Optimizing target node set for the control energy of directed complex networks

Hong Chen and Ee Hou Yong

ABSTRACT

The energy needed in controlling a complex network is a problem of practical importance. Recent works have focused on the reduction of control energy via strategic placement of driver nodes, or by decreasing the cardinality of nodes to be controlled. However, how target nodes are arranged with respect to control energy has yet been explored. Here, we propose an iterative method based on stiefel manifold optimization of selectable target node matrix to reduce control energy. We derived the matrix derivative gradient needed for the search algorithm in a general way, and searched for target nodes which result in reduced control energy, given that driver nodes placement is fixed. Our findings reveal that when the path distances from driver nodes to target nodes are minimised, control energy is optimal. We also applied the algorithm to various model and real networks. The simulation results show that when compared to heuristic selection strategies of choosing target nodes, the control energy is reduced by a few orders of magnitude. Our work may be applicable to a social network voter model, where we are interested in optimizing the control effort needed to influence a fractional number of individuals’ opinions.

Introduction

Complex networks have been extensively studied in recent decades owing to its modelling utility towards social systems, biological systems, Internet, and man-made technological systems. Usually, these networks are modelled as coupled system of ordinary differential equations. The state vector elements are represented as nodes or vertices in graphs and the coupling interaction between state vector elements are represented as links or edges in graphs. The state vectors of such a coupled ordinary differential equation serve to represent a myriad of quantities, depending on the complex dynamical system being considered at hand. For example, they can represent the probability of a person being infected in a social network system, or they can represent the expression level of a gene in a regulatory network. The motivation to study and understand these complex systems can be traced to our desire to obtain control over them. In this case, control refers to exerting influence on the networked system via external control signals to steer the state vector of the networked system from its arbitrary initial, to a predefined goal state vector in finite time $[t_0, t_f]$. It follows then, that if the nodes of a network could be steered towards the predefined goal state vector in finite time, the network is deemed controllable.

Achieving control over a complex network with as few control signals as possible is desirable. In 2011, Liu et al proved that the unmatched nodes from maximum matching algorithm of a bipartite representation of a complex network needed to receive external influence to ensure network structural controllability. The $N_d$ number of unmatched nodes in need of control signals are thus termed the minimum driver node set, or simply the driver nodes. Soon after, Sun and Motter explored the numeric success rate of network controllability when numerically computing the controllability Gramian matrix when using energy optimal control signal to steer the network. They found that for a complex network with more than a handful of nodes, using the minimum driver node set is computationally insufficient as the computation of the controllability Gramian will become ill-conditioned or nearly singular. Instead, beyond using the $N_d$ number of minimum driver nodes, additional control signals are needed to ensure numeric success when computing the controllability Gramian.

Since the problem of minimum driver node set to guarantee controllability for an arbitrary sized complex network was solved, several other prominent research works have followed. Notably, the investigation into the energy cost required by a control signal has been a subject of investigation by several groups. In these studies, the energy cost is defined as a measure of proportionate effort exerted by the control signal over the considered time. It was found that if the number of control signals is small, the energy cost demanded of each of the signal could be prohibitively high. In fact, the energy cost is reduced exponentially as the number of control signal increases. Thus, attaching additional control signals onto a networked system beyond the minimum driver node set is one way to achieve network control with reduced energy cost.

The way in which the additional control signals are attached can also result in reduced control energy.
Altafini considered the eigenvalues of the network and proposed strategies for selecting the placement of additional control signals to minimise control energy cost\textsuperscript{20}. Chen et al analysed stems, obtained from minimum driver node set\textsuperscript{6}, and calculated all possible direct shortest paths from driver nodes to non-driver nodes to obtain the Longest Control Chain (LCC), which is the longest path from the obtained all possible shortest direct paths\textsuperscript{21}. They found that by adding additional control signals in such a way that the length of the LCC is minimised, control energy can be reduced significantly. Li et al proposed an algorithm using matrix derivative projected gradient descent to iteratively search for the energy optimal placement of control signals\textsuperscript{17}. This optimization model was later simplified by Ding et al\textsuperscript{15} and Li et al\textsuperscript{16}. In 2018, Li et al proposed an improved and generalised approach based on previous works to again obtain the energy-optimal placement of control signals\textsuperscript{22}. The problem of reducing control energy by strategic placement of additional control signals has been extensively researched. However, in all of these works, complete control was considered, where the control signals steer the full node set towards the predefined goal state vector.

While full control may be necessary in some types of engineered systems\textsuperscript{23}, controlling just a subset of nodes (typically termed target control or targeted control) may be more sensible in large complex dynamical systems. In 2014, Gao et al proposed an alternate k-walk theory and a greedy algorithm to obtain the minimum number of driver nodes to control just a subset of the full node set in a complex network\textsuperscript{24}. In 2017, Liu et al considered the controllability of the giant connected component in a directed complex network\textsuperscript{25}. With regards to energetic considerations, Klickstein et al showed that the energy cost scales exponentially with the cardinality of the target node set\textsuperscript{19}. Gao et al proposed an algorithm to obtain the placement of control signals to optimise the energy cost when controlling just a subset of nodes in directed complex networks\textsuperscript{26}. In all of these works, the choice of target nodes was pre-selected, and to the best of our knowledge, how the selection of target nodes affects energy cost in a complex network is an open question that has yet been explored.

Thus, different from all previous works, we wish to ask a slightly different question: How can we pick the target node set in such a way that the control energy is minimised? To that end, we will be employing a cost function optimization model based on projected gradient descent, similar to earlier works\textsuperscript{15–17,22,26}. The main difference between our work and the existing literature is the variable matrix being optimized: While previous works have focused on optimizing the choice of input signal (matrix $B$), we optimise the choice of target node set (matrix $C$). Furthermore, previous derivations of the index notation gradient information uses the I-Chain rule, which can be difficult to understand. Here, we derive the index notation gradient information using the standard chain rule and product rule in a general way, which is simpler to follow.

In this paper, we will be examining how the choice of target node set could be optimized to minimise the energy cost function. Using the formulated energy cost function, we derive the matrix derivative of the energy cost function with respect to matrix $C$, which is the energy cost function gradient information. With the gradient information obtained, we perform an iterative search using the trace-constraint-based projected gradient method (TPGM) proposed in ref.$^{22}$ to obtain the node set which are energetically favourable. We then compare the energy cost that we would have gotten from choosing to target control nodes using a heuristic selection scheme such as random selection, and node degree-based selection. Our simulation results show that the solution obtained from TPGM reduces the energy cost by a few orders of magnitude.

**Problem Formulation**

In standard complex network controllability literature, we are interested in studying $N$ coupled system of equations whose dynamics are linear time invariant.

$$
\dot{x}(t) = Ax(t) + Bu(t) \\
y(t) = Cx(t)
$$

(1)

where $x(t) \in \mathbb{R}^{N \times 1}$ is the time-varying state vector, $y(t) \in \mathbb{R}^{P \times 1}$ is the time-varying output state vector of the subset of nodes that we want to target control (where $P \leq N$), $u(t) \in \mathbb{R}^{M \times 1}$ is a time-varying external control signal which we use to drive the network (where $M \leq N$). The time invariant $A \in \mathbb{R}^{N \times N}$ matrix represents the network topology, where for directed complex network we have nonzero matrix element $a_{ij}$ when there is a directed link from node j to node i, and zero element if no link exists from node j to node i. Since many complex systems tend to enjoy passive stability, our modelling also allows for self-links, where $a_{ii}$ is a negative real number\textsuperscript{12}. The time invariant $B \in \mathbb{R}^{N \times M}$ matrix reflects the coupling between nodes and $M$ number of external control signals $u(t)$, where matrix element $b_{ij}$ = 1 if node i receives a time-varying control signal from control input j, and zero element otherwise. Finally, we have the $C \in \mathbb{R}^{P \times N}$ matrix which tells us the choice of the $P$ number of subset of nodes that we wish to target control, where matrix element $c_{ij} = 1$ if node j is in the $i$-th node out of all possible $P$ nodes that we want to target control. We note that matrix $B$ and matrix $C$ are column and row linear independent, where if $b_{ij} = 1$ ($c_{ij} = 1$), then no other nonzero entries may exist for column $j$ and row $i$ in matrix $B$ ($C$). We require this linear independence to reflect our modelling choice for one-to-one connection between control signals and nodes, and for one-to-one connection to individually target control each of the $P$ number of nodes\textsuperscript{19}. 

---

2/12
While we could choose from an infinite number of driving term $u(t)$, here we choose to focus on energy-optimal control signal, which could be derived from optimal control theory\textsuperscript{10,19}.

$$u^*(t) = B^T e^{A^T(t-f)} C^T (CWC^T)^{-1} (y_f - C e^{A(t-f)} x_0)$$  \hspace{1cm} (2)

The $N \times N$ controllability Gramian matrix $W = \int_0^t e^{A(t-f)} B B^T e^{A^T(t-f)} dt$ is an important quantity in control theory, well-known to be real, symmetric, and semi-positive definite. In computation, we can use the controllability Gramian matrix to check that the choice of matrix $B$ is ensuring network (target) controllability if the (output $(CWC^T)$) controllability Gramian matrix is invertible\textsuperscript{9,19}.

We could also determine the energy cost for using control signal (2) to drive the network by considering the cost function $J = \int_{t_0}^t u^T(t) u(t) dt$\textsuperscript{9}. This is the energy cost function that we are interested in optimizing, with respect to the choice of control variable $B$. Similar to previous works which considered trace constraint projected matrix gradient cost function optimization\textsuperscript{15–17,22,26}, here we assume that the initial state vector $x_0 \sim \mathcal{N}(0,1)$. Thus, we can formulate the energy cost as $\mathcal{E}(t_f,B,C) = \mathbb{E} \left[\int_{t_0}^t u^T(t) u(t) dt\right]$, where $t_0 = 0$ is implicit and assumed throughout the rest of the paper. While the energy cost is a function of the control trajectory (the state space pathway between $x_0$ and $y_f$), here, for the energy cost function considered at hand, we are considering the expected energy cost over all realisations of initial state vector $x_0$, picked under standard normal distribution. We note also, that in general, the cost function $\mathcal{E}(t_f,B,C)$ is dependent on matrix $B$ and scalar $t_f$. However, in this paper, we are interested in optimizing the cost function with respect to matrix $C$ while keeping input matrix $B$ and final time $t_f$ fixed. Thus, for clarity we drop this dependence and write the cost function as just $\mathcal{E}(C)$ or $\mathcal{E}(CC^T)$ throughout the paper.

We optimise the cost function as follows:

$$\begin{align*}
\text{min} \quad & \mathcal{E}(C) \\
\text{s.t.} \quad & \text{tr}(CC^T) = P
\end{align*}$$  \hspace{1cm} (3)

where the trace boundary constraint $\text{tr}(CC^T) = P$ represents the limitation we impose on the solution produced by the gradient descent iterative algorithm to stay on the manifold $\text{tr}(CC^T) = P$.

**Algorithm**

In this subsection, we present the trace-constraint-based projected gradient method (TPGM) formulated by Li et al\textsuperscript{22}, and change only the focus on matrix $B$ to matrix $C$.

**Preliminaries**

Let $C^T$ be the basis of matrix $C$. We can obtain $\tilde{C}^T$ by performing Gram Schmidt orthogonalization on matrix $C^T$\textsuperscript{27}. Define projection operator $\mathcal{T}_{C^T} = (I_N - \tilde{C}^T \tilde{C})$, where $I_N$ is the $N \times N$ identity matrix, and we note that $\mathcal{T}_{C^T}$ projects any arbitrary matrix onto the space that is perpendicular to the manifold $\text{tr}(CC^T) = P$.

For two arbitrary matrices $\mathbb{A}$ and $\mathbb{B}$ which have the same dimension, an angle between them can be defined as:

$$\theta = \arccos \left( \frac{\text{tr}(\mathbb{A}^T \mathbb{B})}{\|\mathbb{A}\|_F \|\mathbb{B}\|_F} \right)$$  \hspace{1cm} (4)

where $0 \leq \theta \leq \pi$, and we note that the matrices are perpendicular if $\theta = \pi/2 = 90^\circ$. $\|\cdot\|_F$ denotes frobenius norm.

The TPGM algorithm is given as follows:

1. (Initialization). Initialise $C^T_{k=0}$ as a random $N \times P$ matrix, where the subscript $k$ denotes the $k$-th iteration step.

2. (Projected gradient descent). Compute the $k$-th iteration energy gradient $\frac{\partial \mathcal{E}(C^T_k)}{\partial C^T} = \nabla \mathcal{E}(C^T_k)$ and update the non-normalised next iterative solution as:

$$\tilde{C}^T_{k+1} = C^T_k - \eta \cdot (I_N - \tilde{C}^T_k \tilde{C}_k) \nabla \mathcal{E}(C^T_k)$$  \hspace{1cm} (5)

where $\eta$ is a small scalar quantity. Note that the overhat quantity $\tilde{C}^T_k$ is non-normalised and is different from the normalised quantity that we want, $C^T_k$.
We obtain the energy cost function by substituting energy-optimal control signal $u$:

$$E = \frac{P}{tr(C^T_k C_{k+1})} C_{k+1}^T$$  \hspace{1cm} (6)

(3) (Normalization). Obtain $C_{k+1}^T$ by normalizing $C_{k+1}^T$ onto the manifold surface $tr(CC^T) = P$.

(4) (Convergence check). At each $k$-th iteration, the searched solution is updated as $C_{k+1}^T = C_k^T + \triangle C_k^T$. To check for projected gradient descent solution convergence, we calculate the angle difference, $\theta_k$, between energy gradient matrix $-\nabla \delta(C_k^T)$ and the small incremental update to the $k$-th searched solution, $\triangle C_k^T$ using equation (4):

$$\cos(\theta_k) = -\frac{\nabla \delta(C_k^T)^T \nabla \delta(C_k^T)}{||\nabla \delta(C_k^T)||_F \cdot ||\nabla \delta(C_k^T)||_F}$$  \hspace{1cm} (7)

(5) (Terminating Condition). If $\cos(\theta_k) > \xi$, go to step (2) and update $k = k + 1$; otherwise, terminate the iteration and denote the obtained optimal solution $C^*$. The gradient descent solution is deemed to have converged when $\cos(\theta_k)$ approaches zero, or equivalently, when $\theta_k$ approaches $\pi/2 = 90^\circ$. The quantity $\xi$ represents a small positive scalar quantity.

**Remarks**

**Remark 1.** In TPGM step (1), by random initialization of matrix $C_0^T$, we mean that we start with a $N \times P$ matrix of zeros and randomly choose to fill matrix element $(C^T)_i j$, where node $i$ is chosen to be the $j$-th target node. Once node $i$ is chosen, we maintain row and column linear independence, and thus do not allow for the selection of the same node $i$ to be target controlled. In the numerical experiments that we tried, while starting from a pure random dense matrix is possible when system size is small or the network topology is sparse, it is computationally inefficient. Furthermore, when starting from a pure random dense matrix, we are not able to obtain a sensible sparse optimal matrix at the end, when converting from dense optimal solution $C^*$ to sparse discrete optimal solution (see section Selecting discrete optimal target node set from $C^*$).

**Remark 2.** In TPGM step (2), the numerical value of the learning rate or step size $\eta$ is chosen empirically. When $\eta$ is too large, convergence is not guaranteed and the algorithm will fail. While convergence is guaranteed when $\eta$ is sufficiently small, the time taken to complete the iterative search will suffer if $\eta$ is too small. Typically, we choose the learning rate to be between $1e-8$ and $1e-3$, depending on the fraction $P/N$ to be target controlled as well as the complex network topology. In general, the learning rate $\eta$ can be varied to speed up the iterative process, for example starting at $\eta = 1e-8$, and then changed to $\eta = 1e-4$ when enough iterations have been run. As observed during experimentation, $\eta$ will be convergent inversely proportionately to the numerator of the $\cos(\theta_k)$, $tr\left(\nabla \delta(C_k^T)^T \nabla \delta(C_k^T)\right)$.

**Remark 3.** In TPGM step (5), $\xi$ refers to a small positive quantity, for example, $\xi = 1e-2$, which denotes that the iterative search algorithm has sufficiently converged—that is, the angle between the gradient matrix, $\nabla \delta(C_k^T)$, and the projected gradient matrix, $\nabla \delta(C_{k+1}^T)$, are perpendicular. For some networks where the algorithm is unable to converge towards $\xi = 1e-2$, we may relax this condition to $\xi = 1e-1$.

**Results**

We obtain the energy cost function by substituting energy-optimal control signal $u^*(t)$ into the cost function $\delta(C^T) = \mathbb{E}[(u^T(t)u(t))dt]$:

$$\delta(C^T) = tr((CW^T)^{-1}yy^T) + tr(C^T(CWC^T)^{-1}Ce^{At}e^{A^Tt}).$$  \hspace{1cm} (8)

We obtain the energy gradient as follows:

$$\frac{\partial \delta(C^T)}{\partial C^T} = -2WC^T(CWC^T)^{-1}yy^T(CWC^T)^{-1}$$  
$$-2WC^T(CWC^T)^{-1}Ce^{At}e^{A^Tt}C^T(CWC^T)^{-1}$$  
$$+2e^{At}e^{A^Tt}C^T(CWC^T)^{-1}$$  \hspace{1cm} (9)
For derivation, see Supplementary Notes 1 and 2. Using (8) and (9) and applying it to the TPGM algorithm, we solve for energy-optimal target node set to target control and obtain the optimal solution, \( C^* \). Note that for the time horizon, we are using \( t_f = 2 \), and the final output state vector is \( y_f = [1, 1, \ldots, 1]^T \).

**Numerical Experiment on Elementary Topologies**

To understand the arrangement of energy optimal target node set, we perform the TPGM iterative search on elementary topologies\(^6\), which are the basic network structures that form the basis of a complex network being structurally controllable, when the driver nodes are found. They are: directed stem, circle, and dilation. A stem requires just a control signal, placed at the root, to become mathematically (but not necessarily numerically) controllable\(^{11,25,29}\). A circle is controllable with just one control signal attached to any of the nodes in the circle. A dilation requires a minimum of two driver nodes to become controllable.

We select the driver nodes to be placed in such a way that their path distances are evenly spaced, which actually corresponds to the most energy efficient set up as the longest path distance from driver to non-driver nodes is minimised\(^{21}\). We maintain the driver nodes to be fixed and we allow, for controlling 66.7\% of the nodes in the elementary topologies, the target node set to be variable. Figure 1 shows that, for fixed placement of driver nodes, the energy optimal target node set as found by the TPGM algorithm corresponds to target nodes which have minimised path distance from driver nodes to target nodes. Our finding is consistent within the literature, which states that control energy cost increases exponentially with path distance\(^{21,30}\).

For each elementary topology, we repeat the experiment independently for 1000 times. The success rates of the TPGM algorithm finding the energy optimal target node set solutions are: for Figure 1 (a1) - probability 58.9\%, (a2) - 66.0\%, (a3) - 85.9\%, (b1) - 41.0\%, (b2) - 40.1\%, (b3) - 67.4\%, (c1) - 91.2\%, (c2) - 69.1\%. By analyzing the obtained optimal solution matrix \( C^* \), which is in general a dense matrix of real numbers, we developed new schemes for selecting the discrete optimal target node set \( C^*\_{\text{discrete}} \) based on the obtained final \( k \)-th step solution, \( C^* \). By discrete matrix, we mean that each matrix element is either one or zero and adheres to column and row linear independence.

**Selecting discrete optimal target node set from \( C^* \)**

TPGM will iteratively update the initial proposed solution \( C_0 \) in the direction of quickest decreasing energy cost based on cost function derivative with respect to matrix variable \( C \). When the search has finally converged, the obtained optimal target node set \( C^* \) corresponds to reduced control cost of a dense matrix of real numbers, which corresponds to many-to-many connections from output nodes to complex network nodes of varying link strength. Based on the obtained optimal solution \( C^* \), the challenge is to obtain a one-to-one connection of output nodes and nodes with unity link strength, a sparse discrete optimal solution, \( C^*\_{\text{discrete}} \), while maintaining the characteristics of reduced control cost.

We propose two methods to find \( C^*\_{\text{discrete}} \) from \( C^* \). The first method is based on suppressing insignificant matrix elements in \( C^* \), and then evaluating the normalised quantity \textit{importance index} vector, \( r \):

\[
\begin{align*}
    r &= \frac{[r_1, r_2, \ldots, r_N]}{\max(r_1, r_2, \ldots, r_N)} \\
    &= \frac{\sum_j |[C^*]_{ij}|}{\max(r_1, r_2, \ldots, r_N)} \\
    &\quad \text{where} \quad r_i = \sum_j |[C^*]_{ij}| \quad \text{is the non-normalised importance score of node} \quad i \quad \text{being selected as a target node, and we are taking the row summation of the absolute of the transpose matrix,} \quad C^T. \quad |[C^*]_{ij}| \quad \text{represents the numerical contribution towards} \quad C^* \quad \text{for node} \quad i, \quad \text{target node} \quad j. \quad \text{Therefore, to find discrete optimal solution that remains similar to} \quad C^*, \quad \text{we want to find the nodes with the highest numerical contributions.}
\end{align*}
\]

In our experimentation, we found that directly passing \( C^* \) into (10) to find \( C^*\_{\text{discrete}} \) usually does not allow us to find the optimal target node set. This is because of the row summation of the absolute of the numerical contribution. For example, if row \( a \) has predominantly absolute values of around 0.2 in all its columns, while row \( b \) has an absolute value of around 0.9 in one of its columns, and mostly negligible numerical values close to zero in all of its other columns, then by equation (10), node \( a \) will be ranked higher than node \( b \). Based on experimentation with elementary topologies, we find that numerical contribution characteristics similar to row \( b \) usually correspond to an optimal target node. Thus, we want the row summation to reflect that: we suppress the numerical values in each matrix element of \( C^* \) if they fall below a certain value. (For a more detailed discussion, see Supplementary Information.)

We compute the suppressed optimal matrix by setting \( [C^*]_{ij} = 0 \) iff \( [C^*]_{ij} <= d \), where \( d \) is computed from the standard deviation of the absolute of all matrix elements \( [C^*]_{ij} \), and \( d = \{0.0, 0.1, 0.2, \ldots, 1.1, 1.2, \ldots, 3.0\} \). For each suppressed matrix, we check that the rank of the matrix is \( P \), before considering it as a viable candidate solution. If the suppressed matrix has rank \( P \), then we pass the suppressed matrix into (10), and pick \( P \) nodes with the highest importance index. The case when \( k = 0 \) is similar to the proposed importance index vector formulation in refs\(^{16,17,28} \) for finding optimal driver nodes.

The second method is based on selecting the matrix elements with the largest absolute numerical values. The process is as follows: First, begin with optimal matrix transpose, \([C^*]^T\). Then, search for the absolute largest matrix element in each
column, and order the columns in descending order, keeping track of the associate row indices. Start with a \( N \times P \) zero matrix, \( C^T_{\text{discrete}} \), and set the matrix element for position \((i, j)\) to be one, starting from the columns with the largest absolute matrix elements \(|[C^*]^T_{ij}|\). At each step, check that for assigning position \((i, j)\) to be one, no other nonzero element exists along row \(i\) or along column \(j\). If there exists another nonzero matrix element along the row \(i\) or column \(j\), then do not assign position \((i, j)\) to be one, and record down column \(j\) which did not get filled. If \(\text{rank}(C^T_{\text{discrete}}) = P\), then stop the process; otherwise, repeat the process described for unfilled columns \(j\)'s for the case of second largest absolute matrix elements, third largest,....., until \(\text{rank}(C^T_{\text{discrete}}) = P\).

At the end, we pass all the obtained discrete target node set candidate solutions into the objective cost function, equation (8), and pick the target node set which yields the lowest energy cost, denoting it \([C^*_{\text{discrete}}]^T\), where \(t = \{1, 2, ..., 10\}\) represents each independent iterative search. For the simulation results showing \(C^*_{\text{discrete}}\), we choose the best solution out of all \([C^*_{\text{discrete}}]^T\).

Figure 1. Energy optimal target node set when driver nodes are fixed. (a1) depicts an \(N=9\) directed stem network, with \(M=3\) driver nodes depicted by the cyan (grey in grayscale) control signals, and \(P=6\) optimal target nodes depicted by the magenta (dark grey) output nodes. (a2-a3) represent \(\{N=9,M=2,P=6\}\) and \(\{N=6,M=1,P=4\}\) directed stem network respectively. (b1-b3) represent circle topology with \(\{N=9,M=3,P=6\}\), \(\{N=9,M=2,P=6\}\), and \(\{N=6,M=1,P=4\}\) respectively. (c1-c2) represent dilation with \(\{N=9,M=2,P=6\}\) and \(\{N=9,M=3,P=6\}\) respectively.
**Simulation Results**

Next, we apply the TPGM to different complex networks such as random networks (ER network), scale-free networks (SF network), as well as various real networks spanning a diverse range: electronic circuit networks, food web networks, and social networks. For each network, we perform the iterative search 10 times and compare the performance of the obtained solution $C^*$ as well as the associate discrete optimal solution $C_{\text{discrete}}$ and compare them to heuristic selection schemes such as random selection (repeated over 100 independent realizations) and degree-based selection of target nodes. We select 40% of the nodes to be driver nodes using the standard way\(^6,7\) to ensure controllability, and then picking the remaining nodes randomly. For each network, once the driver nodes are picked, they remain fixed.

For the model networks of $N=100$ SF ($\gamma = 2.8$) and ER, with average degrees $k_{av} = 2.5$, we plot the control energy needed for controlling target node set of varying cardinality $P/N = \{0.1, 0.2, ..., 0.9, 1.0\}$ in Figures 2 (a) and (b). Consistent with the findings of ref.\(^{19}\), we can observe that control energy scales exponentially with the cardinality of the target node set, regardless of the target node selection scheme chosen. However, when comparing the control energy of various selection strategies, our results show that generally, as compared to heuristic selection schemes, the energy cost for controlling target node set as found by TPGM, $C^*$ and $C_{\text{discrete}}$, is lower by a few orders of magnitude. For the result of Figure 2(b), the performance of the in-degree descending selection scheme (meaning that we choose $P/N\%$ of nodes that are ranked within the top $P/N\%$ of largest in-degrees) in the region $P/N = \{0.1, 0.2, 0.3, 0.4\}$ is generally similar to optimal target node set $C_{\text{discrete}}$. This is likely due to the fact that the driver nodes which were randomly selected happened to coincide with the nodes which have high in-degrees, which results in reduced path distances between driver nodes and target nodes, and thus reduced control energy cost.

![Figure 2](image.png)

Figure 2. Control energy needed for controlling target node set of increasing cardinality. Optimal target node set control energy, $\mathcal{E}(C^*)$ and $\mathcal{E}(C_{\text{discrete}})$, is generally a few orders of magnitude better than selecting target node set from heuristic selection schemes. (a) SF network and (b) ER network.

**Simulation Results on Real Networks**

Finally, we apply TPGM to real networks and model networks of $N=300$, and compile the obtained results in table 1, comparing with the control energy of initial random selection $\mathcal{E}(C_0)$ and random selection $\mathcal{E}(C_{\text{rand}})$. With the exception of Circuit-s838 and Teacher-student, where we drive the network with $M/N = 0.5$ and $M/N = 0.6$ fractional number of driver nodes to target control $P/N = 0.7$ and $P/N = 0.75$ fractional number of nodes, each of the network is driven by $M/N = 0.4$ fractional number of driver nodes to target control $P/N = 0.6$ fractional number of nodes. The energy cost of target node set selected by degree-based selection scheme is presented in table 2.

Examining table 1, we observe that for any network, starting from its initial, $\mathcal{E}(C_0)$, TPGM algorithm iteratively updates the $C_k$ matrix in the direction of quickest decreasing control energy until we obtain the convergent solution, $\mathcal{E}(C^*)$, where the energy cost is typically reduced by a few orders of magnitude. Based on section Selecting discrete optimal target node set from $C^*$, we can choose the target nodes from the obtained optimal solution, $C^*$, to obtain $C_{\text{discrete}}$. While the conversion from dense real matrix solution to sparse discrete matrix will in general result in increased control energy, comparing the energy cost $\mathcal{E}(C_{\text{discrete}})$ to heuristic selection strategies of target nodes, such as $\mathcal{E}(C_{\text{rand}})$ in table 1 and degree-based selection in table 2,
we observe that the control energy compares favourably.

**Discussion**

When the learning rate $\eta$ is chosen appropriately, TPGM is convergent (see Remark 2 and Remark 3). We illustrate the convergence of the iterative search algorithm in figures 3 (a) and (b). In figure 3 (a), the k-th step cosine angle, $\cos(\theta_k)$, between energy gradient $\nabla \mathcal{E}(C_k^i)$ and projected energy gradient $\mathcal{T}_{\hat{C}} \nabla \mathcal{E}(C_k^i)$, approaches zero as TPGM iteration increases. The algorithm is deemed to have converged when the gradients are perpendicular. In figure 3 (b), we observe that each iteration of TPGM brings the matrix $C_k$ towards a lower energy cost, starting with a steep decrease in energy cost which becomes less steep with each iteration, until convergence.

The computational complexity of TPGM is $O(N^3)$ owing to its iterative matrix multiplication. It should be noted that the computations of the controllability Gramian matrix, $W = \int_0^T e^{A^T}BB^Te^{A^T}d\tau$ as well as the calling of the exponential function for computing $e^{At}$ ($e^{A^Tt}$) are costly, which can result in bottlenecks when calculating them in the loop each time. An efficient way to code the TPGM is to compute the controllability Gramian $W$ and $e^{At}$ ($e^{A^Tt}$) outside the while loop and storing them as variables to be retrieved within the loop. Furthermore, instead of directly computing the integral, an efficient way to compute the controllability Gramian is to use the method of ref. 38.

The condition number of the controllability Gramian matrix plays an important role in our numerical experiments. When increasing the number of control signals can lower the condition number and render the computation of the Gramian feasible. Note that the condition number is dependent on matrices $A$, $B$, as well as time horizon $t_f$. For some networks, despite increasing the number of driver nodes, the condition number of the Gramian may still be infeasibly high. To lower the condition number, we can normalise the link weights of the connection matrix, $\{a_{ij}\}$, by dividing throughout a normalization constant. For networks whose connection strengths are not

**Table 1.** Control energy needed in various networks when different strategies are applied to selecting target node set $C$. $\mathcal{E}(C_0)$ is the control energy mean of 10 independent initialization of random matrix. $\mathcal{E}(C^*)$ is the control energy mean of the converged final k-th step solution. $\mathcal{E}(C_{\text{discrete}}^*)$ is the lowest control energy from the obtained all possible candidate solutions $\mathcal{E}([C_{\text{discrete}}]^t)$, for $t = 1, 2, ..., 10$. $\mathcal{E}(C_{\text{rand}})$ is the control energy mean of 100 independent realizations of selecting target nodes randomly. Additional information such as standard deviations and mean of $\mathcal{E}([C_{\text{discrete}}]^t)$ is presented in Supplementary Information.

| Network          | Model  | $N$ | edges | $\mathcal{E}(C_0)$   | $\mathcal{E}(C^*)$   | $\mathcal{E}(C_{\text{discrete}}^*)$ | $\mathcal{E}(C_{\text{rand}})$ |
|------------------|--------|-----|-------|----------------------|----------------------|--------------------------------------|-------------------------------|
| **Electronic circuit** | SF300  | 300 | 750   | 8.11E07              | 2.88E03              | 2.65E05                             | 1.48E08                       |
|                  | ER300  | 300 | 750   | 1.88E06              | 5.34E02              | 1.24E05                             | 2.27E06                       |
| **Food web**     | Circuit-s838 | 512 | 819   | 6.02E05              | 4.48E02              | 1.97E04                             | 1.45E06                       |
|                  | Circuit-s420 | 252 | 399   | 6.01E04              | 1.33E02              | 6.92E03                             | 8.97E05                       |
|                  | Circuit-s208 | 122 | 189   | 2.34E04              | 6.90E01              | 1.99E03                             | 1.74E08                       |
| **Social Influence** | StMarks | 54  | 356   | 4.83E03              | 6.31E01              | 2.77E02                             | 5.54E03                       |
|                  | Maspalomas | 24  | 82    | 1.76E03              | 3.72E01              | 3.65E01                             | 2.59E04                       |
|                  | Rhode    | 19  | 53    | 8.07E02              | 2.00E01              | 3.65E01                             | 1.46E05                       |
| **Social**       | Phys-discuss-rev | 231 | 565   | 1.90E04              | 1.99E02              | 4.11E03                             | 1.49E05                       |
|                  | Teacher-student | 60  | 94    | 1.41E02              | 3.96E01              | 6.60E01                             | 1.78E02                       |
|                  | Phys-friend-rev | 228 | 506   | 2.61E04              | 2.46E02              | 3.59E03                             | 2.55E04                       |
|                  | Highschool | 70  | 366   | 4.32E04              | 1.91E02              | 5.45E02                             | 4.06E04                       |
Table 2. Control energy needed in various networks when degree-based selection strategies are applied to selecting target node set \( C \). \( \delta(C_{\text{in,asc}}) \) refers to choosing target node set in order of ascending nodes in-degree. \( \delta(C_{\text{in,dsc}}) \), \( \delta(C_{\text{out,asc}}) \), and \( \delta(C_{\text{out,dsc}}) \) follows analogously, where dsc refers to order of descending and out refers to nodes out-degree.

| Network     | \( N \) | \( \text{edges} \) | \( \delta(C_{\text{in,asc}}) \) | \( \delta(C_{\text{in,dsc}}) \) | \( \delta(C_{\text{out,asc}}) \) | \( \delta(C_{\text{out,dsc}}) \) |
|-------------|--------|---------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|
| **Model**   |        |                     |                               |                               |                               |                               |
| SF300       | 300    | 750                 | 2.99E08                       | 1.35E07                       | 2.29E08                       | 5.23E07                       |
| ER300       | 300    | 750                 | 4.18E06                       | 9.18E04                       | 6.51E05                       | 2.39E06                       |
| **Electronic circuit** |      |                     |                               |                               |                               |                               |
| Circuit-s838 | 512    | 819                 | 4.77E05                       | 4.99E05                       | 4.06E04                       | 4.80E05                       |
| Circuit-s420 | 252    | 399                 | 8.67E04                       | 6.68E04                       | 8.78E05                       | 9.59E04                       |
| Circuit-s208 | 122    | 189                 | 3.20E06                       | 2.65E04                       | 4.00E04                       | 3.20E06                       |
| **Food web** |        |                     |                               |                               |                               |                               |
| StMarks     | 54     | 356                 | 4.73E02                       | 4.53E02                       | 7.16E02                       | 5.19E02                       |
| Maspalomas  | 24     | 82                  | 7.52E02                       | 1.34E03                       | 1.04E03                       | 9.37E02                       |
| Rhode       | 19     | 53                  | 6.52E07                       | 1.35E06                       | 6.11E07                       | 5.94E06                       |
| **Social Influence** |     |                     |                               |                               |                               |                               |
| Phys-discuss-rev | 231  | 565                 | 4.29E04                       | 2.62E04                       | 1.11E04                       | 2.95E05                       |
| Teacher-student | 60   | 94                  | 8.89E01                       | 7.21E02                       | 4.14E02                       | 9.41E01                       |
| **Social**  |        |                     |                               |                               |                               |                               |
| Phys-friend-rev | 228  | 506                 | 6.57E03                       | 1.02E05                       | 7.49E03                       | 6.99E04                       |
| Highschool  | 70     | 366                 | 9.45E04                       | 1.55E03                       | 4.10E03                       | 1.55E05                       |

Figure 3. Convergence of TPGM algorithm being performed on the electronic circuit network, Circuit-s208. (a) \( \cos(\theta_k) \) moves towards zero with each iteration, although it is not always non-increasing. (b) Control energy is always non-increasing with each iteration, moving in the direction of quickest decreasing energy cost. Insets show the initial \( 0.5 \times 10^5 \) iterations of the iterative search.

specified, we set \( \{a_{ij}\} = 1 \) if there is a directed link from node j to node i. However, if this results in an infeasible condition number, then we will set the link strength to be random uniform \([0.5, 1.5]\), which tends to make the condition number feasible.
The problem of optimizing target node set in the interest of control energy could be applicable to a social network system, where the state vector represents opinions of individuals, and the driver nodes are modelled as agents of influence. In such a system resembling a voter model\textsuperscript{39}, we may be interested in influencing a certain fractional share of opinions to align with ours. Since we are only interested in obtaining the majority share of opinions, the specificity of which individuals to target control is not so much important as the control energy, which we would like to minimise. Within the framework of network controllability, recent works which incorporate conformity behaviour\textsuperscript{40,41}, where individuals’ opinions are the average of their neighbours’, are promising areas of research to model the network controllability framework as a voter model problem.

Methods

Similar to refs.\textsuperscript{18,19}, the model networks considered in this paper are modelled with stable dynamics, that is, \( \{a_{ii}\} \) is numerically negative. Specifically, we choose the diagonal of the connection matrix \( A \) to be chosen random uniformly from \([-1,1]\], and then stabilise the nodal dynamics with \( \{a_{ii}\} = \delta_i + \varepsilon \), where \( \varepsilon \) is chosen such that the eigenvalues of \( A \) are all negative and the largest eigenvalue is -1. The scale-free model network is constructed from the static model\textsuperscript{42,43}, and the link weights \( \{a_{ij}\} \) are random uniform \([0,1.5]\).

When considering nodes degree in the degree-based selection strategies, we computed the weighted link weights of each node to determine the in-degree (out-degree). For example, the in-degree of node \( i \) is computed as: \( \sum_{j=1, j \neq i}^{N} a_{ij} \). Note that self-links, \( \{a_{ii}\} \), do not count as node degree as they arise from categorically different sources\textsuperscript{12}.

The controllability Gramian can be efficiently calculated using the method of ref.\textsuperscript{38}:

\[
\exp \left( \begin{bmatrix} -A & BB^T \\ 0 & A^T \end{bmatrix} t_f \right) = \begin{bmatrix} F_2(t_f) & G_2(t_f) \\ 0 & F_3(t_f) \end{bmatrix}
\]

\[W = F_3(t_f)^T G_2(t_f)\] (12)

Where each block partitioned matrix of (11) is a \( N \times N \) matrix. The controllability Gramian is computed using (12).

References

1. R. Albert and A.-L. Barabási, “Statistical mechanics of complex networks,” Reviews of modern physics, vol. 74, no. 1, p. 47, 2002.
2. H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai, and A.-L. Barabási, “The large-scale organization of metabolic networks,” Nature, vol. 407, no. 6804, p. 651, 2000.
3. R. Albert, H. Jeong, and A.-L. Barabási, “Error and attack tolerance of complex networks,” nature, vol. 406, no. 6794, p. 378, 2000.
4. J. Ruths and D. Ruths, “Control profiles of complex networks,” Science, vol. 343, no. 6177, pp. 1373–1376, 2014.
5. B. Barzel and A.-L. Barabási, “Universality in network dynamics,” Nature physics, vol. 9, no. 10, p. 673, 2013.
6. Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, “Controllability of complex networks,” Nature, vol. 473, no. 7346, p. 167, 2011.
7. J. E. Hopcroft and R. M. Karp, “An n5/2 algorithm for maximum matchings in bipartite graphs,” SIAM Journal on computing, vol. 2, no. 4, pp. 225–231, 1973.
8. C.-T. Lin, “Structural controllability,” IEEE Transactions on Automatic Control, vol. 19, no. 3, pp. 201–208, 1974.
9. W. J. Rugh and W. J. Rugh, Linear system theory. prentice hall Upper Saddle River, NJ, 1996, vol. 2.
10. D. E. Kirk, Optimal control theory: an introduction. Courier Corporation, 2012.
11. J. Sun and A. E. Motter, “Controllability transition and nonlocality in network control,” Physical review letters, vol. 110, no. 20, p. 208701, 2013.
12. N. J. Cowan, E. J. Chastain, D. A. Vilhena, J. S. Freudenberg, and C. T. Bergstrom, “Nodal dynamics, not degree distributions, determine the structural controllability of complex networks,” PloS one, vol. 7, no. 6, p. e38398, 2012.
13. I. Klickstein, A. Shirin, and F. Sorrentino, “Locally optimal control of complex networks,” Physical review letters, vol. 119, no. 26, p. 268301, 2017.
14. G. Yan, J. Ren, Y.-C. Lai, C.-H. Lai, and B. Li, “Controlling complex networks: How much energy is needed?” *Physical review letters*, vol. 108, no. 21, p. 218703, 2012.

15. J. Ding, C. Wen, and G. Li, “Key node selection in minimum-cost control of complex networks,” *Physica A: Statistical Mechanics and its Applications*, vol. 486, pp. 251–261, 2017.

16. G. Li, J. Ding, C. Wen, and J. Pei, “Optimal control of complex networks based on matrix differentiation,” *EPL (Europhysics Letters)*, vol. 115, no. 6, p. 68005, 2016.

17. G. Li, W. Hu, G. Xiao, L. Deng, P. Tang, J. Pei, and L. Shi, “Minimum-cost control of complex networks,” *New Journal of Physics*, vol. 18, no. 1, p. 013012, 2015.

18. G. Yan, G. Tsekenis, B. Barzel, J.-J. Slotine, Y.-Y. Liu, and A.-L. Barabási, “Spectrum of controlling and observing complex networks,” *Nature Physics*, vol. 11, no. 9, p. 779, 2015.

19. I. Klickstein, A. Shirin, and F. Sorrentino, “Energy scaling of targeted optimal control of complex networks,” *Nature communications*, vol. 8, p. 15145, 2017.

20. G. Lindmark and C. Altafini, “Minimum energy control for complex networks,” *Scientific reports*, vol. 8, no. 1, p. 3188, 2018.

21. Y.-Z. Chen, L.-Z. Wang, W.-X. Wang, and Y.-C. Lai, “Energy scaling and reduction in controlling complex networks,” *Royal Society open science*, vol. 3, no. 4, p. 160064, 2016.

22. G. Li, P. Tang, Z. Meng, C. Wen, J. Pei, and L. Shi, “Optimization on matrix manifold based on gradient information and its applications in network control,” *Physica A: Statistical Mechanics and its Applications*, 2018.

23. J. A. Stankovic, M. Spuri, K. Ramamritham, and G. C. Buttazzo, *Deadline scheduling for real-time systems: EDF and related algorithms*. Springer Science & Business Media, 2012, vol. 460.

24. J. Gao, Y.-Y. Liu, R. M. D’soouza, and A.-L. Barabási, “Target control of complex networks,” *Nature communications*, vol. 5, p. 5415, 2014.

25. X. Liu, L. Pan, H. E. Stanley, and J. Gao, “Controllability of giant connected components in a directed network,” *Physical Review E*, vol. 95, no. 4, p. 042318, 2017.

26. L. Gao, G. Zhao, G. Li, L. Deng, and F. Zeng, “Towards the minimum-cost control of target nodes in directed networks with linear dynamics,” *Journal of the Franklin Institute*, vol. 355, no. 16, pp. 8141–8157, 2018.

27. R. Ahmazadeh, “Stabilized Gram-Schmidt Orthogonal Method, MATLAB Central File Exchange,” (https://www.mathworks.com/matlabcentral/fileexchange/51467), 2016, [Online; accessed Dec 2018].

28. G. Li, L. Deng, G. Xiao, P. Tang, C. Wen, W. Hu, J. Pei, L. Shi, and H. E. Stanley, “Enabling controlling complex networks with local topological information,” *Scientific reports*, vol. 8, no. 1, p. 4593, 2018.

29. L.-Z. Wang, Y.-Z. Chen, W.-X. Wang, and Y.-C. Lai, “Physical controllability of complex networks,” *Scientific reports*, vol. 7, p. 40198, 2017.

30. I. S. Klickstein and F. Sorrentino, “Control distance and energy scaling of complex networks,” *IEEE Transactions on Network Science and Engineering*, 2018.

31. R. Milo, S. Itzkovitz, N. Kashtan, R. Levitt, S. Shen-Orr, I. Ayzenshtat, M. Sheffer, and U. Alon, “Superfamilies of evolved and designed networks,” *Science*, vol. 303, no. 5663, pp. 1538–1542, 2004.

32. D. Baird, J. Luczkovich, and R. R. Christian, “Assessment of spatial and temporal variability in ecosystem attributes of the st marks national wildlife refuge, apalachee bay, florida,” *Estuarine, Coastal and Shelf Science*, vol. 47, no. 3, pp. 329–349, 1998.

33. J. Almunia, G. Basterretxea, J. Aristegui, and R. Ulanowicz, “Benthic-pelagic switching in a coastal subtropical lagoon,” *Estuarine, Coastal and Shelf Science*, vol. 49, no. 3, pp. 363–384, 1999.

34. D. Correll, “Water exchanges in the rhode river watershed, anne arundel county, maryland usa,” (Unpublished manuscript) Smithsonian Institute, Chesapeake Bay Center for Environmental Research, Edgewater, Maryland 21037-0028 USA.

35. R. S. Burt, “Social contagion and innovation: Cohesion versus structural equivalence,” *American journal of Sociology*, vol. 92, no. 6, pp. 1287–1335, 1987.

36. D. R. White and K. P. Reitz, “Rethinking the role concept: Homomorphisms on social networks,” *Research methods in social network analysis*, pp. 429–488, 1989.

37. J. S. Coleman *et al.*, “Introduction to mathematical sociology.” *Introduction to mathematical sociology.*, 1964.
38. C. Van Loan, “Computing integrals involving the matrix exponential,” *IEEE transactions on automatic control*, vol. 23, no. 3, pp. 395–404, 1978.

39. N. Masuda, “Opinion control in complex networks,” *New Journal of Physics*, vol. 17, no. 3, p. 033031, 2015.

40. X.-W. Wang, S. Nie, W.-X. Wang, and B.-H. Wang, “Controlling complex networks with conformity behavior,” *EPL (Europhysics Letters)*, vol. 111, no. 6, p. 68004, 2015.

41. S. Nie, H. E. Stanley, S.-M. Chen, B.-H. Wang, and X.-W. Wang, “Control energy of complex networks towards distinct mixture states,” *Scientific reports*, vol. 8, no. 1, p. 10866, 2018.

42. M. Catanzaro and R. Pastor-Satorras, “Analytic solution of a static scale-free network model,” *The European Physical Journal B-Condensed Matter and Complex Systems*, vol. 44, no. 2, pp. 241–248, 2005.

43. K.-I. Goh, B. Kahng, and D. Kim, “Universal behavior of load distribution in scale-free networks,” *Physical Review Letters*, vol. 87, no. 27, p. 278701, 2001.

Acknowledgement

We would like to thank Tan Yew Lee for discussion on computation speed and bottlenecks and Dr. Lim Yi Xian for guiding us to create high resolution images. Research for this paper was made possible by a Start-Up Grant No. M4081583 from Nanyang Technological University, Singapore.

Author Contribution

H.C. conceived and designed the project, and performed the numerical computations. Both authors wrote the manuscript. E.H.Y. supervised.

Additional Information

Supplementary information accompanies this paper at [to be placed]

Competing Interests: The authors declare that they have no competing interests.