On Controllability of Neuronal Networks with Constraints on the Average of Control Gains

Yang Tang, *Member, IEEE*, Zidong Wang, *Senior Member, IEEE*, Huijun Gao, *Senior Member, IEEE*, and Jürgen Kurths

***Abstract*—Control gains play an important role in the control of a natural or technical system since they reflect how much resource is required to optimize a certain control objective. This paper is concerned with the controllability of neuronal networks with constraints on the average value of the control gains injected in driver nodes, which are in accordance with engineering and biological backgrounds. In order to deal with constraints on control gains, the controllability problem is transformed into a constrained optimization problem (COP). The introduction of the constraints on the control gains unavoidably leads to substantial difficulty in finding feasible solutions as well as refining solutions. As such, a modified dynamic hybrid framework (MDyHF) is developed to solve this COP, based on an adaptive differential evolution and the concept of Pareto dominance. By comparing with statistical methods and several recently reported constrained optimization evolutionary algorithms (COEAs), we show that our proposed MDyHF is competitive and promising in studying the controllability of neuronal networks. Based on the MDyHF, we proceed to show the controlling regions under different levels of constraints. It is revealed that we should allocate the control gains economically when strong constraints are considered. In addition, it is found that, as the constraint becomes more restrictive, the driver nodes are more likely to be selected from the nodes with a large degree. The results and methods presented in this paper will provide insightful lights on developing new techniques to control a realistic complex network efficiently.**

***Index Terms*—synchronization, pinning control, neuronal networks, evolutionary algorithms.**

I. Introduction

The past one decade has witnessed a tremendous upsurge in the research interest toward theoretical modeling, analysis and application of complex networks from a variety of research communities [1]. The main reason lies in the fact that complex networks can describe many practical systems such as genetic networks, social networks, sensor networks, neuronal networks, electronic networks or transportation networks. Among them, the modeling of neuronal networks of a brain can be viewed as a typical application of complex networks [2], [3]. In [4]–[6], the theoretical modeling, tackling learning of neuronal networks and the applications of neuronal networks to image processing have been investigated, respectively. Modern brain mapping approaches such as diffusion MRI, functional MRI, EEG, and MEG have constantly produced large datasets of anatomical and functional connection patterns. Complex network theory has been used to describe important properties of large connection datasets by quantifying structures of their respective network representations. It has been widely recognized that network characterization of structural and functional connectivity data of brain has attracted increasing attention due to its reliability and effectiveness [2]. Recently, the existence has been revealed with respect to the communities, hierarchy, centrality and distribution of cortical hubs in anatomical connectivity of the mammalian brain [3], [7]–[10].

Complex networks, especially neuronal networks, have been investigated in the context of dynamical systems and have already become an interdisciplinary research area for mathematicians, computer scientists and biologists to interpret functional information and explore network robustness and vulnerability, which are likely to become increasingly relevant in relation to neuroscience, physics and engineering. Recently, as an emerging phenomenon of neuroscience and multi-agent systems, synchronization has gained particular research attention for complex networks (neuronal networks) in various fields [11]–[17]. Synchronization of distributed brain activity has been revealed to serve a central role in high-level neural information processing [18]. Experimentally observed evidence has asserted that certain brain disorders, such as schizophrenia, epilepsy, autism, Alzheimer's disease, and Parkinson's disease, are highly relevant to abnormal neural synchronization [19]. In [7], [20], synchronization in the cortical brain network of the cat is investigated by modeling each node (cortical area) with a subnetwork of interacting excitable neurons. In [21], the distributed synchronization problem is investigated for networks of agent systems with nonlinearities and controller failure subject to Bernoulli switchings and conditions are given in terms of a semi-definite programming problem.

On another research frontier, controllability of complex networks has received considerable attention in the past ten years. Controllability of complex networks can be referred to that a set of nodes are regarded as driver nodes/references and used to control the dynamics of entire networks to a desired state, which is required for an engineering, medical or biological purpose [22]–[28]. In particular, as illustrated in

Yang Tang is with the Research Institute of Intelligent Control and Systems, Harbin Institute of Technology, Harbin 150080, China and the Institute of Physics, Humboldt University, Berlin 12489, Germany and the Potsdam Institute for Climate Impact Research, Potsdam 14415, Germany. e-mail: tangtany@gmail.com.

Zidong Wang is with the Department of Information Systems and Computing, Brunel University, Uxbridge, Middlesex, UB8 3PH, United Kingdom. E-mail: Zidong.Wang@brunel.ac.uk.

Huijun Gao is with the Research Institute of Intelligent Control and Systems, Harbin Institute of Technology, Harbin 150080, China. e-mail: hjgao@hit.edu.cn.

Jürgen Kurths is with the Potsdam Institute for Climate Impact Research, Potsdam, Germany and the Institute of Physics, Humboldt University, Berlin, Germany and the Institute for Complex Systems and Mathematical Biology, University of Aberdeen, Aberdeen AB24 3UE, United Kingdom. e-mail: Juergen.Kurths@pik-potsdam.de.

[28], the importance of investigating controllability of neuronal networks will not only help us to elucidate how to control an intricate system efficiently, but also be beneficial to understand the processing of high level information in brains [7] and dynamical properties of neuronal networks [10]. Recently, a variety of works have been proposed to realize pinning control or detection of controlling regions in complex networks. In [29], pinning state feedback controllers have been designed to synchronize a state coupled dynamical networks. In addition to the mathematical methods to study the controllability of complex networks, some efforts on choosing key nodes by utilizing evolutionary methods have been made [27], [28], [30]. The problem of pinning control of complex networks has been converted into an unconstrained problem [27] and a constrained one [28], respectively. In particular, two measures of controllability of neuronal networks have been incorporated into one unified framework [28], where the more important measure is regarded as an objective and the other one is viewed as a constraint.

It should be noted that, up to now, almost all research efforts on controllability (pinning control) of complex networks have been devoted to the case of choosing effective nodes to control the entire network. However, in reality, constraints on control gains should be taken into account. The importance of such considerations resides in twofolds: 1) The first one is from the constraint on implementation of engineering equipment and biological background. Saturation in actuator exists widely in practical control systems as a physical actuator can only generate bounded signals, and the control of plants with actuator saturation is also challenging [31]; 2) The second one is that only suitable control input could result in an ideal control performance. For example, in therapy, the patient's recovery is closely related to the dosage of antibiotics, where the input of dosage can be viewed as control gains. The excessive injection of dosage of drugs will result in the creation of multidrug-resistant bacteria and finally no efficient antibiotics are available in some severe cases [32], [33]. Misuse of antibiotics can also destroy the beneficial bacteria and cause immune system disorders in human body. On the other hand, a small injection of dosage will not be conducive to patients' recovery and prolong the recovery time of patients. Therefore, a dosage should be injected at an appropriate level which would work on the infected cells and would not upset the normal mechanism.

For the sake of simplicity, in [25], [28], the controllability of complex networks and neuronal networks has been investigated, respectively, where the intrinsic constraint on the control gains on the dynamics of networks has been overlooked and only the boundary of control gains is discussed. Despite the fact that many phenomena in nature are closely related to the constraint on gains on controllability of complex networks, the gain constraint issue has, unfortunately, been been largely neglected in the area due primarily to the complexity in optimizing and tackling the existence of gain constraints. It is, therefore, the main purpose of this paper to investigate how much the controllability of weighted and directed neuronal networks is affected in the presence of constraint on control gains, which aims to improve our recent work [28], which does not consider the importance of gain constraints.

In this study, we aim to make the one of the first few attempts to address the controllability of neuronal networks with several constraints on control gains. Such a controllability issue is later converted into a constrained optimization problem (COP). Due to the nature of combinatorial optimization problems in selecting controlling nodes, constrained optimization evolutionary algorithms (COEAs) [34]–[36] are promising candidates to solve this COP. COEAs are composed of two major parts: a search technique and a constraint-handling scheme. Their performance rests largely on these two components. The constraint-handling scheme can be categorized into several classes [35]. In addition, it is important to develop an effective search algorithm to refine solutions that can find global optimal solutions for COPs. Recently, multi-objective optimization-based constraint handling schemes are used to tackle COPs, together with differential evolution (DE) due to its prospect and potential [37], [38]. Nonetheless, the search performance of these methods can still be further improved by introducing adaptive mechanisms in DEs.

In this paper, the controllability of neuronal networks with constraints on control gains is investigated. By adding an adaptive differential evolution (JaDE) [39] into a search scheme of a dynamic hybrid framework (DyHF), a modified dynamic hybrid framework (MDyHF) is proposed here to study controllability of neuronal networks. The main contributions of this paper are threefold: (1) the average value of control gains is considered as an constraint in controllability of neuronal networks and transformed into a COP; (2) based on an adaptive DE and Pareto dominance, a MDyHF is proposed to show its competitive performance by several experiments; (3) the controlling regions of the neuronal network are identified by the proposed MDyHF and the relationship between controllability and control gains is presented, which will interpret the mechanism of controlling natural systems.

The organization of this paper is arraged as follows. Section 2 presents some preliminaries and problem formulation briefly. The MDyHF is presented in Section 3. In Section 4, comparisons and results are provided. Conclusions are given in Section 5.

Notation: Throughout this paper, let a graph be $\mathcal{G} = [\mathcal{V}, \mathcal{E}]$, where $\mathcal{V} = \{1, \cdots, N\}$ denotes the vertex set and $\mathcal{E} = \{e(i,j)\}$ stands for the edge set. The graph $\mathcal{G}$ is directed, weighted and simple (without self-loops and multiple edges). Let $G = [g_{ij}]_{i,j=1}^{N}$ be the adjacency matrix of neuronal network of cats' brain $\mathcal{G}$, which is defined as follows: for any pair $i \neq j, g_{ij} < 0$ if $e(i,j) \in \mathcal{E}$; otherwise, $g_{ij} = 0$. $l \in [1, N]$ is the number of driver nodes of a network. $\phi_{\mathcal{P}}(\cdot)$ represents the characteristic function of the set $\mathcal{P}$, i.e., $\phi_{\mathcal{P}}(i) = 1$ if $i \in \mathcal{P}$; otherwise, $\phi_{\mathcal{P}}(i) = 0$. $g_{ii} = -\sum_{j=1, j\neq i}^{N} g_{ij}$ $(i = 1, 2, \cdots, N)$. The adjacency matrix G can be converted into the Laplacian matrix L by neglecting the weights over the network. For any pair $i \neq j, l_{ij} = -1$ if $e(i,j) \in \mathcal{E}$; otherwise, $l_{ij} = 0$. $l_{ii} = -\sum_{j=1, j\neq i}^{N} l_{ij}$, $(i = 1, 2, \cdots, N)$.

II. Problem formulation

In this section, some preliminaries and problem formulation are provided.

A. Controllability of the neuronal network

Hereafter, a desired state can be described as follows:

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

This differential equation is widely used to represent extensive real-world natural and technological systems [24].

In order to control the states of the neuronal network to the reference evolution $s(t)$, the dynamics of the neuronal network with output feedback controllers can be written as:

$$\begin{aligned}\frac{dx_i(t)}{dt} &= F(x_i,t) - C\sum_{j=1}^{N} g_{ij}H(x_j(t)) \\ &\quad - C\phi_{\mathcal{P}}(i)\kappa_i(H(s(t)) - H(x_i(t))), \\ &\quad i = 1,\cdots,N, \end{aligned} \tag{1}$$

where $x_i(t) = [x_{i1}(t), x_{i2}(t), \cdots, x_{id}(t)]^T \in \mathbb{R}^d$ $(i = 1, 2\cdots, N)$ is the state vector of the ith node/brain region/cortical area, $F(x_i,t) = [F_1(x_i,t), \cdots, F_d(x_i,t)]^T$ is a continuous vector function and $H(x_i(t))$ is the coupling continuous function. C is the global coupling gain of the neuronal network. Let $\mu_p = \mu_p^r + j\mu_p^m (j = \sqrt{-1}), (p = 1, 2, \cdots, N)$, be the set of eigenvalues of G and are sorted in such a way that $\mu_1^r \leq \mu_2^r \leq \cdots \leq \mu_N^r$. $\kappa_i, i \in \mathcal{P}$ is the control gain injected in driver nodes. Apparently, $1 \leq \sum_{i=1}^{N} \phi_{\mathcal{P}}(i) \leq N$. The objective of controllability is to regulate the states of the neuronal network (1) toward the desired reference state $s(t)$, i. e., $x_1(t) = x_2(t) = \cdots = x_N(t) = s(t)$.

For demonstration purpose, we use cortical network as an example. Here, $G = [g_{ij}]_{i,j=1}^{N}$ is the adjacency matrix of neuronal network of cats' brain, where nodes usually represent brain regions with coherent patterns of extrinsic anatomical or functional connections, while links stand for anatomical, functional, or effective connections [40], [41] and are differentiated on the basis of their weight and directionality. Here, the version of a dataset presented in [42] is used. The cerebral cortex of cats' brain can be separated into 53 cortical areas, linked by about 830 fibres of different densities into a weighted and directed complex network G. Abundance of evidence have shown that this network exhibits short average path-length, high clustering coefficient, the existence of hubs and hierarchically clusters, implying the information coordination for effective inter-area communication and for achieving high functional and structural complexity [7], [9], [43].

Following the way in [25], the extended network of $N+1$ dynamical systems y_i is considered, where $y_i = x_i$ for $i = 1, 2, \ldots, N$ and $y_{N+1} = s(t)$. Hence, (1) is:

$$\begin{aligned}&\frac{dy_i(t)}{dt} = F(y_i,t) - C\sum_{j=1}^{N+1} \mathcal{W}_{ij}H(y_j(t)), \\ &i = 1,\cdots,N+1, \end{aligned} \tag{2}$$

where $\mathcal{H} = [\mathcal{W}_{ij}] \in \mathbb{R}^{(N+1)\times(N+1)}$ in the form of

$$\mathcal{H} = \begin{pmatrix} \mathcal{D}_1 & g_{12} & \ldots & g_{1N} & -\phi_{\mathcal{P}}(1)\kappa_1 \\ g_{21} & \mathcal{D}_2 & \ldots & g_{2N} & -\phi_{\mathcal{P}}(2)\kappa_2 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ g_{N1} & g_{N2} & \ldots & \mathcal{D}_N & -\phi_{\mathcal{P}}(N)\kappa_N \\ 0 & 0 & \ldots & 0 & 0 \end{pmatrix}, \tag{3}$$

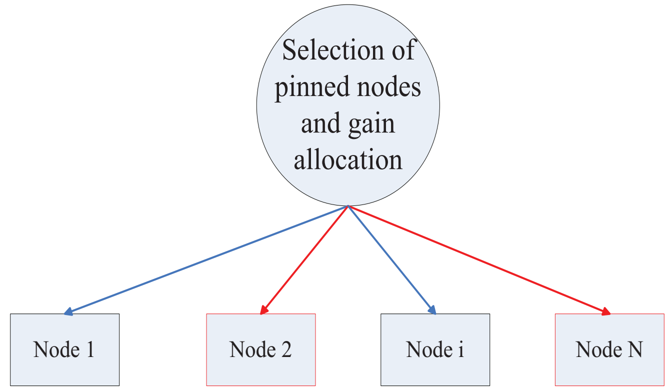

Fig. 1. Gain allocation of networks. Red rectangle and red line mean that the node is selected and gain should be allocated in $\mathcal{P}$; Dark rectangle and blue line mean that the node is not selected and gain will not be allocated.

in which $\mathcal{D}_i = g_{ii} + \phi_{\mathcal{P}}(i)\kappa_i$. Let $\lambda_p = \lambda_p^r + j\lambda_p^m$ be the pth eigenvalue of $\mathcal{H}$ and suppose that λ_p is sorted as $\lambda_1^r \leq \lambda_2^r \leq \cdots \leq \lambda_{N+1}^r$, where $\lambda_1^r = 0$.

The controllability can be measured in terms of

$$R = \frac{\lambda_{N+1}^r}{\lambda_2^r},$$

and

$$\delta = \max_p\{\lambda_p^m\}.$$

In order to enhance controllability of neuronal networks, we should minimize R and δ as much as possible [25], [44].

Remark 1. In previous works, the value of δ was usually ignored in measuring synchronizability and controllability of networks [45] since δ is small compared with R in most of coupling graphs. However, δ cannot be overlooked in some special cases such as normalized Laplacian matrix or the number of driver nodes l is large, where the value of δ is comparable to that of R. Hence, the assumption of neglecting δ will inevitably cause conservativeness and cannot reflect actual controllability of networks. In [28], we combine these two measures into a unified framework to investigate controllability of networks, in which R is viewed as an objective and δ is regarded as a constraint.

B. The incorporation of constraint on control gains in controllability of neuronal networks

As stated in [31], [46], the generation of signals is saturated in realistic networked control systems and artificial neural networks. Usually, in neural networks, the original network utilize multiple layers of weight-sum units of the type $S = h(w^T x + b)$, where $h(.)$ is a sigmoid function or logistic function to be bounded [46], [47]. Also, in biological meaning [32], [33], though antibiotics are required to treat severe bacterial infections, misuse will give rise to bacterial resistance and thereby inhibiting the treatment. Inadequate antibiotics will prolong the recovery of patients. Therefore, the consideration of constraints on control gains is very important from the view of engineering and biology, as seen from Fig. 1. The control systems under saturation has been investigated in model predictive control (MPC) [48], networked control systems [31],

[49] and synchronization of coupled systems [30]. In [21], the impacts of control gains on distributed synchronization are shown analytically and an upper bound of control gains was derived. In [25], by simply assuming the control gain to be identical in each node, it is shown that there exists an intermediate control gain to maximize controllability.

It should be noted that, in all the references mentioned above, control gains under consideration are assumed to be bounded, which would largely neglect the typical restrictions in applications. In this paper, we investigate the constraints on control gains in controllability in detail. Not only the case of control gains being bounded is studied, but also the total gain under constraint is investigated.

Remark 2. The problem of resource allocation widely exists in medicine and engineering. For example, in [50], the authors investigate the resource allocation in sensor networks, i. e., how to allocate limited energy, radio bandwidth, and other resources to optimize the value of each node's contribution to the entire network. In [51], scarcity of resources, such as drugs, equipment or time make it difficult to supply the full measure of service and devotion. When circumstances of scarcity occur, it is necessary to face up to the tradeoffs in a fair and compassionate manner. In this paper, controllability of neuronal networks is investigated by taking both boundary and entire costs of control gains into account simultaneously.

C. Problem transformation into a constrained optimization problem

Denote $\mathcal{K} = \text{mean}(\kappa_i), (i \in \mathcal{P})$, where mean(.) is the mean value operator. We convert the problem of controllability of a neuronal network into a COP, where R is here the objective to be minimized and both δ and $\mathcal{K}$ are the constraints.

In the following, some preliminaries of the constrained optimization problem (COP) are given. The COP is formulated as follows: find the decision variables $y = (y_1, \cdots, y_D) \in \mathbb{R}^D$ to minimize the objective function

$$\min \quad f_j(y), y \in \Omega \subseteq S,$$

where Ω is the feasible region and S is the decision space defined by the parametric constraints $Y_i \leq y_i \leq Z_i, i = 1, 2, \cdots, D$. The decision variables y should satisfy s constraints including u inequality constraints

$$q_j(y) \leq 0, j = 1, 2, \cdots, u,$$

and $\iota = s - u$ equality constraints

$$h_j(y) = 0, j = u+1, 2, \cdots, s.$$

The degree of constraint violation of a vector y on the jth constraint is defined as

$$M_j(y) = \begin{cases} \max\{0, q_j(y)\}, & 1 \leq j \leq u, \\ \max\{0, |h_j(y)|\}, & u+1 \leq j \leq s. \end{cases} \tag{4}$$

Then,

$$\Psi(y) = \sum_{j=1}^{s} M_j(y), \tag{5}$$

reflects the degree of constraint violation of the vector y.

We consider the following two cases in this work:
(i) The first case is formulated as follows:

$$\begin{aligned} &\min R = \frac{\lambda^r_{N+1}}{\lambda^r_2}, \\ &\text{subject to: } q_1(y) \leq 0, \\ &\text{subject to: } q_2(y) \leq 0, \end{aligned} \tag{6}$$

where $q_1(y) = \delta - \alpha, q_2(y) = \mathcal{K} - \beta$, $\alpha \in [0, +\infty)$ and $\beta \in [0, +\infty)$. If $\alpha = +\infty$, the problem considered here only focuses on the constraint on $\mathcal{K}$ and minimizes R. If $\alpha \neq +\infty$, the problem considered here focuses on the constraint on $\mathcal{K}$ and δ simultaneously and minimizes the objective R.
(ii) The second case can be written as:

$$\begin{aligned} &\min R = \frac{\lambda^r_{N+1}}{\lambda^r_2}, \\ &\text{subject to: } h_1(y) = 0, \\ &\text{subject to: } q_1(y) \leq 0, \end{aligned} \tag{7}$$

where $h_1(y) = \delta - \alpha, \alpha = 0$ and $q_1(y) = \mathcal{K} - \beta$. The second case is to minimize R as well as make the inequality and equality constraints feasible.

Remark 3. Actually, there are two types of constraints on control gains considered here. The first one is to consider control gains to be bounded like actuator saturation [31], i. e., $\kappa_i (i \in \mathcal{P}) \in [\kappa_{i,\min}, \kappa_{i,\max}]$. In addition to the boundary of control gains, the constraint on entire costs is also included and the total costs injected in the networks have to be allocated in an appropriate way to maximize the controllability of the network.

Remark 4. Most of existing works studied the pinning problem of complex networks with undirected graphs [25], [27], which is a special case of directed graph considered in this paper. In addition, the criteria for pinning synchronization or stabilization of complex networks have been presented [29], [52]. Nonetheless, how to choose driver nodes in complex networks under different numbers of l still remains open [29], [52]. The driver nodes should be given beforehand and then the criteria can be used to check whether the networks are synchronized. Note that the problem of choosing key nodes to control the dynamics of the entire network is a natural combinatorial problem and the design of control gains is a continuous optimization problem [27], [28], which can be solved by evolutionary algorithms. Different from these works, limited costs will affect the selection of key nodes, which will increase the difficulty and the complexity of the problem, as stated its importance in Remark 2. To the best of authors' knowledge, this is one of the first attempts to use constraint optimization evolutionary algorithms (COEAs) to study controllability/pinning control of complex networks/neuronal networks with several constraints on control gains.

III. Modified dynamic hybrid framework (MDyHF) and its encoding scheme

A. Dynamic hybrid framework (DyHF)

COEAs usually include a search algorithm for refining solutions and a constraint-handling technique to make solutions feasible. In [37], a dynamic hybrid framework (DyHF) was proposed, which includes global and local search schemes. The global search model is used to refine the solutions, while the local search model is to motivate the population to approach or enter the feasible region from different directions promptly. In order to fit the search environments adaptively, the global and the local search methods are switched according to the probabilities of proportion of feasible solutions in the population. In addition, traditional differential evolution (DE) works as a search algorithm in global and local search schemes.

Based on the concept of Pareto dominance used in multiobjective optimization, the DyHF transforms a COP into a biobjective optimization problem $\overrightarrow{F}(y) = (f(y), \Psi(y))$ by regarding the degree of constraint violation $\Psi(y)$ as an additional objective. In this way, the original objective function $f(y)$ and the degree of constraint violation $\Psi(y)$ can be considered simultaneously when comparing the solutions of individuals in the population. The performance of DyHF has been verified on 22 benchmark test functions and it is shown that DyHF has the capability to solve all the test functions successfully [37].

Although DyHF is an effective attempt to solve the constrained optimization problem, the search engine in DyHF is not adaptive to fit complicated search circumstances. In particular, the main purpose of a global search scheme in DyHF is exploited to detect more promising regions, where a simple mutation and a crossover scheme from the conventional DE is utilized. Unfortunately, the traditional DE suffers from slow convergence speed, lack of ability to find the global optimum and cannot tune itself to confront with complex optimization problems. Motivated by these points, we preserve the constraint-handling approach of the DyHF due to its efficiency from Pareto dominance and aim to improve the part of the global search algorithm by an adaptive DE. Note that the efficiency of adaptive DE (JaDE) was demonstrated in [39] and here JaDE is utilized to generate offspring to enhance the search ability of the global search scheme and exploit more promising areas, which can efficiently adjust the control parameters in DE and thus make DE adapt to various search situations. In the following, the numerical experiments will validate its performance on different dimensional controllability problem.

B. MDyHF

1) JaDE: JaDE initializes a population of *NP* individuals/particles in a *D*-dimensional search space, which can be used to deal with our optimization problem. Each individual can be viewed as a chromosome, representing a potential solution. After initialization, mutation, crossover, and selection operators are carried out at each generation to guide its population towards the global optimum. The population with its individuals can be written as $\mathbf{P} = (y_{1,n}, y_{2,n}, ..., y_{i,n}), i = 1, 2, ..., NP, n = 0, 1, 2, ..., n_{\max}$, and $y_{i,n} = (y^1_{i,n}, y^j_{i,n}, ..., y^D_{i,n}), j = 1, 2, ..., D$, where n is the generation counter.

JaDE is used to serve as the search engine in the global search of DyHF. In JaDE, a mutation strategy and an external archive are used to provide information of the progress direction. The DE/current-to-ϵbest strategy adopts multiple best solutions to balance the convergence speed and the diversity of the population, which is updated according to the following equation:

$$v_{i,n} = y_{i,n} + \mathcal{F}_i \cdot (y^{\epsilon}_{best,n} - y_{i,n}) + \mathcal{F}_i \cdot (y_{r_1,n} - \tilde{y}_{r_2,n}),$$

where $y^{\epsilon}_{best,n}$ is randomly selected as one of the top $100\epsilon\%$ individuals of the current swarm with $\epsilon = 0.05$. $y_{i,n}$, $y^{\epsilon}_{best,n}$, and $y_{r_1,n}$ are chosen from the current population $\mathbf{P}$. $\tilde{y}_{r_2,n}$ is randomly selected from the union $\mathbf{P} \cup \mathcal{A}$, where $\mathcal{A}$ is an archive and is used to store the recently explored inferior solutions. $\mathcal{F}_i$ and $\mathcal{C}_i$ are the scaling factors associated with the ith individual and crossover probability, respectively. $\mathcal{F}_i$ and $\mathcal{C}_i$ are updated dynamically at each generation according to a Normal distribution and a Cauchy distribution, respectively:

$$\mathcal{F}_i = \mathrm{randc}_i(\varphi_{\mathcal{F}}, 0.1), \mathcal{C}_i = \mathrm{randn}_i(\varphi_{\mathcal{C}}, 0.1).$$

where $\varphi_{\mathcal{F}}$ is the mean value of a Normal distribution and $\varphi_{\mathcal{C}}$ is the mean value of a Cauchy distribution. The two parameters are initialized to be 0.5 and then adjusted at each generation according to:

$$\varphi_{\mathcal{F}} = (1 - w) \cdot \varphi_{\mathcal{F}} + w \cdot \mathrm{mean}_L(S_{\mathcal{F}}),$$
$$\varphi_{\mathcal{C}} = (1 - w) \cdot \varphi_{\mathcal{C}} + w \cdot \mathrm{mean}_A(S_{\mathcal{C}}),$$

where $w = 0.1$ is a constant; $S_{\mathcal{F}}$ and $S_{\mathcal{C}}$ stand for the set of all successful mutation/crossover rates; $\mathrm{mean}_A(\cdot)$ indicates the usual arithmetic mean and $\mathrm{mean}_L(\cdot)$ the Lehmer mean:

$$\mathrm{mean}_L(S_{\mathcal{F}}) = \frac{\sum_{i=1}^{|S_{\mathcal{F}}|} \mathcal{F}_i^2}{\sum_{i=1}^{|S_{\mathcal{F}}|} \mathcal{F}_i}.$$

2) Details of MDyHF: It is worth mentioning that the major algorithmic structure of the DyHF, i. e., the local search strategy and the constraint handling technique, are retained in the MDyHF, the details can be referred to [37].

Remark 5. Note that DE/current-to-ϵbest strategy is adopted in JaDE, which means that $y^{\epsilon}_{best,n}$ is randomly selected as one of the top $100\epsilon\%$ individuals of the current swarm with $\epsilon = 0.05$. Different from the single objective optimization problem, the $100\epsilon\%$ best individuals cannot be measured by only considering objective values. In this paper, we adopt the method in [53] and sort the solutions according to dominance, which is shown in the following way:

A solution i is said to constrained-dominate a solution j, if any of the following conditions is true:

1) Solution i is feasible but solution j is not.

2) Solutions i and j are both infeasible, but solution i has a smaller overall constraint violation.

3) Solutions i and j are feasible and solution i dominates solution j.

Based on JaDE and the above dominance mechanisms, the adaptive global search model is proposed and concentrates on exploring more promising regions and refining the overall objective values of the population. Based on multiobjective optimization, if u_i dominates y_i, the trial vector u_i will replace the target vector y_i according to $\mathcal{C}_i$, else no replacement take places.

By employing the trial vector u_i to remove the inferior target vector y_i, the population **P** is updated through Pareto dominance. Apparently, our modification of the DyHF is technically simple and easily implemented. Even so, the following experimental results will illustrate the encouraging and promising performance of MDyHF. Therefore, MDyHF follows the following steps:

① Set the generation counter $n = 0, f_e = 0$ and obtain an initial population **P** by uniformly and randomly generating from the search space, calculate the objective value f and the constraint violation Ψ for each individual i, and evaluate the number of feasible solutions (NOFS) in **P**.

② Let $\chi = \frac{NP-NOFS}{NP}$ and if $rand(0,1) \geq \chi$ (where $rand(0,1)$ is a uniformly distributed random number between 0 and 1), then the global search with JaDE is implemented to refine feasible solutions, which is equipped with adaptive mechanism; otherwise, the local search is used to detect potential areas of feasible solutions.

③ Compute NOFS in **P** and set $n = n + 1$. If the stopping criterion is met, stop and output the best solution in **P**, else go to ②.

Remark 6. Evolutionary algorithms with an elitism method (the best individual survives with probability one) such as MDyHF can be ensured to find the global optimum with probability 1 if the number of generations tends to infinity, by using the concept of nonhomogeneous Markov chains, as proved in [54]–[56].

C. The encoding scheme of COEAs

In this subsection, an appropriate encoding scheme is used and can be referred to [27]. The encoding scheme consists of two parts with equal dimension size l: the first one is an integer search space to denote the locations of the driver nodes and the second one is a continuous search space to represent their corresponding control gains.

The encoding scheme is illustrated briefly as follows. For example, $l = 2$ areas are selected as driver nodes and the dimension size D of the swarm is $D = 2 \times l$. The parameter space is set as $Y_i = 0, Z_i = N$. Let an individual/particle be $y = (y_1, y_2, y_3, y_4) = (31.5, 21.1, 52.3, 12.9)$. Since the node index of the network is an integer, the round operators are performed in the first part of each particle and thus $y = (31.5, 21.1, 52.3, 12.9)$. Then, the regions (nodes) $i = 32$ and $i = 21$ are chosen as driver nodes and controlled, i. e., $\phi_{\mathcal{P}}(32) = 1$ and $\phi_{\mathcal{P}}(21) = 1$. The second part of the encoding scheme indicates that the control gains of the regions 32 and 21 are 52.3 and 12.9, respectively, implying $\kappa_{32} = 52.3$ and $\kappa_{21} = 12.9$. Obviously, the encoding scheme is simple to implement.

Remark 7. As stated in [27], the search range of each dimension is assumed to be the same and therefore can be written as $\triangle y = (Z_i - Y_i)$. In order to identify the driver nodes from $N = 53$ as a function of l, there are C_N^l distinct combinations, which is a natural NP-hard problem and it is difficult to adopt a Brute-force method to select the driver nodes. In addition, even if the locations of the driver nodes can be determined *a priori*, the problem is reduced into an l-dimensional continuous optimization problem. One effective method to handle a NP-hard problem is evolutionary computation algorithms. In this paper, we use MDyHF to study the controllability of neuronal networks.

IV. MAIN RESULTS

In this section, several examples are presented to verify the performance of the MDyHF in comparison with two COEAs and several methods from graph theory. The controlling regions are identified in microscopic and macroscopic ways.

A. Methods for determining the locations of driver nodes

In this subsection, the following methods are used for detecting the locations of driver nodes/controlling regions.

(i) **Degree-based methods**. The controlling regions are selected according to out-degree in an ascending or a descending way, which are named the ascending and the descending degree-based methods, respectively.

(ii) **Betweenness centrality (BC)-based methods**. Descending and ascending BC-based methods are used here.

(iii) **Closeness-based methods**. Two types of closeness-based methods, i. e. descending and ascending closeness-based strategies are used.

(iv) **Evolutionary algorithm-based methods**. COEAs are used to select driver nodes and design their control gains. Two evolutionary computation approaches, the CMODE [38] and the DyHF [37] are used to compared with the MDyHF. The CMODE and the DyHF have been recently developed and have shown their advantages over some well-known COEAs [37], [38].

B. Parameter settings of COEAs

If not mentioned differently, the parameter setting of COEAs is adopted as follows. The maximum fitness evaluation $f_{e,\max}$ is set to $f_{e,\max} = \eta * D$ and $D = 2 * l$ is the dimension size. $\eta = 18,750$ is a predefined constant. When comparing the performance among COEAs, COEAs will be repeated 20 times independently for eliminating random discrepancy and terminated when COEAs algorithms attain $f_{e,\max}$. When showing the advantages of MDyHF over statistical methods, MDyHF will be repeated 10 times when MDyHF achieves $f_{e,\max}$. The parameter settings of the CMODE and the DyHF [37], [38], respectively. The parameter setting of the MDyHF is according to [37], [39]. Similar to [25], in the degree-based, the BC-based and the closeness-based, the control gains in all the vertices are supposed to be identical and are gradually adjusted by a stepsize $\frac{N}{100}$ gradually.

C. Comparisons of the MDyHF with evolutionary algorithms and statistical methods

In this subsection, the performance of the proposed MDyHF is compared with other COEAs and statistical methods in Sec. IV-A. The COEAs used for comparison are the CMODE and the DyHF [37], [38]. When compared with COEAs, we

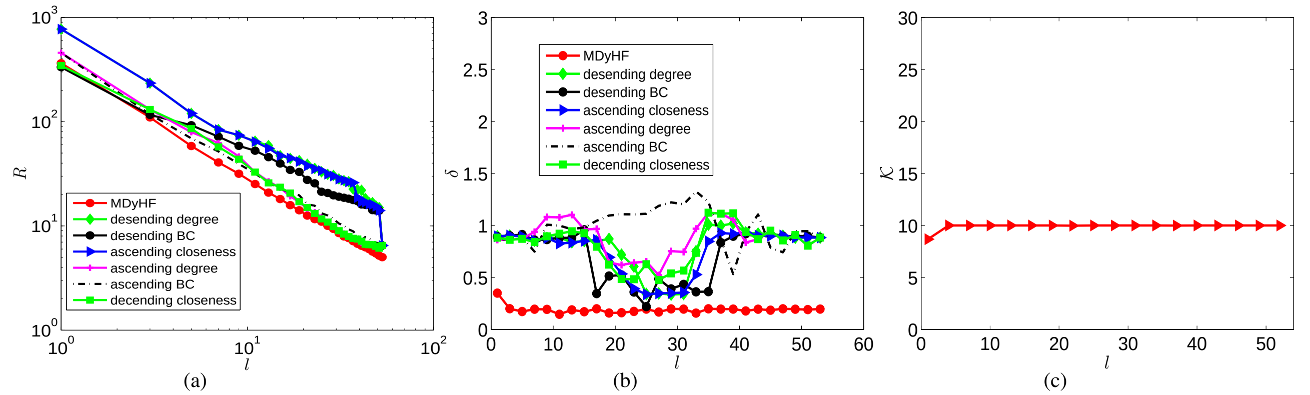

Fig. 2. Minimizing R using different schemes, $\alpha = 0.2$ and $\beta = 10$ as a function of l. (a) Comparison of R with different schemes as a function of l; (b) Comparison of δ with different schemes as a function of l; (c) The average control gain $\mathcal{K}$ using MDyHF.

TABLE I

SEARCH RESULT COMPARISONS AMONG THREE ALGORITHMS FOR DIFFERENT l OF DRIVER NODES IN THE NEURONAL NETWORK WITH A NETWORK SIZE $N = 53$, SEE FIG. 1. THE CALCULATION OF Q IS GIVEN IN (8). ALL THE ALGORITHMS ARE RUN 20 TIMES, $\eta = 12500$ AND $\beta = 30$. THE BEST RESULTS AMONG THE THREE ALGORITHMS ARE SHOWN IN BOLD FONTS.

		$\alpha = 0$						$\alpha = 0.2$					
		CMODE		DyHF		MDyHF		CMODE		DyHF		MDyHF	
		R	Ψ	R	Ψ	R	Ψ	R	Ψ	R	Ψ	R	Ψ
$l = 6$	Mean	46.4711	0	39.0959	0	**35.0913**	0	39.1083	0	29.5861	0	**28.7236**	0
	Best	38.6713	0	31.0939	0	**30.0323**	0	34.0034	0	28.0146	0	**28.1822**	0
	Q	42.3922	0	34.8661	0	**32.4634**	0	36.4666	0	28.7896	0	**28.4516**	0
$l = 12$	Mean	27.7295	0	21.3476	0	**16.7967**	0	23.7688	0	**17.9131**	0	18.1309	0
	Best	24.0568	0	16.0881	0	**15.461**	0	22.6745	0	15.3074	0	**15.2964**	0
	Q	25.8279	0	18.5322	0	**16.115**	0	23.2152	0	16.5591	0	16.6535	0
$l = 18$	Mean	19.9953	0	16.1677	0	**11.5577**	0	18.441	0	14.3858	0	**13.1699**	0
	Best	17.9219	0	12.5422	0	**10.8525**	0	16.6528	0	10.9857	0	**10.2588**	0
	Q	18.9302	0	14.2401	0	**11.1995**	0	17.5241	0	12.5713	0	**11.6236**	0
$l = 24$	Mean	18.1467	0	12.5788	0	**8.4081**	0	14.2305	0	10.7726	0	**10.0962**	0
	Best	15.4222	0	9.491	0	**7.7428**	0	11.8573	0	8.341	0	**7.4466**	0
	Q	16.7291	0	10.9264	0	**8.0686**	0	12.9898	0	9.4792	0	**8.6708**	0
$l = 30$	Mean	14.0262	0	10.9395	0	**6.577**	0	12.2628	0	8.7522	0	**8.0139**	0
	Best	12.7251	0	7.3187	0	**6.1259**	0	11.2307	0	6.8406	0	**6.759**	0
	Q	13.3598	0	8.9478	0	**6.3474**	0	11.7354	0	7.7376	0	**7.3598**	0
$l = 36$	Mean	12.9602	0	9.3311	0	**5.5889**	0	11.2212	0	7.3659	0	**6.4517**	0
	Best	11.0792	0	6.3573	0	**4.9809**	0	8.8852	0	5.8935	0	**5.0716**	0
	Q	11.9828	0	7.702	0	**5.2761**	0	9.9851	0	6.5887	0	**5.7201**	0
$l = 42$	Mean	12.5129	0	9.7678	0	**4.5777**	0	10.3934	0	6.4943	0	**5.1529**	0
	Best	8.7404	0	7.4653	0	**4.1078**	0	8.8442	0	4.808	0	**4.1159**	0
	Q	10.4579	0	8.5393	0	**4.3364**	0	9.5876	0	5.5879	0	**4.6053**	0
$l = 48$	Mean	11.0912	0	7.6824	0	**3.8361**	0	8.9326	0	5.0001	0	**4.5939**	0
	Best	8.8865	0	5.9346	0	**3.2418**	0	7.61	0	4.1705	0	**3.3113**	0
	Q	9.9278	0	6.7522	0	**3.5265**	0	8.2448	0	4.5665	0	**3.9002**	0

compare the objective value R and the constraint handling results Ψ. When compared with statistical methods, the objective value R and the value of δ are compared, since the constraint on control gains in all the statistical methods are identical and satisfy the gain constraint.

As stated in [27], both the mean value and the best value of the solutions are important for measuring the reliability of the algorithm, the following measure is considered to incorporate them together

$$Q = \sqrt{Best \times Mean}. \quad (8)$$

Obviously, Q should be made as small as possible.

Firstly, we show the comparison results of COEAs. Table 1 shows the comparison results of CMODE, DyHF and MDyHF under different dimension size. Table 1 reveals that Ψ achieves zero in all the three algorithms under different l, implying that all the three algorithms find feasible solutions. Hence, we only focus on the objective value R. Clearly, the MDyHF performs best among the three algorithms. The DyHF performs better than the CMODE but works worse than the MDyHF. The improvement of MDyHF arises from the introduction of JaDE into the global search scheme and retains other parts of efficient strategies in the MDyHF. After the local model finds possible feasible solutions, the adaptive global search method can well adjust itself to various search situations, thereby improving the performance of DyHF. From the above observations, the MDyHF is the most powerful algorithm among the three COEAs. The adaptive global search method will help the MDyHF to refine solutions and maintain good convergence speed, while the local search scheme is kept to

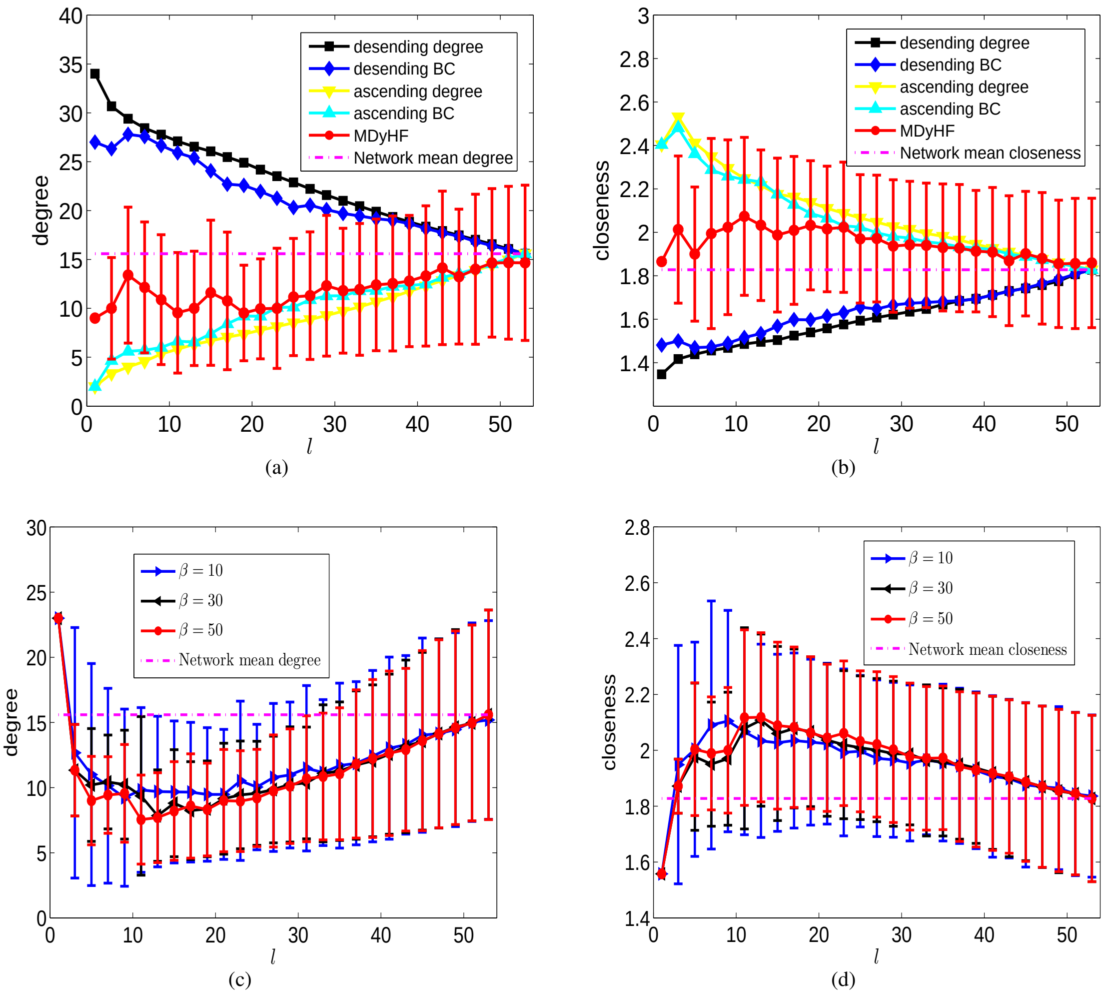

Fig. 3. The degree and closeness information of driver nodes with different schemes as a function of l. (a) The mean values of degree information of driver nodes with $\alpha = 0.2, \beta = 10$ as a function of l; (b) The mean values of closeness information of driver nodes with $\alpha = 0.2, \beta = 10$ as a function of l; (c) The mean values of degree of driver nodes under different β as a function of l, when $\alpha = +\infty$; (d) The mean values of closeness of driver nodes under different β as a function of l, when $\alpha = +\infty$.

explore feasible solutions. Due to the reliability of the MDyHF, it is used to carry out the following simulations.

Next, the MDyHF is compared with the statistical schemes in Sec. IV-A. The statistical methods only deal with minimizing R and δ is neglected, different from the MDyHF. Also, since the control gain of each node in neuronal networks is assumed to be identical in all the statistical methods, the search boundary of the control gain can be simply assumed to be $[0, \beta]$. In the following, we consider $\alpha = 0.2$ and $\beta = 10$. It is noteworthy that other parameters will yield similar results. The comparison of different β will be presented in Sec. IV-F.

Fig. 2(a) shows that the MDyHF performs better than the other methods in terms of R in most cases. The MDyHF offers poorer performance than a few methods in terms of R in several cases, since MDyHF consider the constraint on δ while the statistical methods neglect the effect of δ. Fig. 2(c) shows that the average control gain satisfies the constraint $\beta = 10$. Fig. 2(a) shows that, at the initial stage, the descending BC-based method performs best. When l increases, the descending degree-based, the descending BC-based, and the ascending closeness-based strategies are becoming worse. Conversely, the ascending degree-based, the ascending BC-based and the descending closeness-based strategies perform well in most cases, as depicted in Fig. 2(a).

In Fig. 2(b), although some statistical methods might work better than the MDyHF in terms of R, the MDyHF performs best among all the methods in terms of δ and the statistical methods cannot find feasible solutions in all the cases. In order to satisfy the constraints on δ and β, the MDyHF has to encounter a tradeoff between losing the performance of minimizing R to satisfy constraints. Also, when l crosses a threshold, the MDyHF always salsifies the constraint δ. From Figs. 2(a) and 2(b), it can be observed that the values of δ are much smaller than R, especially when l is small. This is consistent with the findings regarding synchronizability of complex networks [44], [45]. However, when l increases, δ is turning more important since R is becoming smaller and the value of R is comparable to the value of δ.

TABLE II

CONTROLLING TIMES AND ITS COMMUNITY OF EACH NODE BELONGING TO WHEN OPTIMIZING R UNDER DIFFERENT β, WHEN $\alpha = \infty$. ξ CAN BE SEEN FROM EQ. (9).

$\beta = 50$			$\beta = 30$			$\beta = 10$		
Node Name	ξ	Community	Node Name	ξ	Community	Node Name	ξ	Community
VPc	50	Auditory	21a	51	Visual	VPc	53	Auditory
2	50	Somato-motor	VPc	49	Auditory	36	53	Frontolimbic
AMLS	48	Visual	2	49	Somato-motor	21a	50	Visual
21a	48	Visual	AMLS	47	Visual	Hipp	49	Frontolimbic
21b	47	Visual	Sb	46	Frontolimbic	AMLS	48	Visual
PS	47	Visual	21b	44	Visual	SII	48	Somato-motor
Sb	46	Frontolimbic	AAF	44	Auditory	AAF	47	Auditory
ALLS	44	Visual	Hipp	44	Frontolimbic	Sb	46	Frontolimbic
AAF	44	Auditory	ALLS	43	Visual	2	45	Somato-motor
Hipp	44	Frontolimbic	PS	43	Visual	PLLS	44	Visual
3a	42	Somato-motor	SIV	42	Somato-motor	21b	44	Visual
Tem	41	Auditory	Tem	41	Auditory	ALLS	42	Visual
SIV	41	Somato-motor	SII	41	Somato-motor	PS	41	Visual
1	40	Somato-motor	PLLS	40	Visual	P	40	Auditory
DLS	38	Visual	3a	40	Somato-motor	19	39	Visual
PSb	37	Frontolimbic	P	39	Auditory	3a	39	Somato-motor
P	36	Auditory	DLS	37	Visual	Tem	37	Auditory
SII	35	Somato-motor	1	37	Somato-motor	1	37	Somato-motor
4	35	Somato-motor	PSb	37	Frontolimbic	PSb	35	Frontolimbic
RS	35	Frontolimbic	PMLS	35	Visual	DLS	34	Visual
PLLS	34	Visual	4	35	Somato-motor	SIV	33	Somato-motor
AII	34	Auditory	RS	33	Frontolimbic	PMLS	32	Visual
PMLS	30	Visual	AII	32	Auditory	4	31	Somato-motor
PFCMiI	30	Frontolimbic	19	31	Visual	AII	29	Auditory
19	29	Visual	20b	30	Visual	18	27	Visual
VLS	29	Visual	PFCMiI	30	Frontolimbic	3b	27	Somato-motor
20b	27	Visual	Enr	30	Frontolimbic	35	27	Frontolimbic
SSAo	27	Somato-motor	3b	26	Somato-motor	Enr	27	Frontolimbic
3b	26	Somato-motor	SSAo	25	Somato-motor	RS	25	Frontolimbic
17	25	Visual	18	24	Visual	AES	24	Visual
Enr	25	Frontolimbic	VLS	24	Visual	20b	23	Visual
18	23	Visual	17	22	Visual	AI	22	Auditory
AI	21	Auditory	36	22	Frontolimbic	17	21	Visual
36	21	Frontolimbic	7	20	Visual	VLS	20	Visual
4g	20	Somato-motor	AI	20	Auditory	4g	20	Somato-motor
61	20	Somato-motor	4g	20	Somato-motor	PFCMiI	19	Frontolimbic
7	19	Visual	PFCI	19	Frontolimbic	Ig	19	Frontolimbic
PFCI	19	Frontolimbic	61	17	Somato-motor	7	15	Visual
Ig	16	Frontolimbic	6m	14	Somato-motor	SSAo	15	Somato-motor
5Bm	14	Somato-motor	SSAi	14	Somato-motor	EPp	13	Auditory
SSAi	14	Somato-motor	Ig	14	Frontolimbic	61	13	Somato-motor
6m	12	Somato-motor	5Bm	13	Somato-motor	Ia	12	Frontolimbic
Ia	12	Frontolimbic	AES	11	Visual	SSAi	11	Somato-motor
PFCMd	11	Frontolimbic	PFCMd	10	Frontolimbic	5Bm	10	Somato-motor
AES	9	Visual	EPp	9	Auditory	PFCI	10	Frontolimbic
EPp	8	Auditory	Ia	9	Frontolimbic	20a	9	Visual
Cga	7	Frontolimbic	35	7	Frontolimbic	6m	7	Somato-motor
5BI	6	Somato-motor	Cga	6	Frontolimbic	PFCMd	5	Frontolimbic
35	5	Frontolimbic	20a	5	Visual	5Am	4	Somato-motor
5Am	4	Somato-motor	5BI	4	Somato-motor	5BI	4	Somato-motor
20a	3	Visual	5Am	3	Somato-motor	5AI	3	Somato-motor
CGp	2	Frontolimbic	5AI	2	Somato-motor	Cga	3	Frontolimbic
5AI	1	Somato-motor	CGp	1	Frontolimbic	CGp	0	Frontolimbic

D. Microscopic identification of controlling regions using the MDyHF under gain constraint

In Figs. 3(a) and 3(b), we show the mean values of the degree and closeness information of the driver nodes by various methods as a function of l, when $\alpha = 0.2, \beta = 10$. We find that the mean values of the driver nodes selected by the MDyHF are intermediate, belonging to the range of mean values of the ascending and the descending degree-based schemes, which are also less than the mean values of the network degree. This phenomenon shows that the nodes with neither a large nor a small degree are optimal to be selected as driver nodes. As l increases, the mean values of driver nodes selected by the MDyHF gradually increase and finally converge to the mean value of network degree. In summary, one should pick more nodes with a large degree as l increases, while the nodes with a small degree should also be chosen. The findings observed here, which largely depend on l, are different from the work in [23].

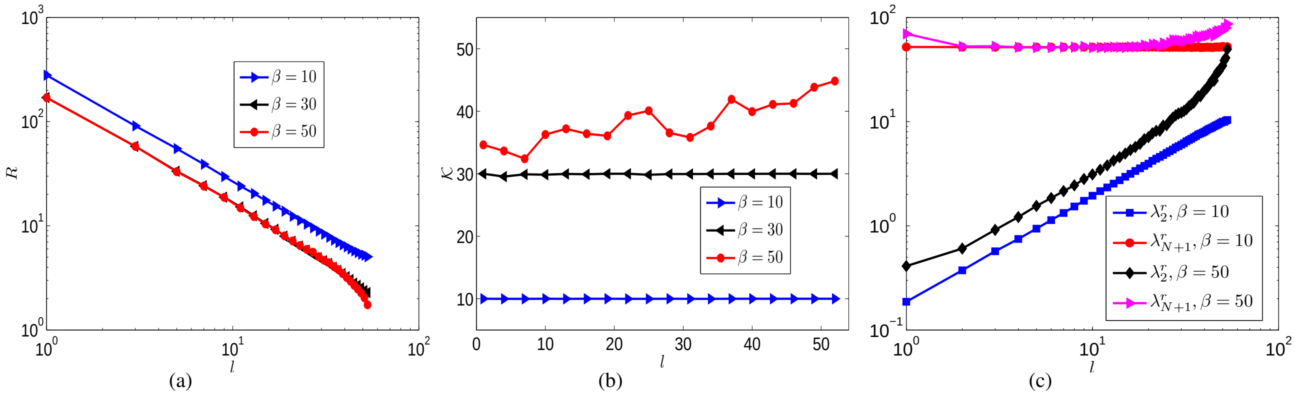

Fig. 4. Optimizing R by the MDyHF with different β as a function of l, when $\alpha = +\infty$. (a) Comparison of R with different β as a function of l; (b) Comparison of δ with different β as a function of l, when $\alpha = +\infty$; (c) Comparison of λ_2^r and λ_{N+1}^r with different β as a function of l, when $\alpha = +\infty$.

E. Macroscopic identification of controlling regions using the MDyHF under gain constraint

Due to the efficiency of the MDyHF, the identification of controlling regions of the neuronal network with different β is studied now. Denote

$$\xi_i = \sum_{l=1}^{N} \phi_{\mathcal{P}}(i), \tag{9}$$

which calculates the times of each node to serve as driver nodes as a function of l. The regions with a large ξ_i are more important to control the network. After control of the neuronal network with an increase of l (stepsize 1), ξ_i are sorted under $\beta = 10, \beta = 30$ and $\beta = 50$. The results are illustrated in Table 2. There the regions are sorted according to their importance in the neuronal networks. Table 2 shows that, there exist some differences for the pinned times of each node in the three cases of β. The controlling regions are spread widely in four communities. It can be found that the regions such as VPc and 21a are important to control the neuronal network to a desired state, which are different from the usual hubs [9]. Meanwhile, the regions such as CGp and 5AI are unsuitable to serve as driver nodes.

F. Comparisons of the MDyHF with different β

In this subsection, enhancing controllability of the neuronal network is examined using the MDyHF under different constraints β. The comparisons of R and $\mathcal{K}$ are presented in Figs. 4(a) and 4(b). Fig. 4(a) shows that, when $\beta = 50$, the MDyHF performs best. However, the differences between the lines of $\beta = 30$ and $\beta = 50$ are close to each other. When l is large, the line of $\beta = 50$ decays faster than $\beta = 30$. Fig. 4(b) shows that when $\beta = 50$, there is no need to use the allowed control gains completely, i. e. $\mathcal{K} < 50$. However, different from the case of $\beta = 50$, when $\beta = 10$ and $\beta = 30$, the allocations of control gains should be used completely to enhance the controllability of the neuronal network, i. e. $\mathcal{K} = 10$ or $\mathcal{K} = 30$, respectively. Also, one should carefully allocate the resources to each node to make the controllability maximal. This phenomenon shows that there exists an intermediate control cost to maximize controllability of neuronal networks, which verifies the phenomena in biological observations and engineering background [33], [51], as illustrated in Remark 2. In summary, the MDyHF can enhance the controllability of the neuronal network, while keeping the solutions in a feasible space.

In Figs. 3(c) and 3(d), the mean values of degree and closeness of the driver nodes under different l and β are shown. As l increases, the mean values of the degree of the driver nodes attain minimum and then increases, which shows a clear transition of the mean values of the degree of the driver nodes, i. e., from nodes with a large degree to nodes with a small degree and again nodes with a large degree. As an increase of β (the constraint is more restrictive), the driver nodes tend to be chosen from the nodes with a large degree. The reason for this is that to enhance controllability of complex networks, the nodes with a small degree requires a large control input [27]. Therefore, when the constraint on control gains is considered, we have to control the nodes with a large degree and allocate control gains economically.

In the following, the dependence of R, λ_2^r, λ_{N+1}^r on l and β are investigated in Figs. 4(a) and 4(c). As plotted in Fig. 4(a), we get $R(l) \propto l^{-\gamma}$. In addition, in order to minimize R whenever $\beta = 10$ or $\beta = 50$, λ_2^r should be increased as much as possible, while λ_{N+1}^r should be suppressed as much as possible. Fig. 4(c) shows that the shape of R depends largely on λ_2^r when $\beta = 10$ and $\beta = 50$. When $l \to N$ and $\beta = 50$, $\lambda_2^r \approx \lambda_{N+1}^r$ and λ_2^r grows faster than λ_{N+1}^r, which leads to $R \approx 1$. However, when $\beta = 10$, the focus is on keeping λ_{N+1}^r stable and enhance λ_2^r as much as possible. λ_2^r of $\beta = 50$ is larger than that of $\beta = 10$. This observation means that when more resources are allowed, it is desirable to enlarge λ_2^r and keep λ_{N+1}^r stable to enhance controllability. Therefore, when more resources are allowed, it is more efficient to find ways to enlarge λ_2^r. The above observations show that controlling λ_2^r plays a more important role in controllability than λ_{N+1}^r.

V. Conclusion

In this study, we investigate the problem of controllability of a realistic neuronal network of the cat under constraints on control gains by utilizing a modified dynamic hybrid framework (MDyHF). The problem of detecting driver nodes under constraints on control gains is converted into a constrained optimization problem (COP), in which two measures of controllability R and δ are viewed as an objective and a constraint, respectively and the constraint on control gains is regarded as a constraint, thereby the objective and the constraints are incorporated into one unified framework. By adding the JaDE with Pareto dominance into the dynamic hybrid framework (DyHF), the MDyHF can fit the search circumstances adaptively. By compared with two recent COEAs and statistical methods, the experimental results demonstrate the effectiveness of the MDyHF. By using the MDyHF, the controlling regions under gain constraints are identified. Some interesting findings about constraints on control gains, the objective R, the number of driver nodes l and the eigenvalues of the extended topology graph are illustrated by simulations. We show that there exist intermediate control costs to enhance controllability of neuronal networks and the control costs should be carefully allocated to maximize controllability of neuronal networks. The effects of constraints on β on controllability of neuronal networks are also investigated and it is shown that the variation of β does affect the selection of controlling regions and controllability of neuronal networks. We find that the controlling regions vary under different β.

Many extensions and refinements of this work are possible. These include the analysis of data on other kinds of biological networks, the development of more powerful COEAs to handle the controllability of neuronal networks and the consideration of other types of real world constraints. With the arrival of new methods, it will be feasible to apply our methods to more natural systems, and thereby to further enhancing our understanding of how to control a directed and weighted complex network with a suitable control cost and a small number of driver nodes.

VI. Acknowledgements

This research is supported by 973 Project (2009CB320600), the National Natural Science Foundation of China (60825303, 60834003, 61021002, 61203235) and the Key Laboratory of Integrated Automation for the Process Industry (Northeastern University), SUMO (EU), IRTG 1740 (DFG) and the Alexander von Humboldt Foundation of Germany.

References

[1] A. Arenas, A. Guilera, J. Kurths, Y. Moreno, and C. Zhou, “Synchronization in complex networks,” *Physics Reports*, vol. 469, pp. 93–153, 2008.

[2] M. Rubinov and O. Sporns, “Complex network measures of brain connectivity: Uses and interpretations.” *NeuroImage*, vol. 52, pp. 1059–1069, 2010.

[3] E. Bullmore and O. Sporns, “Complex brain networks: graph theoretical analysis of structural and functional systems.” *Nature Reviews Neuroscience*, vol. 10, p. 1, 2009.

[4] J. Piersa, F. Piekniewski, and T. Schreiber, “Theoretical model for mesoscopic-level scale-free self-organization of functional brain networks,” *IEEE Trans. On Neural Networks*, vol. 21, pp. 1747–1758, 2010.

[5] K. Zhan, H. Zhang, and Y. Ma, “New spiking cortical model for invariant texture retrieval and image processing,” *IEEE Trans. On Neural Networks*, vol. 20, pp. 1980–1986, 2009.

[6] J. Thangavelautham and G. D’Eleuterio, “Tackling learning intractability through topological organization and regulation of cortical networks,” *IEEE Trans. On Neural Networks and Learning Systems*, vol. 23, pp. 552–564, 2012.

[7] C. Zhou, L. Zemanová, G. Zamora-López, C. Hilgetag, and J. Kurths, “Hierarchical organization unveiled by functional connectivity in complex brain networks,” *Phys Rev Lett*, vol. 97, p. 238103, 2006.

[8] M. Müller, C. Hilgetag, and M. Hütt, “Organization of excitable dynamics in hierarchical biological networks,” *PLoS Comput. Biol.*, vol. 4, p. e1000190, 2008.

[9] O. Sporns, C. Honey, and R. Kotter, “Identification and classification of hubs in brain networks,” *PLoS ONE*, vol. 10, p. e1049, 2007.

[10] J. Gómez-Gardeñes, G. Zamora-López, Y. Moreno, and A. Arenas, “From modular to centralized organization of synchronization in functional areas of the cat cerebral cortex,” *PLoS ONE*, vol. 5, p. e12313, 2010.

[11] J. Liang, Z. Wang, Y. Liu, and X. Liu, “Global synchronization control of general delayed discrete-time networks with stochastic coupling and disturbances,” *IEEE Transactions on Systems, Man and Cybernetics, Part B*, vol. 38, pp. 1073–1083, 2008.

[12] J. Liang, Z. Wang, and X. Liu, “State estimation for coupled uncertain stochastic networks with missing measurements and time-varying delays: the discrete-time case,” *IEEE Trans. on Neural Networks*, vol. 20, pp. 781–793, 2009.

[13] Y. Liu, Z. Wang, J. Liang, and X. Liu, “Synchronization and state estimation for discrete-time complex networks with distributed delays,” *IEEE Transactions on Systems, Man and Cybernetics, Part B*, vol. 38, pp. 1314–1325, 2008.

[14] H. Karimi and H. Gao, “New delay-dependent exponential h_∞ synchronization for uncertain neural networks with mixed time delays.” *IEEE Trans. on Systems, Man, and Cybernetics, Part B*, vol. 40, pp. 173–185, 2010.

[15] G. Chen and F. Lewis, “Distributed adaptive tracking control for synchronization of unknown networked lagrangian systems.” *IEEE Trans. on Systems, Man, and Cybernetics, Part B*, vol. 41, pp. 805–816, 2011.

[16] H. Zhang, T. Ma, G. Huang, and Z. Wang, “Robust global exponential synchronization of uncertain chaotic delayed neural networks via dual-stage impulsive control.” *IEEE Trans. on Systems, Man, and Cybernetics, Part B*, vol. 40, pp. 831–844, 2010.

[17] B. Shen, Z. D. Wang, and X. Liu, “Bounded h-infinity synchronization and state estimation for discrete time-varying stochastic complex networks over a finite-horizon.” *IEEE Trans. on Neural Networks*, vol. 22, pp. 145–157, 2011.

[18] N. Kopell and B. Ermentrout, “Chemical and electrical synapses perform complementary roles in the synchronization of interneuronal networks,” *Proc Natl Acad Sci(USA)*, vol. 101, pp. 15 482–15 487, 2004.

[19] P. Uhlhaas and W. Singer, “Neural synchrony in brain disorders: Relevance for cognitive dysfunctions and pathophysiology,” *Neuron*, vol. 52, pp. 155–168, 2006.

[20] C. Zhou, L. Zemanová, C. Hilgetag, and J. Kurths, “Structure function relationship in complex brain networks expressed by hierarchical synchronization,” *New J Phys*, vol. 9, p. 178, 2007.

[21] Y. Tang, H. Gao, W. Zou, and J. Kurths, “Distributed synchronization in networks of agent systems with nonlinearities and random switchings,” *IEEE Trans. Systems, Man, and Cybernetics-Part B: Cybernetics*, accepted, 2012.

[22] W. Ren and R. Beard, *Distributed Consensus in Multi-vehicle Cooperative Control, Communications and Control Engineering Series*. Springer-Verlag, London., 2008.

[23] Y. Liu, J. Slotine, and A. Barabasi, “Controllability of complex networks,” *Nature*, vol. 473, pp. 167–173, 2011.

[24] M. Egerstedt, “Degrees of control,” *Nature*, vol. 473, pp. 158–159, 2011.

[25] F. Sorrentino, M. Bernardo, F. Garofalo, and G. Chen, “Controllability of complex networks via pinning,” *Phys. Rev. E*, vol. 75, p. 046103, 2007.

[26] X. Wang and G. Chen, “Synchronization in scale-free dynamical networks: Robustness and fragility,” *IEEE Trans. Circuits Syst. I: Regular Papers*, vol. 49, pp. 54–62, 2002.

[27] Y. Tang, H. Gao, J. Kurths, and J. Fang, “Evolutionary pinning control and its application in UAV coordination,” *IEEE Trans. On Industrial Informatics, accepted*, 2012.

[28] Y. Tang, Z. Wang, H. Gao, S. Swift, and J. Kurths, “A constrained evolutionary computation method for detecting controlling regions of cortical networks,” *IEEE-ACM Trans. on Computational Biology and Bioinformatics*, accepted, 2012.

[29] J. Lu, J. Kurths, J. Cao, N. Mahdavi, and C. Huang, “Synchronization control for nonlinear stochastic dynamical networks: Pinning impulsive strategy,” *IEEE Trans. on Neural Networks and Learning Systems*, vol. 23, pp. 285–292, 2012.

[30] Y. Tang, Z. D. Wang, W. K. Wong, J. Kurths, and J. Fang, “Multiobjective synchronization of coupled systems,” *Chaos*, vol. 21, p. 025114, 2011.

[31] Y. Cao and Z. Lin, “Stability analysis of discrete-time systems with actuator saturation by a saturation-dependent lyapunov function,” *Automatica*, vol. 39, pp. 1235–1241, 2003.

[32] J. Harrison and T. Svec, “The beginning of the end of the antibiotic era? part ii. proposed solutions to antibiotic abuse,” *Quintessence International*, vol. 29, pp. 223–9, 1998.

[33] T. Pickering, J. Gurwitz, D. Zaleznik, J. Noonan, and J. Avorn, “The appropriateness of oral fluoroquinolone-prescribing in the long-term care setting,” *Journal of the American Geriatrics Society*, vol. 42, pp. 28–32, 1994.

[34] Z. Michalewicz and M. Schoenauer, “Evolutionary algorithm for constrained parameter optimization problems,” *Evol. Comput.*, vol. 4, pp. 1–32, 1996.

[35] C. A. C. Coello, “Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art.” *Computer Methods in Applied Mechanics and Engineering*, vol. 191, pp. 1245–287, 2002.

[36] Z. Cai and Y. Wang, “A multiobjective optimization-based evolutionary algorithm for constrained optimization,” *IEEE Trans. Evol. Comput.*, vol. 10, pp. 658–75, 2006.

[37] Y. Wang and Z. Cai, “A dynamic hybrid framework for constrained evolutionary optimization,” *IEEE Trans. Systems, man, and cybernetics - Part B: Cybernetics*, vol. 42, pp. 203–217, 2012.

[38] ——, “Combining multiobjective optimization with differential evolution to solve constrained optimization problems,” *IEEE Trans. Evol. Comput.*, vol. 16, pp. 117–134, 2012.

[39] J. Q. Zhang and A. C. Sanderson, “Jade: adaptive differential evolution with optional external archive,” *IEEE Trans. on Evolutionary Computation*, vol. 13, pp. 945–958, 2009.

[40] J. Scannell and M. Young, “The connectional organization of neural systems in the cat cerebral cortex,” *Curr. Biol.*, vol. 3, pp. 191–200, 1993.

[41] J. Scannell, C. Blakemore, and M. Young, “Analysis of connectivity in the cat cerebral cortex.” *J Neurosc*, vol. 15, pp. 1463–1483, 1995.

[42] C. Hilgetag, G. Burns, M. O’Neill, J. Scannell, and M. Young, “Anatomical connectivity defines the organization of clusters of cortical areas in the macaque monkey and the cat,” *Phil Trans R Soc London B*, vol. 355, pp. 91–110, 2000.

[43] G. Zamora-López, C. Zhou, and J. Kurths, “Cortical hubs form a module for multisensory integration on top of the hierarchy of cortical networks,” *Frontiers in Neuroinformatics*, vol. 4, pp. 1–12, 2010.

[44] D. U. Hwang, M. Chavez, A. Amann, and S. Boccaletti, “Synchronization in complex networks with age ordering,” *Phys. Rev. Lett.*, vol. 94, p. 138701, 2005.

[45] S. Son, B. Kim, H. Hong, and H. Jeong, “Dynamics and directionality in complex networks,” *Phys Rev Lett*, vol. 103, p. 228702, 2009.

[46] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, *Neurocomputing: foundations of research, Learning internal representations by error propagation*. MIT Press Cambridge, MA, USA., 1988.

[47] J. Liang, Z. Wang, Y. Liu, and X. Liu, “Robust synchronization of an array of coupled stochastic discrete-time delayed neural networks,” *IEEE Trans. on Neural Networks*, vol. 19, pp. 1910–1921, 2008.

[48] M. V. Kothare, V. Balakrishnan, and M. Morari, “Robust constrained model predictive control using linear matrix inequalities,” *Automatica*, vol. 32, pp. 1361–1379, 1996.

[49] B. Zhou, G. R. Duan, and Z. L. Lin, “Global stabilization of the double integrator system with saturation and delay in the input,” *IEEE Trans. on Circuits and Systems-I: Regular Papers*, vol. 57, pp. 1371–1383, 2010.

[50] G. Mainland, D. C. Parkes, and M. Welsh, “Decentralized, adaptive resource allocation for sensor networks,” in *Proceedings of the 2nd USENIX/ACM Symposium on Networked Systems Design and Implementation (NSDI ’05)*, vol. 2, 2005, pp. 315–328.

[51] E. W. Kluge, “Resource allocation in healthcare: Implications of models of medicine as a profession,” *MedGenMed*, vol. 9, p. 57, 2007.

[52] W. Yu, G. Chen, and J. Lü, “On pinning synchronization of complex dynamical networks,” *Automatica*, vol. 45, pp. 429–435, 2009.

[53] K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms*. Wiley., 2002.

[54] G. Rudolph, “Convergence properties of canonical genetic algorithms,” *IEEE Trans. on Neural Networks*, vol. 5, pp. 96–101, 1994.

[55] ——, *Convergence Properties of Evolutionary Algorithms*. Hamburg: Kovac., 1997.

[56] D. V. Veldhuizen and G. B. Lamont, “Evolutionary computation and convergence to a pareto front,” in *Late Breaking Papers at the Genetic Programming 1998 Conference*. Stanford University Bookstore, 1998.