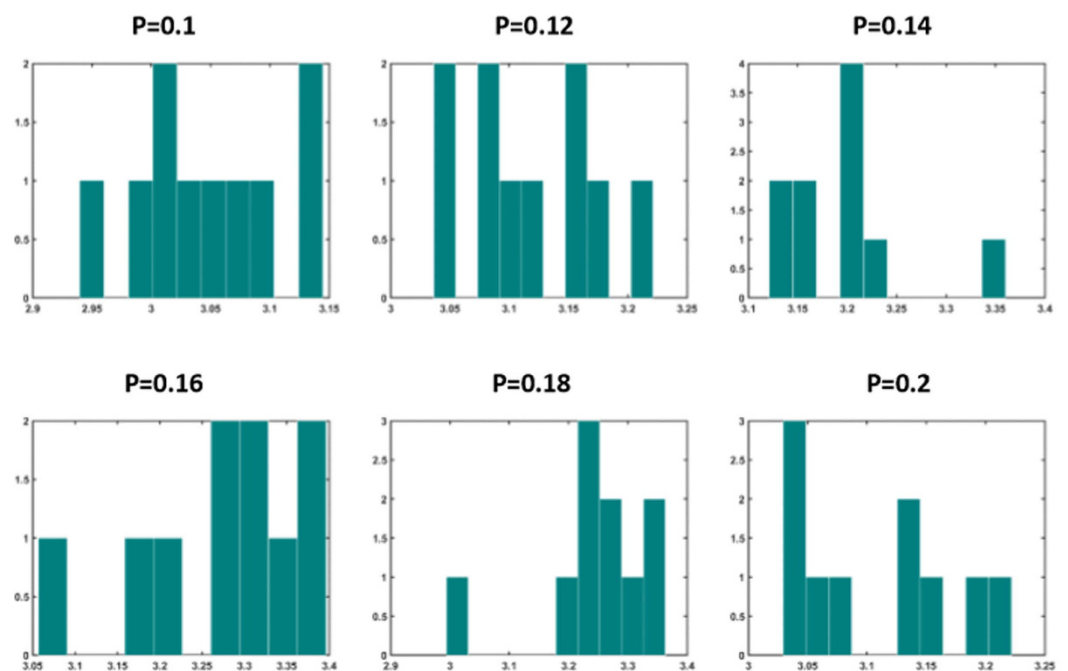
Figure 14. Distribution of strength values of driver nodes in Watts-Strogatz networks for different values of rewiring probability *P*. The graphs show averages over 20 realizations.

Network	Percentage of hubs that have lower strength than leaves	Network	Percentage of link weights of hubs lower than average link weights of leaves
BA ($m = 2$)	54%	WS ($P = 0.1$)	56%
BA ($m = 3$)	51%	WS ($P = 0.12$)	62%
BA ($m = 4$)	58%	WS ($P = 0.14$)	50%
BA ($m = 5$)	59%	WS ($P = 0.16$)	55%
BA ($m = 6$)	65%	WS ($P = 0.18$)	51%
BA ($m = 7$)	55%	WS ($P = 0.2$)	51%
BA ($m = 8$)	60%		
BA ($m = 9$)	53%		
BA ($m = 10$)	58%		

Table 4. The percentage of hubs that have lower average link weight than leaves. The networks are Barabasi-Albert scale-free with different *m* and Watts-Strogatz small-world with different rewiring probability *P*.

mean degree of the whole network), while it is close to the network mean degree in small-world networks. Indeed, in scale-free networks the optimization algorithm always searches for optimal drivers within a small subset of the nodes (that are often those with high degrees), and thus varying or fixing the feedback gains has little effect on the final optimal drivers set. However, in homogeneous networks such as Watts-Strogatz small-world networks, the search space is much wider and the set of candidate nodes for optimal drivers is much larger than scale-free ones. Therefore, by optimizing the feedback gains, one will have higher degree of freedom and the set of optimal drivers may substantially change. When comparing the optimal feedback gains found in task 2 (fixed optimal drivers) and task 3 (drivers are simultaneously optimized), there is small difference (maximum 2.9%) in both scale-free and small-world networks. This indicates that optimization of feedback gain does not significantly differ whether the optimal drivers are separately fixed or simultaneously optimized with the gains.

Task 4 is to optimize the connections weights. Figure 10 shows the influence of optimal link weighting on the controllability of model scale-free and small-world networks. Having the same feedback gain for the driver nodes, first, the locations of best drivers were found through the optimization technique, and then, the link weights were optimized. This figure compares the original unweighted network with the one with optimized weights; the optimal weighting strategy could significantly improve the controllability (more than 50% in many cases).

Next we study the distribution of nodes' strength in optimally weighted networks (strength of a node is the sum of connection weights pointing to that node). It is well-known that unweighted Barabasi-Albert networks have scale-free degree distribution with exponent 3, while Watts-Strogatz networks have Poissonian degree distribution. Figures 11 and 12 show the distribution of nodal strength in optimally weighted Barabasi-Albert and Watts-Strogatz networks, respectively. These distributions are similar to the degree distribution of original networks (not shown here), but with different parameters. This indicates that the optimization strategy used in task 4 did not dramatically change the distribution of the strength values (which equals to degree in unweighted networks). However, this is not the case when the distribution is obtained for drivers (Figs 13 and 14 for Barabasi-Albert and Watts-Strogatz networks, respectively). The drivers in the optimally weighted networks have almost a flat distribution of strength values, indicating that the optimization strategy tries to make a balance between the weights connected to the drivers such that they have strength values in the same range. Indeed, the optimization process homogenizes the properties of driver nodes. In order to have deeper understanding on the distribution of optimal weights in the network, we compare the weights in hubs and leaves. To this end, the nodes are classified into two groups of hubs (high degrees) and leaves (low degrees). Table 4 shows the percentage of the hubs that have lower average link weight than the leaves. As it sees, in all cases, more than half of the hubs have lower average link weights than the leaves. This means that the optimization strategy assigns in average more weights on the links pointing to the leaves.

Conclusions

Controlling complex network systems from an initial state to a final desired state has many applications in science and engineering. Pinning control of dynamical networks is a type of control at which the dynamics of individual nodes is pinned to a single reference trajectory. In this work, we considered a number of problems in optimal pinning controllability and used a population-based optimization technique to solve them. Using linear stability analysis and the master stability function approach, we considered a metric quantifying the controllability of the network. The metric is the largest eigenvalue of the augmented Laplacian matrix divided by its smallest eigenvalue. We then formulated four problems: *i*) optimizing the locations of driver nodes with fixed feedback gains, *ii*) optimizing the feedback gains with fixed locations for the drivers, *iii*) simultaneously optimizing the locations of drivers and their associated feedback gains, and *iv*) optimizing the connection weights. In order to solve the optimization problem, we used proper versions of cat swarm optimization technique, which is a new population-based optimization technique developed to solve non-convex optimization problems. Our extensive simulation results on both real and model networks showed that the proposed optimization method could successfully optimize the pinning controllability under different conditions. The proposed method showed superior performance as compared to a number of heuristic methods. The outcome of this research can be used where an optimal structure is required for pinning control.

References

1. Rad, A. A., Khadivi, A. & Hasler, M. Information processing in complex networks. *IEEE Circuits and Syst. Mag.* **10,** 26–37 (2010).
2. Barabási, A.-L. Network science. *Philosophical Trans. of the Royal Society of London A: Math., Phys. and Eng. Sci.* **371,** 20120375 (2013).
3. Estrada, E. Journal of Complex Networks: Quo Vadis? *Journal of Complex Networks* **1,** 1–2 (2013).
4. Pikovsky, A., Rosenblum, M. & Kurths, J. *Synchronization: A Universal Concept In Nonlinear Sciences* (Cambridge university press, 2003).
5. Arenas, A., Díaz-Guilera, A., Kurths, J., Moreno, Y. & Zhou, C. Synchronization in complex networks. *Phys. Rep.* **469,** 93–153 (2008).
6. Jalili, M. A graph weighting method for reducing consensus time in random geographical networks. *Adv. Info. Networking and App. Workshops* 317–322, doi: 10.1109/WAINA.2010.55 (2010).
7. Liu, Y.-Y., Slotine, J.-J. & Barabási, A.-L. Controllability of complex networks. *Nature* **473,** 167–173 (2011).
8. Nepusz, T. & Vicsek, T. Controlling edge dynamics in complex networks. *Nat. Phys.* **8,** 568–573 (2012).
9. Barzel, B. & Barabási, A.-L. Universality in network dynamics. *Nat. Phys.* **9,** 673–681 (2013).
10. Cornelius, S. P., Kath, W. L. & Motter, A. E. Realistic control of network dynamics. *Nat. Comm.* **4,** 1942 (2013).
11. Ruths, J. & Ruths, D. Control profiles of complex networks. *Science* **343,** 1373–1376 (2014).
12. Sorrentino, F., di Bernardo, M., Garofalo, F. & Chen, G. Controllability of complex networks via pinning. *Phys. Rev. E* **75,** 046103 (2007).
13. Porfiri, M. & Di Bernardo, M. Criteria for global pinning-controllability of complex networks. *Automat* **44,** 3100–3106 (2008).
14. Zhou, J., Lu, J.-A. & Lü, J. Pinning adaptive synchronization of a general complex dynamical network. *Automat* **44,** 996–1003 (2008).
15. Pecora, L. M. & Carroll, T. L. Master stability functions for synchronized coupled systems. *Phys. Rev. Lett.* **80,** 2109 (1998).
16. Jalili, M., Sichani, O. A. & Yu, X. Optimal pinning controllability of complex networks: Dependence on network structure. *Phys. Rev. E* **91,** 012803 (2015).
17. Zhou, M. Y., Zhou, Z., Liao, H., Fu, Z. Q. & Cai, S. M. Enhancing speed of pinning synchronizability: low-degree nodes with high feedback gains. *Sci. Rep.* **5,** 17459, doi: 10.1038/srep17459 (2015).
18. Grigoriev, R., Cross, M. & Schuster, H. Pinning control of spatiotemporal chaos. *Phys. Rev. Lett.* **79,** 2795 (1997).
19. Wang, X. F. & Chen, G. Pinning control of scale-free dynamical networks. *Physica A* **310,** 521–531 (2002).
20. Chung, F. R. *Spectral graph theory* (American Mathematical Soc., 1997).
21. Jalili, M., Rad, A. A. & Hasler, M. Enhancing synchronizability of weighted dynamical networks using betweenness centrality. *Phys. Rev. E* **78,** 016105 (2008).
22. Jalili, M. Enhancing synchronizability of diffusively coupled dynamical networks: a survey. *IEEE Trans. Neural Net. and Learning Syst.* **24,** 1009–1022 (2013).
23. Chu, S.-C., Tsai, P.-W. & Pan, J.-S. Cat swarm optimization. *PRICAI 2006: Trends in Artificial Intelligence* 854–858, doi: 10.1007/978-3-540-36668-3_94 (2006).
24. Sharafi, Y., Khanesar, M. A. & Teshnehlab, M. Discrete binary cat swarm optimization algorithm. *3rd International Conference on Computer, Control & Communications* 1–6, doi: 10.1109/IC4.2013.6653754 (2013).
25. Wiegand, R. P., Liles, W. C. & De Jong, K. A. An empirical analysis of collaboration methods in cooperative coevolutionary algorithms. *GECCO* **2611,** 1235–1245 (2001).
26. Rosato, V., Bologna, S. & Tiriticco, F. Topological properties of high-voltage electrical transmission networks. *Electric Power Syst. Research* **77,** 99–105 (2007).
27. Monfared, M. A. S., Jalili, M. & Alipour, Z. Topology and vulnerability of the Iranian power grid. *Physica A* **406,** 24–33 (2014).
28. Watts, D. J. & Strogatz, S. H. Collective dynamics of 'small-world' networks. *Nature* **393,** 440–442 (1998).
29. Barabási, A.-L. & Albert, R. Emergence of scaling in random networks. *Science* **286,** 509–512 (1999).
30. Orouskhani, M., Orouskhani, Y., Mansouri, M. & Teshnehlab, M. A novel cat swarm optimization algorithm for unconstrained optimization problems. *IJITCS* **5,** 32–41 (2013).

Acknowledgements

Author Contributions

M.J. designed the research and managed the project. Y.O. performed the numerical simulations. Y.O., M.J. and X.Y. analysed the results and wrote the manuscript.

Additional Information

Competing financial interests: The authors declare no competing financial interests.

How to cite this article: Orouskhani, Y. *et al.* Optimizing Dynamical Network Structure for Pinning Control. *Sci. Rep.* **6**, 24252; doi: 10.1038/srep24252 (2016).