Topology identification and optimal design of noisy consensus networks

Sepideh Hassan-Moghaddam and Mihailo R. Jovanović

***Abstract*— We study an optimal control problem aimed at achieving a desired tradeoff between the network coherence and communication requirements in the distributed controller. Our objective is to add a certain number of edges to an undirected network, with a known graph Laplacian, in order to optimally enhance closed-loop performance. To promote controller sparsity, we introduce ℓ_1-regularization into the optimal $\mathcal{H}_2$ formulation and cast the design problem as a semidefinite program. We derive a Lagrange dual and exploit structure of the optimality conditions for undirected networks to develop three customized algorithms that are well-suited for large problems. These are based on the infeasible primal-dual interior-point, the proximal gradient, and the proximal Newton methods. We illustrate that all of our algorithms significantly outperform the general-purpose solvers and that the proximal methods can solve the problems with more than million edges in the controller graph in a few minutes, on a PC. We also exploit structure of connected resistive networks to demonstrate how additional edges can be systematically added in order to minimize the $\mathcal{H}_2$ norm of the closed-loop system.**

***Index Terms*— Convex optimization, coordinate descent, effective resistance, interior-point method, ℓ_1-regularization, network coherence, preconditioned conjugate gradients, proximal gradient and Newton methods, semidefinite programming, sparsity-promoting optimal control, stochastically-forced networks.**

I. Introduction

Conventional optimal control of distributed systems relies on centralized implementation of control policies. In large networks of dynamical systems, centralized information processing imposes a heavy burden on individual nodes and is often infeasible. This motivates the development of distributed control strategies that require limited information exchange between the nodes to reach consensus or guarantee synchronization. Over the last decade, a vast body of literature has dealt with analysis, fundamental performance limitations, and design of distributed averaging protocols; e.g., see [1]–[12].

Optimal design of the edge weights for networks with prespecified topology has received significant attention. In [2], the design of the fastest averaging protocol for undirected networks was cast as a semidefinite program (SDP). Two customized algorithms, based on primal barrier interior-point (IP) and subgradient methods, were developed and the advantages of optimal weight selection over commonly used heuristics were demonstrated. Similar SDP characterization, for networks with state-dependent graph Laplacians, was provided in [3]. The allocation of symmetric edge weights that minimize the mean-square deviation from average for networks with additive stochastic disturbances was solved in [5]. A related problem, aimed at minimizing the total effective resistance of resistive networks, was addressed in [7]. In [9], the edge Laplacian was used to provide graph-theoretic characterization of the $\mathcal{H}_2$ and $\mathcal{H}_\infty$ symmetric agreement protocols.

Network coherence quantifies the ability of distributed estimation and control strategies to guard against exogenous disturbances [6], [10]. The coherence is determined by the sum of reciprocals of the non-zero eigenvalues of the graph Laplacian and its scaling properties cannot be predicted by algebraic connectivity of the network. In [10], fundamental performance limitations of spatially-localized consensus protocols were examined. Since *these are dictated by the network topology rather than by the optimal selection of the edge weights*, design of optimal topology represents an important challenge in network science. It is precisely this problem that we address in the paper.

More specifically, we study an optimal control problem aimed at achieving a *desired tradeoff* between the *network performance* and *communication requirements* in the distributed controller. Our goal is to add a certain number of edges to a given undirected network in order to optimally enhance the closed-loop performance. One of our key contributions is the formulation of topology design as an optimal control problem that admits convex characterization and is amenable to the development of efficient optimization algorithms. In our formulation, the plant network can contain disconnected components and optimal topology of the controller network is an integral part of the design. In general, this problem amounts to an intractable combinatorial search. Several references have examined convex relaxations or greedy algorithms to identify topology that optimizes algebraic connectivity [13], [14] or coherence [15]–[19] of the network.

We tap on recent developments regarding sparse representations in conjunction with regularization penalties on the level of communication in a distributed controller. This allows us to formulate convex optimization problems that exploit the underlying structure and are amenable to the development of efficient optimization algorithms. To avoid combinatorial complexity, we approach optimal topology design using a recently introduced sparsity-promoting optimal control framework [20], [21]. Performance is captured by the $\mathcal{H}_2$ norm of the closed-loop network and ℓ_1-regularization is introduced to promote controller sparsity. While this problem is in general nonconvex [21], for undirected networks we show that it admits a convex characterization with a non-differentiable objective function and a positive definite constraint. This problem can be transformed into an SDP with linear equality and inequality constraints. For small size networks, the optimal solution can be computed using standard IP method solvers [22], [23].

To enable design of large networks, we pay particular attention to the computational aspects of solving the ℓ_1-regularized $\mathcal{H}_2$ problem. We derive a Lagrange dual of the optimal control

Financial support from the 3M Graduate Fellowship, the UMN Informatics Institute Transdisciplinary Faculty Fellowship, and the National Science Foundation under award ECCS-1407958 is gratefully acknowledged.

Sepideh Hassan-Moghaddam and Mihailo R. Jovanović are with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455. E-mails: hassa247@umn.edu, mihailo@umn.edu.

problem, provide interpretation of dual variables, and develop three efficient customized algorithms. Furthermore, building on preliminary work [24], we specialize our algorithm to the problem of growing connected resistive networks [7], [13]. In these, the plant graph is connected and inequality constraints amount to non-negativity of controller edge weights. This allows us to simplify optimality conditions and further improve computational efficiency of our customized algorithms.

Specialized algorithms, based on inexact IP methods, for ℓ_1-regularized least-squares, logistic regression, and sparse covariance selection problems were developed in [25]–[27]. Our first customized algorithm utilizes the infeasible primal-dual IP method. We employ an inexact iterative method based on the preconditioned conjugate gradients (PCG) to find an approximate search direction and provide a balance between the rate of convergence and computational efficiency [28].

Proximal gradient algorithms and their accelerated variants have recently found use in distributed optimization, statistics, machine learning, image and signal processing. They can be interpreted as generalization of standard gradient projection to problems with non-smooth and extended real-value objective functions [29], [30]. When the proximal operator is easy to evaluate, these algorithms are simple yet extremely efficient.

For networks that can contain disconnected components and non-positive edge weights, the proximal gradient algorithm iteratively updates the controller graph Laplacian via convenient use of the soft-thresholding operator. This extends the Iterative Shrinkage Thresholding Algorithm (ISTA) [29] to optimal topology design of undirected networks. In contrast to the ℓ_1-regularized least-squares, however, the step-size has to be selected to guarantee positivity of the second smallest eigenvalue of the closed-loop graph Laplacian. We combine the Barzilai-Borwein (BB) step-size initialization [31] with backtracking to achieve this goal and enhance the rate of convergence. The biggest computational challenge comes from evaluation of the objective function and its gradient. We exploit problem structure to speed up computations and save memory. Finally, for the problem of growing connected resistive networks, the proximal algorithm simplifies to gradient projection which additionally improves the computational efficiency.

We also develop a customized algorithm based on the proximal Newton method [32]. In contrast to the proximal gradient, this method sequentially employs the second-order Taylor series approximation of the smooth part of the objective function. We use cyclic coordinate descent over the set of active variables to efficiently compute the Newton direction by consecutive minimization with respect to individual coordinates [33]–[35]. Similar approach has been recently utilized in a number of applications, including sparse inverse covariance estimation in graphical models [36].

We illustrate that all of our customized algorithms significantly outperform the general-purpose solvers. The primal-dual IP method based on PCG (with a simple diagonal preconditioner) can solve the problems with hundreds of thousands of edges in the controller graph in several hours, on a PC. Moreover, the algorithms based on proximal gradient and Newton methods can solve the problems with millions of edges in several minutes, on a PC, and are considerably faster than the greedy algorithm with efficient rank-one updates [19].

Our presentation is organized as follows. In Section II, we formulate the problem of optimal topology design for undirected networks subject to additive stochastic disturbances. In Section III, we derive a Lagrange dual of the sparsity-promoting optimal control problem, provide interpretation of dual variables, and construct dual feasible variables from the primal ones. In Section IV, we develop customized algorithms based on the infeasible primal-dual IP method as well as the proximal gradient and Newton methods. In Section V, we achieve additional speedup by specializing our algorithms to the problem of growing connected resistive networks. In Section VI, we use computational experiments to design optimal topology of a controller graph for benchmark problems. We also demonstrate efficiency of our algorithms and their advantage over general-purpose solvers. In Section VII, we provide a brief overview of the paper and highlight future directions.

II. Problem formulation

We consider undirected consensus networks with n nodes

$$\begin{array}{rcl}\dot{\psi} & = & -L_p\,\psi \,+\, u \,+\, d \\ \zeta & = & \left[\begin{array}{c} Q^{1/2} \\ 0 \end{array}\right]\psi \,+\, \left[\begin{array}{c} 0 \\ R^{1/2} \end{array}\right] u \\ u & = & -L_x\,\psi\end{array}$$

where d and ζ denote the disturbance input and performance output, ψ is the state of the network, and u is the control input. Symmetric $n \times n$ matrices L_p and L_x represent Laplacians of the plant and the controller, while $Q = Q^T \succeq 0$ and $R = R^T \succ 0$ are the state and control weights in the standard quadratic performance index. Upon closing the loop we obtain

$$\begin{array}{rcl}\dot{\psi} & = & -\,(L_p \,+\, L_x)\,\psi \,+\, d \\ \zeta & = & \left[\begin{array}{c} Q^{1/2} \\ -R^{1/2}L_x \end{array}\right]\psi.\end{array} \qquad (1)$$

Our objective is to design the optimal topology for L_x and to choose the corresponding edge weights x in order to achieve the desired tradeoff between controller sparsity and network performance. The performance is quantified by the steady-state variance amplification of the stochastically-forced network (from the white-in-time input d to the performance output ζ which penalizes deviation from consensus and control effort).

The interesting features of this problem come from structural restrictions on the matrices L_p, L_x, and Q. All of them are symmetric and are restricted to having an eigenvalue at zero with the corresponding eigenvector of all ones,

$$L_p\,\mathbb{1} \,=\, 0, \quad L_x\,\mathbb{1} \,=\, 0, \quad Q\,\mathbb{1} \,=\, 0. \qquad (2)$$

To guarantee observability of the remaining eigenvalues of L_p, we consider state weights that are positive definite on the orthogonal complement of the subspace spanned by the vector of all ones, $Q + (1/n)\,\mathbb{1}\mathbb{1}^T \succ 0$; e.g., $Q = I - (1/n)\,\mathbb{1}\mathbb{1}^T$ penalizes mean-square deviation from the network average.

In what follows, we express L_x as

$$L_x \;:=\; \sum_{l\,=\,1}^{m} x_l\,\xi_l\,\xi_l^T \;=\; E\,\mathrm{diag}\,(x)\,E^T \qquad (3)$$

where E is the incidence matrix of the controller graph L_x, m is the number of edges in L_x, and $\operatorname{diag}(x)$ is a diagonal matrix containing the vector of the edge weights $x \in \mathbb{R}^m$. Vectors $\xi_l \in \mathbb{R}^n$ determine the columns of E and they signify the connection with weight x_l between nodes i and j: the ith and jth entries of ξ_l are 1 and -1 and all other entries are equal to 0. Thus, L_x given by (3) satisfies structural requirements on the controller graph Laplacian in (2) by construction.

To achieve consensus in the absence of disturbances, the closed-loop network has to be connected [1]. Equivalently, the second smallest eigenvalue of the closed-loop graph Laplacian, $L := L_p + L_x$, has to be positive, i.e., L has to be positive definite on $\mathbb{1}^\perp$. This amounts to positive definiteness of the "strengthened" graph Laplacian of the closed-loop network

$$\begin{aligned} G \;&:=\; L_p \,+\, L_x \,+\, (1/n)\,\mathbb{1}\mathbb{1}^T \\ &=\; G_p \,+\, E\operatorname{diag}(x)\,E^T \,\succ\, 0 \end{aligned} \tag{4a}$$

where

$$G_p \;:=\; L_p \,+\, (1/n)\,\mathbb{1}\mathbb{1}^T. \tag{4b}$$

Structural restrictions (2) on the Laplacian matrices introduce an additional constraint on the matrix G,

$$G\,\mathbb{1} \;=\; \mathbb{1}. \tag{4c}$$

A. Design of optimal sparse topology

Let d be white stochastic disturbance with zero-mean and unit variance,

$$\mathbf{E}\left(d(t)\right) \,=\, 0, \;\; \mathbf{E}\left(d(t_1)\,d^T(t_2)\right) \,=\, I\,\delta(t_1 \,-\, t_2)$$

where $\mathbf{E}$ is the expectation operator. The $\mathcal{H}_2$ norm of the transfer function from d to ζ,

$$\|H\|_2^2 \;=\; \lim_{t\,\to\,\infty} \mathbf{E}\left(\psi^T(t)\,(Q \,+\, L_x\,R\,L_x)\,\psi(t)\right)$$

quantifies the steady-state variance amplification of the closed-loop system (1). The network average, $\bar{\psi}(t) := (1/n)\,\mathbb{1}^T\psi(t)$, corresponds to the zero eigenvalue of the graph Laplacian and it is not observable from the performance output in (1). Thus, the $\mathcal{H}_2$ norm is equivalently given by

$$\begin{aligned} \|H\|_2^2 \;&=\; \lim_{t\,\to\,\infty} \mathbf{E}\left(\tilde{\psi}^T(t)\,(Q \,+\, L_x\,R\,L_x)\,\tilde{\psi}(t)\right) \\ &=\; \operatorname{trace}\left(P\,(Q \,+\, L_x\,R\,L_x)\right) \,=\, \langle P, Q \,+\, L_x\,R\,L_x\rangle \end{aligned}$$

where $\tilde{\psi}(t)$ is the vector of deviations of the states of individual nodes from $\bar{\psi}(t)$,

$$\tilde{\psi}(t) \;:=\; \psi(t) \,-\, \mathbb{1}\,\bar{\psi}(t) \;=\; \left(I \,-\, (1/n)\,\mathbb{1}\mathbb{1}^T\right)\psi(t)$$

and P is the steady-state covariance matrix of $\tilde{\psi}$,

$$P \;:=\; \lim_{t\,\to\,\infty} \mathbf{E}\left(\tilde{\psi}(t)\,\tilde{\psi}^T(t)\right).$$

The above measure of the amplification of stochastic disturbances is determined by $\|H\|_2^2 = (1/2)J(x)$, where

$$J(x) \;:=\; \left\langle \left(G_p \,+\, E\operatorname{diag}(x)\,E^T\right)^{-1}, Q + L_x\,R\,L_x \right\rangle. \tag{5}$$

It can be shown that J can be expressed as

$$\begin{aligned} J(x) \;=\; &\left\langle \left(G_p \,+\, E\operatorname{diag}(x)\,E^T\right)^{-1}, Q_p \right\rangle \,+ \\ &\operatorname{diag}\left(E^T R\,E\right)^T x \,-\, \langle R, L_p\rangle \,-\, 1 \end{aligned} \tag{6}$$

with

$$Q_p \;:=\; Q \,+\, (1/n)\,\mathbb{1}\mathbb{1}^T \,+\, L_p\,R\,L_p.$$

Note that the last two terms in (6) do not depend on the optimization variable x and that the term $L_p\,R\,L_p$ in Q_p has an interesting interpretation: it determines a state-weight that guarantees inverse optimality (in LQR sense) of $u = -L_p\psi$ for a system with no coupling between the nodes, $\dot{\psi} = u + d$.

We formulate the design of a controller graph that provides an optimal tradeoff between the $\mathcal{H}_2$ performance of the closed-loop network and the controller sparsity as

$$\begin{array}{ll} \underset{x}{\text{minimize}} & J(x) \,+\, \gamma\,\|x\|_1 \\ \text{subject to} & G_p \,+\, E\operatorname{diag}(x)\,E^T \,\succ\, 0 \end{array} \tag{SP}$$

where $J(x)$ and G_p are given by (6) and (4b), respectively. The ℓ_1 norm of x, $\|x\|_1 := \sum_{l=1}^m |x_l|$, is introduced as a proxy for promoting sparsity [37]–[40]. In (SP), the vector of the edge weights $x \in \mathbb{R}^m$ is optimization variable; the problem data is given by the positive regularization parameter γ, the state and control weights Q and R, the plant graph Laplacian L_p, and the incidence matrix of the controller graph E.

The sparsity-promoting optimal control problem (SP) is a constrained optimization problem with a convex non-differentiable objective function [15] and a positive definite inequality constraint. This implies convexity of (SP). Positive definiteness of the strengthened graph Laplacian, $G = G_p + E\operatorname{diag}(x)\,E^T$, guarantees stability of the closed-loop network (1) on the subspace $\mathbb{1}^\perp$, and thereby consensus in the absence of disturbances [1].

The consensus can be achieved even if some edge weights are negative [2], [5]. By expressing x as a difference between two non-negative vectors, $x = x_+ - x_-$, (SP) can be written as

$$\begin{array}{ll} \underset{x_+,\,x_-}{\text{minimize}} & \left\langle \left(G_p \,+\, E\operatorname{diag}(x_+ \,-\, x_-)\,E^T\right)^{-1}, Q_p \right\rangle \,+ \\ & (\gamma\,\mathbb{1} \,+\, c)^T x_+ \,+\, (\gamma\,\mathbb{1} \,-\, c)^T x_- \\ \text{subject to} & G_p \,+\, E\operatorname{diag}(x_+ \,-\, x_-)\,E^T \,\succ\, 0 \\ & x_+ \,\geq\, 0, \;\; x_- \,\geq\, 0 \end{array} \tag{7}$$

where $c := \operatorname{diag}\left(E^T R\,E\right)$. By utilizing the Schur complement, (7) can be cast as an SDP, and solved via standard IP method algorithms for small size networks.

Reweighted ℓ_1 norm: An alternative proxy for promoting sparsity is given by the weighted ℓ_1 norm [41],

$$\|w \circ x\|_1 \;:=\; \sum_{l\,=\,1}^{m} w_l\,|x_l|$$

where $\circ$ denotes elementwise (Hadamard) product. The vector of non-negative weights $w \in \mathbb{R}^m$ can be selected to provide better approximation of non-convex cardinality function than the ℓ_1 norm. An effective heuristic for weight selection is given by the iterative reweighted algorithm [41], with w_l inversely proportional to the magnitude of x_l in the previous iteration,

$$w_l^+ \;=\; 1/(|x_l| \,+\, \varepsilon). \tag{8}$$

This puts larger emphasis on smaller optimization variables, where a small positive parameter ε ensures that w_l^+ is welldefined. If the weighted ℓ_1 norm is used in (SP), the vector of all ones $\mathbb{1}$ should be replaced by the vector w in (7).

B. Structured optimal control problem: debiasing step

After the structure of the controller graph Laplacian L_x has been designed, we eliminate the columns from the incidence matrix E that correspond to zero elements in the vector of the optimal edge weights $x^\star$. This yields a new incidence matrix $\hat{E}$ and leads to the optimization problem

$$\begin{array}{ll}\underset{x}{\text{minimize}} & \left\langle \left(G_p \,+\, \hat{E}\,\mathrm{diag}\,(x)\,\hat{E}^T\right)^{-1}, Q_p \right\rangle \,+ \\ & \mathrm{diag}\left(\hat{E}^T R\,\hat{E}\right)^T x \\ \text{subject to} & G_p \,+\, \hat{E}\,\mathrm{diag}\,(x)\,\hat{E}^T \,\succ\, 0\end{array}$$

whose solution provides the optimal controller graph Laplacian with the desired structure. This optimization problem is obtained by setting $\gamma = 0$ in (SP) and by replacing the incidence matrix E with $\hat{E}$. This "polishing" or "debiasing" step is used to improve the performance relative to the solution of the sparsity-promoting optimal control problem (SP).

C. Gradient and Hessian of $J(x)$

We next summarize the first- and second-order derivatives of the objective function J, given by (6), with respect to the vector of the edge weights x. The second-order Taylor series approximation of $J(x)$ around $\bar{x} \in \mathbb{R}^m$ is given by

$$J(\bar{x} + \tilde{x}) \;\approx\; J(\bar{x}) \,+\, \nabla J(\bar{x})^T \tilde{x} \,+\, \frac{1}{2}\,\tilde{x}^T\,\nabla^2 J(\bar{x})\,\tilde{x}.$$

For related developments we refer the reader to [7].

Proposition 1: The gradient and the Hessian of J at $\bar{x} \in \mathbb{R}^m$ are determined by

$$\begin{array}{rcl}\nabla J(\bar{x}) & = & -\,\mathrm{diag}\left(E^T (Y(\bar{x}) \,-\, R)\,E\right) \\ \nabla^2 J(\bar{x}) & = & H_1(\bar{x}) \,\circ\, H_2(\bar{x})\end{array}$$

where

$$\begin{array}{rcl}Y(\bar{x}) & := & \left(G_p + E\,D_{\bar{x}}\,E^T\right)^{-1} Q_p \left(G_p + E\,D_{\bar{x}}\,E^T\right)^{-1} \\ H_1(\bar{x}) & := & E^T\,Y(\bar{x})\,E \\ H_2(\bar{x}) & := & E^T\left(G_p \,+\, E\,D_{\bar{x}}\,E^T\right)^{-1} E \\ D_{\bar{x}} & := & \mathrm{diag}\,(\bar{x})\,.\end{array}$$

III. Dual problem

Herein, we study the Lagrange dual of the sparsity-promoting optimal control problem (7), provide interpretation of dual variables, and construct dual feasible variables from primal feasible variables. Since minimization of the Lagrangian associated with (7) does not lead to an explicit expression for the dual function, we introduce an auxiliary variable G and find the dual of

$$\begin{array}{ll}\underset{G,\,x_\pm}{\text{minimize}} & \left\langle G^{-1}, Q_p\right\rangle \,+\, (\gamma\,\mathbb{1} \,+\, c)^T x_+ \,+\, (\gamma\,\mathbb{1} \,-\, c)^T x_- \\ \text{subject to} & G \,-\, G_p \,-\, E\,\mathrm{diag}\,(x_+ \,-\, x_-)\,E^T \,=\, 0 \\ & G \,\succ\, 0, \;\; x_+ \,\geq\, 0, \;\; x_- \,\geq\, 0.\end{array} \tag{P}$$

In (P), G represents the "strengthened" graph Laplacian of the closed-loop network and the equality constraint comes from (4a). As we show next, the Lagrange dual of the primal optimization problem (P) admits an explicit characterization.

Proposition 2: The Lagrange dual of the primal optimization problem (P) is given by

$$\begin{array}{ll}\underset{Y}{\text{maximize}} & 2\;\mathrm{trace}\left((Q_p^{1/2}\,Y\,Q_p^{1/2})^{1/2}\right) \,-\, \langle Y, G_p\rangle \\ \text{subject to} & \|\,\mathrm{diag}\left(E^T (Y \,-\, R)\,E\right)\|_\infty \,\leq\, \gamma \\ & Y \,\succ\, 0, \;\; Y\,\mathbb{1} \,=\, \mathbb{1}\end{array} \tag{D}$$

where $Y = Y^T \in \mathbb{R}^{n \times n}$ is the dual variable associated with the equality constraint in (P).

Proof: The Lagrangian of (P) is given by

$$\begin{array}{rcl}\mathcal{L} & = & \left\langle G^{-1}, Q_p\right\rangle \,+\, \langle Y, G\rangle \,-\, \langle Y, G_p\rangle \,+ \\ & & \left(\gamma\,\mathbb{1} \,-\, \mathrm{diag}\left(E^T (Y - R)\,E\right) \,-\, y_+\right)^T x_+ \,+ \\ & & \left(\gamma\,\mathbb{1} \,+\, \mathrm{diag}\left(E^T (Y - R)\,E\right) \,-\, y_-\right)^T x_-\end{array} \tag{9}$$

where Y and $y_\pm \geq 0$ are Lagrange multipliers (i.e., dual variables) associated with equality and elementwise inequality constraints in (P). Note that no Lagrange multiplier is assigned to the positive definite constraint on G in $\mathcal{L}$. Instead, we determine conditions on Y and $y_\pm$ that guarantee $G \succ 0$.

Minimizing $\mathcal{L}$ with respect to G yields

$$G^{-1}\,Q_p\,G^{-1} \,=\, Y \tag{10a}$$

or, equivalently,

$$G \,=\, Q_p^{1/2}\left(Q_p^{1/2}\,Y\,Q_p^{1/2}\right)^{-1/2} Q_p^{1/2}. \tag{10b}$$

Positive definiteness of G and Q_p implies $Y \succ 0$. Furthermore, since $Q_p \mathbb{1} = \mathbb{1}$, from (4c) and (10a) we have

$$Y\,\mathbb{1} \,=\, \mathbb{1}.$$

Similarly, minimization with respect to x_+ and x_- leads to

$$y_+ \,=\, \gamma\,\mathbb{1} \,-\, \mathrm{diag}\left(E^T (Y - R)\,E\right) \,\geq\, 0 \tag{11a}$$

$$y_- \,=\, \gamma\,\mathbb{1} \,+\, \mathrm{diag}\left(E^T (Y - R)\,E\right) \,\geq\, 0. \tag{11b}$$

Thus, non-negativity of y_+ and y_- amounts to

$$-\gamma\,\mathbb{1} \,\leq\, \mathrm{diag}\left(E^T (Y - R)\,E\right) \,\leq\, \gamma\,\mathbb{1}$$

or, equivalently,

$$\|\,\mathrm{diag}\left(E^T (Y - R)\,E\right)\|_\infty \,\leq\, \gamma.$$

Substitution of (10) and (11) into (9) eliminates y_+ and y_- from the dual problem. We can thus represent the dual function, $\inf_{G,\,x_\pm} \mathcal{L}(G, x_\pm; Y, y_\pm)$, as

$$2\;\mathrm{trace}\left((Q_p^{1/2}\,Y\,Q_p^{1/2})^{1/2}\right) \,-\, \langle Y, G_p\rangle$$

which allows us to bring the dual of (P) to (D). ■

Any dual feasible Y can be used to obtain a lower bound on the optimal value of the primal problem (P). Furthermore, the difference between the objective functions of the primal (evaluated at the primal feasible $(G, x_\pm)$) and the dual (evaluated at the dual feasible Y) problems yields the duality gap,

$$\eta \,=\, y_+^T\,x_+ \,+\, y_-^T\,x_- \,=\, \mathbb{1}^T\,(y_+ \circ x_+ \,+\, y_- \circ x_-) \tag{12}$$

where y_+ and y_- are given by (11a) and (11b). This positive quantity can be used to estimate distance to optimality.

Strong duality follows from convexity of the primal problem (P) and strict feasibility of the constraints in (P). This implies that at optimality, the duality gap η for the primal problem (P) and the dual problem (D) is zero. Furthermore, if $(G^\star, x^\star_\pm)$ are optimal points of the primal problem (P), then $Y^\star = (G^\star)^{-1} Q_p \, (G^\star)^{-1}$ is the optimal point of the dual problem (D). Similarly, if $Y^\star$ is the optimal point of (D),

$$G^\star \;=\; Q_p^{1/2} \left(Q_p^{1/2} Y^\star Q_p^{1/2} \right)^{-1/2} Q_p^{1/2}$$

is the optimal point of (P). The optimal vector of the edge weights $x^\star$ is determined by the non-zero off-diagonal elements of the controller graph Laplacian, $L_x^\star = G^\star - G_p$.

Interpretation of dual variables: For electrical networks, the dual variables have appealing interpretations. Let $\iota \in \mathbb{R}^n$ be a random current injected into the resistor network satisfying

$$\mathbb{1}^T \iota \;=\; 0, \quad \mathbf{E}\,(\iota) \;=\; 0, \quad \mathbf{E}\left(\iota \iota^T\right) \;=\; Q \;+\; L_p \, R \, L_p.$$

The vector of voltages $\vartheta \in \mathbb{R}^m$ across the edges of the network is then given by $\vartheta = E^T G^{-1} \iota$. Furthermore, since

$$\mathbf{E}\left(\vartheta \, \vartheta^T\right) \;=\; E^T \, G^{-1} \, \mathbf{E}\left(\iota \iota^T\right) G^{-1} \, E \;=\; E^T \, Y \, E,$$

the dual variable Y is related to the covariance matrix of voltages across the edges. Moreover, (11) implies that y_+ and y_- quantify the deviations between variances of edge voltages from their respective upper and lower bounds.

Remark 1: For a primal feasible x, Y resulting from (10a) with G given by (4a) may not be dual feasible. Let

$$\hat{Y} \;:=\; \beta \, Y \;+\; \frac{1 - \beta}{n} \, \mathbb{1}\mathbb{1}^T \tag{13a}$$

and let the control weight be $R = r\,I$ with $r > 0$. If

$$\beta \;\leq\; \frac{\gamma \;+\; 2\,r}{\|\operatorname{diag}\left(E^T (Y \,-\, R)\, E\right)\|_\infty \;+\; 2\,r} \tag{13b}$$

then $\hat{Y}$ satisfies the inequality constraint in (D) and it is thus dual feasible.

IV. Customized algorithms

We next exploit the structure of the sparsity-promoting optimal control problem (SP) and develop customized algorithms based on the primal-dual interior-point, the proximal gradient, and the proximal Newton methods. The proximal gradient algorithm is a first-order method that uses a simple quadratic approximation of J in (SP). This yields an explicit update of the vector of the edge weights via application of the soft-thresholding operator. The Newton direction in the IP method is obtained using an inexact iterative PCG-based procedure. This avoids storage of the Hessian and leads to a significant speedup relative to standard IP method solvers. Finally, in the proximal Newton method a sequential quadratic approximation of the smooth part of the objective function in (SP) is used and the search direction is efficiently computed via cyclic coordinate descent over the set of active variables.

A. Primal-dual interior-point method

Although the problem is SDP representable, we do not use standard SDP characterization in our algorithmic developments. In fact, we treat our problem as a convex program and provide implementation that is based on a modification of a primal-dual IP algorithm for linear programming to our setup. While the major algorithmic component, namely infeasible primal-dual interior-point method, is well-known, our use of this standard method is novel and it exploits specific structure of the optimality conditions for undirected networks.

1) Central path equations and search direction: To develop a customized algorithm, we introduce $D_x := \operatorname{diag}(x)$ and combine (10a) and (4a) to express Y in terms of x,

$$Y(x) \;=\; \left(G_p + E\,D_x\,E^T\right)^{-1} Q_p \left(G_p + E\,D_x\,E^T\right)^{-1}. \tag{14}$$

This facilitates the use of an infeasible primal-dual interior-point method to solve the central path equations

$$\begin{aligned}
\gamma\,\mathbb{1} \;-\; \operatorname{diag}\left(E^T (Y(x) - R)\,E\right) \;-\; y_+ \;&=\; 0 &\text{(15a)}\\
\gamma\,\mathbb{1} \;+\; \operatorname{diag}\left(E^T (Y(x) - R)\,E\right) \;-\; y_- \;&=\; 0 &\text{(15b)}\\
x \;-\; x_+ \;+\; x_- \;&=\; 0 &\text{(15c)}\\
y_+ \circ x_+ \;&=\; \sigma\,\mu\,\mathbb{1} &\text{(15d)}\\
y_- \circ x_- \;&=\; \sigma\,\mu\,\mathbb{1} &\text{(15e)}
\end{aligned}$$

for x, $x_\pm$, and $y_\pm$. Equations (15d) and (15e) are obtained by relaxing complementary slackness conditions $y_\pm \circ x_\pm = 0$. Here, μ and σ are positive parameters that provide continuous deformation of the optimality conditions; μ quantifies the current duality gap and the centering parameter $\sigma \leq 1$ determines the desired duality gap reduction in the current iteration [42], [43]. Finally, we enforce the inequality constraints

$$G_p \;+\; E\operatorname{diag}(x)\,E^T \;\succ\; 0, \quad x_\pm \;\geq\; 0, \quad y_\pm \;\geq\; 0 \tag{16}$$

via an appropriate step-size selection.

We assume that equations (15a), (15b), and (15c) can be violated. For $(\bar{x}, \bar{x}_\pm, \bar{y}_\pm)$ that satisfy (16) but are infeasible (i.e., do not satisfy (15a), (15b), and (15c) with $\bar{Y} := Y(\bar{x})$ given by (14)), the primal and dual residuals are

$$\begin{aligned}
r_p(\bar{x},\, \bar{x}_\pm) \;&:=\; \bar{x} \;-\; \bar{x}_+ \;+\; \bar{x}_-\\
r_d^+(\bar{x},\, \bar{y}_+) \;&:=\; \gamma\,\mathbb{1} \;-\; \operatorname{diag}\left(E^T (\bar{Y} - R)\,E\right) \;-\; \bar{y}_+ \qquad (17)\\
r_d^-(\bar{x},\, \bar{y}_-) \;&:=\; \gamma\,\mathbb{1} \;+\; \operatorname{diag}\left(E^T (\bar{Y} - R)\,E\right) \;-\; \bar{y}_-.
\end{aligned}$$

By linearizing (15) around $(\bar{x}, \bar{x}_\pm, \bar{y}_\pm)$ and eliminating $\tilde{x}$ from the resulting equations, we obtain the search direction $(\tilde{x}_\pm, \tilde{y}_\pm)$ as the solution to

$$\begin{aligned}
A \left[\begin{array}{c} \tilde{x}_+ \\ \tilde{x}_- \end{array}\right] \;&=\; b\\
\tilde{y}_\pm \;&=\; \sigma\,\mu\,D_{\bar{x}_\pm}^{-1}\,\mathbb{1} \;-\; D_\pm\,\tilde{x}_\pm \;-\; \bar{y}_\pm.
\end{aligned} \tag{18}$$

Here,

$$\begin{aligned} A &= \begin{bmatrix} H + D_+ & -H \\ -H & H + D_- \end{bmatrix}, \quad D_\pm = D_{\bar{x}_\pm}^{-1} D_{\bar{y}_\pm} \\ b &= \begin{bmatrix} \sigma\,\mu\, D_{\bar{x}_+}^{-1}\mathbb{1} + H\, r_p - r_d^+ - \bar{y}_+ \\ \sigma\,\mu\, D_{\bar{x}_-}^{-1}\mathbb{1} - H\, r_p - r_d^- - \bar{y}_- \end{bmatrix} \\ H(\bar{x}) &= 2\,(H_1(\bar{x}) \circ H_2(\bar{x})) \end{aligned}$$

and the matrices $H_1(\bar{x})$ and $H_2(\bar{x})$ are given in Proposition 1.

We note that the matrix A in (18) is positive definite. Clearly, we can write A as $A = A_1 + A_2$ with

$$A_1 = \begin{bmatrix} D_+ & 0 \\ 0 & D_- \end{bmatrix}, \quad A_2 = \begin{bmatrix} H & -H \\ -H & H \end{bmatrix}.$$

The matrix A_1 is block-diagonal with $D_\pm \succ 0$ and thus positive definite. Since H is an elementwise product of two positive definite matrices H_1 and H_2, it is positive definite. The Schur complement lemma implies positive-semi definiteness of A_2 and, thus, positive definiteness of A.

2) Algorithm: As typically done in IP methods, our customized algorithm is enhanced via Mehrotra's predictor-corrector step [44]. The predictor step yields affine scaling direction $(\tilde{x}^a, \tilde{x}^a_\pm, \tilde{y}^a_\pm)$, and the corrector step is used to obtain the search direction $(\tilde{x}, \tilde{x}_\pm, \tilde{y}_\pm)$ [42, Section 14.2].

The challenging aspect of the primal-dual IP algorithm is the computation of the search directions $(\tilde{x}^a, \tilde{x}^a_\pm, \tilde{y}^a_\pm)$ and $(\tilde{x}, \tilde{x}_\pm, \tilde{y}_\pm)$. These are obtained by solving linear systems of equations for $(\tilde{x}^a_+, \tilde{x}^a_-)$ and $(\tilde{x}_+, \tilde{x}_-)$. Vectors $\tilde{y}^a_\pm$, $\tilde{y}_\pm$, and $\tilde{x}$ are then computed using (18). The use of Mehrotra's predictor-corrector step implies that the same matrix A appears in the equations for $(\tilde{x}^a_+, \tilde{x}^a_-)$ and $(\tilde{x}_+, \tilde{x}_-)$. Thus, for moderately sized problems, Cholesky factorization of A followed by back solve operations can be used to to determine the search directions. These respectively take $O(m^3)$ and $O(m^2)$ operations.

3) Search direction via the PCG method: Since Cholesky factorization is not well-suited for large problems, we next provide an efficient inexact method for computing the search directions $(\tilde{x}^a, \tilde{x}^a_\pm, \tilde{y}^a_\pm)$ and $(\tilde{x}, \tilde{x}_\pm, \tilde{y}_\pm)$. Our approach utilizes the PCG algorithm, which is an indirect iterative method for solving a linear system of equations with a positive definite matrix A [42], [45].

In exact arithmetics, conjugate gradients algorithm converges in m iterations. Each iteration requires a few inner products and one matrix-vector multiplication [42, Algorithm 5.3]. If the matrix A is dense, matrix-vector multiplication costs $O(m^2)$. Thus, the total cost is $O(m^3)$ which is of the same order as the direct method based on Cholesky factorization.

A computational advantage can be gained if matrix-vector multiplication is cheaper than $O(m^2)$; e.g., if the matrix A is sparse [45]. Moreover, an acceptable solution can often be reached in less than m iterations. On the other hand, conjugate gradients method can perform poorly for ill-conditioned matrix A and in the presence of round-off errors. In many problems, introduction of suitably selected preconditioners is essential to ensure fast convergence or even convergence.

In our implementation, we use the following preconditioner

$$\Pi := \begin{bmatrix} \operatorname{diag}(H) + D_+ & -\operatorname{diag}(H) \\ -\operatorname{diag}(H) & \operatorname{diag}(H) + D_- \end{bmatrix}.$$

Similar preconditioner was used to solve ℓ_1-regularized least-squares problem via inexact IP method [25]. The action of Π^{-1} on the vector $p := \begin{bmatrix} p_1^T & p_2^T \end{bmatrix}^T$ is given by

$$\Pi^{-1} p = \begin{bmatrix} D_3\, p_1 + D_3\, D_2\, p_2 \\ D_3\, D_2\, p_1 + (D_1 + D_2\, D_3\, D_2)\, p_2 \end{bmatrix} \tag{19a}$$

where

$$\begin{aligned} D_1 &:= (\operatorname{diag}(H) + D_-)^{-1} \\ D_2 &:= \operatorname{diag}(H)\, D_1 \\ D_3 &:= (\operatorname{diag}(H)\,(I - D_2) + D_+)^{-1}. \end{aligned} \tag{19b}$$

Since all matrices in (19) are diagonal, no matrix inversion is required to compute $\Pi^{-1}p$, and the result can be obtained via elementwise vector multiplication.

Another desirable feature of the PCG method is that we do not need to store the matrix A; computation of only matrix-vector products is required. This offers significant memory saving in large-scale problems. The action of the matrix H, which is determined by the Hadamard product of H_1 and H_2, on the vector p_i is determined by

$$(H_1 \circ H_2)\, p_i = \operatorname{diag}(H_1 \operatorname{diag}(p_i)\, H_2).$$

From the definitions of the matrices H_1 and H_2 (see Proposition 1), it follows that

$$(H_1 \circ H_2)\, p_i = \operatorname{diag}(E^T M\, E)$$

where we cache the $n \times n$ matrix M,

$$M := Y\, E \operatorname{diag}(p_i)\, E^T \big(G_p + E \operatorname{diag}(\bar{x})\, E^T\big)^{-1}.$$

Efficiency of iterative algorithms can be further improved with a better choice of preconditioner. It is an open issue how to systematically obtain these. We note that the performance of a preconditioner based on incomplete Cholesky factorization was inferior compared to the preconditioner that we used.

In our implementation, the initial values of both the affine scaling direction $(\tilde{x}^a_+, \tilde{x}^a_-)$ and the search direction $(\tilde{x}_+, \tilde{x}_-)$ are set to zero. The PCG algorithm is terminated if either the number of PCG iterations exceeds the desired value N_{pcg} or if the following stopping criterion is satisfied

$$\|A \begin{bmatrix} \tilde{x}_+ \\ \tilde{x}_- \end{bmatrix} - b\|_2 \le \epsilon_{\mathrm{pcg}}$$

Here, $\epsilon_{\mathrm{pcg}} = \min\{0.1,\, \delta\,\eta/\|b\|_2\}$, η is the current value of the duality gap, and δ is an algorithm parameter. In early iterations, the Newton system is solved with low accuracy that does not deteriorate beyond 10%. The dependence of ϵ_{pcg} on the duality gap η implies that, as η decreases, the accuracy improves [42]. Our computational experiments indicate that the constant $\delta = 0.3$ in the expression for ϵ_{pcg} appears to work well for a broad range of problems. Similar stopping criteria were used in inexact IP method solvers for ℓ_1-regularized least-squares and logistic regression problems [25], [26].

B. Proximal gradient method

We next use the proximal gradient method to solve (SP). A simple quadratic approximation of $J(x)$ around the current iterate x^k,

$$J(x) \;\approx\; J(x^k) \,+\, \nabla J(x^k)^T (x \,-\, x^k) \,+\, \frac{1}{2\alpha_k}\, \| x \,-\, x^k \|_2^2$$

is substituted to (SP) to obtain

$$x^{k+1} \;=\; \underset{x}{\operatorname{argmin}} \;\, g(x) \,+\, \frac{1}{2\alpha_k}\, \| x \,-\, (x^k \,-\, \alpha_k \nabla J(x^k)) \|_2^2.$$

Here, α_k is the step-size and the update is determined by the proximal operator of the function $\alpha_k\, g$ [30],

$$x^{k+1} \;=\; \mathbf{prox}_{\alpha_k g} \big(x^k \,-\, \alpha_k \nabla J(x^k) \big) \,.$$

In particular, for $g(x) = \gamma \, \| x \|_1$, we have

$$x^{k+1} \;=\; \mathcal{S}_{\gamma \alpha_k} \big(x^k \,-\, \alpha_k \nabla J(x^k) \big)$$

where $\mathcal{S}_\kappa(y) \;=\; \operatorname{sign}(y) \max(|y| - \kappa, 0)$ is the soft-thresholding function.

The proximal gradient algorithm converges with rate $O(1/k)$ if $\alpha_k < 1/L$, where L is the Lipschitz constant of ∇J [29], [30]. It can be shown that ∇J is Lipschitz continuous but, since it is challenging to explicitly determine L, we adjust α_k via backtracking. To provide a better estimate of L, we initialize α_k using the Barzilai-Borwein (BB) method [31] which provides an effective heuristic for approximating the Hessian of the function J via the scaled version of the identity, $(1/\alpha_k) I$. At the kth iteration, the initial BB step-size $\alpha_{k,0}$,

$$\alpha_{k,0} \;:=\; \frac{\| x^k \,-\, x^{k-1} \|_2^2}{(x^{k-1} \,-\, x^k)^T \, (\nabla J(x^{k-1}) \,-\, \nabla J(x^k))} \qquad (20)$$

is adjusted via backtracking until the inequality constraint in (SP) is satisfied and

$$J(x^{k+1}) \le J(x^k) + \nabla J(x^k)^T (x^{k+1} - x^k) + \frac{1}{2\alpha_k} \| x^{k+1} - x^k \|_2^2.$$

Since J is continuously differentiable with Lipschitz continuous gradient, this inequality holds for any $\alpha_k < 1/L$ and the algorithm converges sub-linearly [29]. This condition guarantees that objective function decreases at every iteration. Our numerical experiments in Section VI suggest that BB step-size initialization significantly enhances the rate of convergence.

Remark 2: The biggest computational challenge comes from evaluation of the objective function and its gradient. Since the inverse of the strengthened graph Laplacian G has to be computed, with direct computations these evaluations take $O(n^3)$ and $O(nm^2)$ flops, respectively. However, by exploiting the problem structure, ∇J can be computed more efficiently. The main cost arises in the computation of $\operatorname{diag}(E^T Y E)$. We instead compute it using $\operatorname{sum}(E^T \circ (Y E))$ which takes $O(n^2 m)$ operations. Here, $\operatorname{sum}(A)$ is a vector which contains summation of each row of the matrix A in its entries. For networks with $m \gg n$ this leads to significant speed up. Moreover, in contrast to direct computation, we do not need to store the $m \times m$ matrix $E^T Y E$. Only formation of the columns is required which offers memory saving.

C. Proximal Newton method

In contrast to the proximal gradient algorithm, the proximal Newton method benefits from second-order Taylor series expansion of the smooth part of the objective function in (SP). Herein, we employ cyclic coordinate descent over the set of active variables to efficiently compute the Newton direction.

By approximating the smooth part of the objective function J in (SP) with the second-order Taylor series expansion around the current iterate $\bar{x}$,

$$J(\bar{x} + \tilde{x}) \;\approx\; J(\bar{x}) \,+\, \nabla J(\bar{x})^T \, \tilde{x} \,+\, \frac{1}{2}\, \tilde{x}^T \, \nabla^2 J(\bar{x}) \, \tilde{x}$$

the problem (SP) becomes

$$\begin{array}{ll} \underset{\tilde{x}}{\text{minimize}} & \nabla J(\bar{x})^T \, \tilde{x} \,+\, \dfrac{1}{2}\, \tilde{x}^T \, \nabla^2 J(\bar{x}) \, \tilde{x} \,+\, \gamma \, \| \bar{x} \,+\, \tilde{x} \|_1 \\ \text{subject to} & G_p \,+\, E \operatorname{diag}(\bar{x} \,+\, \tilde{x}) E^T \,\succ\, 0. \end{array} \qquad (21)$$

Let $\tilde{x}$ denote the current iterate approximating the Newton direction. By perturbing $\tilde{x}$ in the direction of the ith standard basis vector e_i in $\mathbb{R}^m$, the objective function in (21) becomes

$$\nabla J(\bar{x})^T (\tilde{x} \,+\, \delta_i \, \mathrm{e}_i) \,+\, \frac{1}{2}\, (\tilde{x} \,+\, \delta_i \, \mathrm{e}_i)^T \, \nabla^2 J(\bar{x}) \, (\tilde{x} \,+\, \delta_i \, \mathrm{e}_i) \\ \,+\, \gamma \, |\bar{x}_i \,+\, \tilde{x}_i \,+\, \delta_i|.$$

Elimination of constant terms allows us to bring (21) into

$$\underset{\delta_i}{\text{minimize}} \quad \frac{1}{2}\, a_i \, \delta_i^2 \,+\, b_i \, \delta_i \,+\, \gamma \, |c_i \,+\, \delta_i| \qquad (22)$$

where the optimization variable is the scalar δ_i and $(a_i,\, b_i,\, c_i,\, \bar{x}_i,\, \tilde{x}_i)$ are the problem data with

$$\begin{array}{rcl} a_i & := & \mathrm{e}_i^T \, \nabla^2 J(\bar{x}) \, \mathrm{e}_i \\ b_i & := & \big(\nabla^2 J(\bar{x}) \, \mathrm{e}_i \big)^T \tilde{x} \,+\, \mathrm{e}_i^T \, \nabla J(\bar{x}) \\ c_i & := & \bar{x}_i \,+\, \tilde{x}_i. \end{array}$$

The explicit solution to (22) is given by

$$\delta_i \;=\; -\, c_i \,+\, \mathcal{S}_{\gamma / a_i} (c_i \,-\, b_i / a_i) \,.$$

After the Newton direction $\tilde{x}$ has been computed, we determine the step-size α via backtracking. This guarantees positive definiteness of the strengthened graph Laplacian and sufficient decrease of the objective function. We use generalization of Armijo rule [46], [47] to find an appropriate step-size α such that $G_p + E \operatorname{diag}(\bar{x} + \alpha \tilde{x}) E^T$ is positive definite matrix and

$$J(\bar{x} + \alpha \tilde{x}) \,+\, \gamma \, \| \bar{x} + \alpha \tilde{x} \|_1 \,\le\, J(\bar{x}) \,+\, \gamma \, \| \bar{x} \|_1 \,+ \\ \alpha \, \sigma \, \big(\nabla J(\bar{x})^T \tilde{x} \,+\, \gamma \, \| \bar{x} + \tilde{x} \|_1 \,-\, \gamma \, \| \bar{x} \|_1 \big) \,.$$

Remark 3: The parameter a_i in (22) is determined by the ith diagonal element of the Hessian $\nabla^2 J(\bar{x})$. On the other hand, the ith column of $\nabla^2 J(\bar{x})$ and the ith element of the gradient vector $\nabla J(\bar{x})$ enter into the expression for b_i. All of these can be obtained directly from $\nabla^2 J(\bar{x})$ and $\nabla J(\bar{x})$ and forming them does not require any multiplication. Computation of a single vector inner product between the ith column of the Hessian and $\tilde{x}$ is required in b_i, which typically takes $O(m)$ operations. To avoid direct multiplication, in each iteration after finding δ_i, we update the vector $\nabla^2 J(\bar{x})^T \tilde{x}$ using the correction term $\delta_i (E^T Y E_i) \circ ((G^{-1} E_i)^T E)^T$ and take its ith element to form b_i. Here, E_i is the ith column of the incidence matrix of the controller graph. This also avoids the need to store the Hessian of J, which is an $m \times m$ matrix, thereby leading to a significant memory saving.

Remark 4: Active set strategy is an effective means for determining the directions that do not need to be updated in the coordinate descent algorithm. At each outer iteration, we classify the variable as either active or inactive based on the values of $\bar{x}_i$ and the ith component of the gradient vector $\nabla J(\bar{x})$. For $g(x) = \gamma \, \|x\|_1$, the ith search direction is inactive if

$$\bar{x}_i \; = \; 0 \;\; \text{and} \;\; |\, \mathrm{e}_i^T \, \nabla J(\bar{x}) \,| \; < \; \gamma \, - \, \epsilon$$

and it is active otherwise. Here, $\epsilon > 0$ is a small number (e.g., $\epsilon = 0.0001\gamma$). The Newton direction is then obtained by solving the optimization problem over the set of active variables. This significantly improves algorithmic efficiency for large values of the regularization parameter γ.

Convergence analysis: In (SP), $J(x)$ is smooth for $G_p + E \, \mathrm{diag}(x) \, E^T \succ 0$ and the non-smooth part is given by the ℓ_1 norm of x. The objective function of the form $J(x) + g(x)$ was studied in [36], where J is smooth over the positive definite cone and g is a separable non-differentiable function. Theorems 1 and 2 from [36] thus imply super-linear convergence of the quadratic approximation method for (SP).

Stopping criteria: In all three algorithms, the norms of the primal and dual residuals r_p and $r_d^{\pm}$ as well as the duality gap η are used as stopping criteria. In contrast to the stopping criteria available in the literature, this choice enables fair comparison of the algorithms. In our proximal algorithms, we use (13) to construct a dual feasible $\hat{Y}$ and obtain y_+ and y_- from (11). In each iteration, η, r_p, and $r_d^{\pm}$ are evaluated using (12) and (17).

V. Growing connected resistive networks

The problem of topology identification and optimal design of stochastically-forced networks has many interesting variations. An important class is given by resistive networks in which all edge weights are non-negative, $x \, \geq \, 0$ [7]. Here, we study the problem of growing connected resistive networks [13], [14]. In this, the plant graph is connected and there are *no joint edges* between the plant and the controller graphs. Our objective is to enhance the closed-loop performance by adding a small number of edges. As we show below, inequality constraints in this case amount to non-negativity of controller edge weights. This simplifies optimality conditions and enables further improvement of the computational efficiency of our customized algorithms.

The restriction on connected plant graphs implies positive definiteness of the strengthened graph Laplacian of the plant, $G_p \, = \, L_p \, + \, (1/n) \, \mathbb{1}\mathbb{1}^T \, \succ \, 0$. Thus, $G_p \, + \, E \, \mathrm{diag}\,(x) \, E^T$ is always positive definite for connected resistive networks and (SP) simplifies to

$$\underset{x}{\text{minimize}} \quad f(x) \; + \; g(x) \tag{23}$$

where

$$f(x) \; := \; J(x) \; + \; \gamma \, \mathbb{1}^T x$$

and $g(x)$ is the indicator function for the non-negative orthant,

$$g(x) \; := \; I_+(x) \; = \; \left\{ \begin{array}{cl} 0, & x \, \geq \, 0 \\ +\infty, & \text{otherwise.} \end{array} \right.$$

As in Section III, in order to determine the Lagrange dual of the optimization problem (23), we introduce an additional optimization variable G and rewrite (23) as

$$\begin{array}{ll} \underset{G, \, x}{\text{minimize}} & \big\langle G^{-1}, Q_p \big\rangle \, + \, (\gamma \, \mathbb{1} \, + \, \mathrm{diag}\, \big(E^T R \, E\big))^T x \\ \text{subject to} & G \, - \, G_p \, - \, E \, \mathrm{diag}\,(x) \, E^T \; = \; 0 \\ & x \; \geq \; 0. \end{array} \tag{P1}$$

Proposition 3: The Lagrange dual of the primal optimization problem (P1) is given by

$$\begin{array}{ll} \underset{Y}{\text{maximize}} & 2 \, \mathrm{trace} \left((Q_p^{1/2} \, Y \, Q_p^{1/2})^{1/2} \right) \, - \, \langle Y, G_p \rangle \\ \text{subject to} & \mathrm{diag}\, \big(E^T (Y \, - \, R) \, E\big) \; \leq \; \gamma \, \mathbb{1} \\ & Y \, \succ \, 0, \;\; Y \, \mathbb{1} \; = \; \mathbb{1} \end{array} \tag{D1}$$

where Y is the dual variable associated with the equality constraint in (P1). The duality gap is

$$\eta \; = \; y^T x \; = \; \mathbb{1}^T (y \, \circ \, x) \tag{24}$$

where

$$y \; := \; \gamma \, \mathbb{1} \; - \; \mathrm{diag}\, \big(E^T (Y - R) \, E\big) \; \geq \; 0 \tag{25}$$

represents the dual variable associated with the non-negativity constraint on the vector of the edge weights x.

Remark 5: For connected resistive networks with the control weight $R = r \, I$, $\hat{Y}$ given by (13a) is dual feasible if

$$\beta \; \leq \; \frac{\gamma \, + \, 2 \, r}{\max\,(\mathrm{diag}\,(E^T (Y \, - \, R) \, E)) \, + \, 2 \, r}. \tag{26}$$

A. Primal-dual interior-point method

In this case, the central path equations are given by

$$\gamma \, \mathbb{1} \; - \; \mathrm{diag}\, \big(E^T (Y(x) \, - \, R) \, E\big) \, - \, y \; = \; 0 \tag{27a}$$

$$y \, \circ \, x \; = \; \sigma \, \mu \, \mathbb{1} \tag{27b}$$

with $x \geq 0$ and $y \geq 0$. Equation (27b) is obtained by relaxing the complementary slackness condition and $Y(x)$ is given by (14). For $\bar{x} > 0$ and $\bar{y} > 0$ that are infeasible (i.e., do not satisfy (27a)), the dual residual is determined by

$$r_d(\bar{x}, \bar{y}) \; := \; \gamma \, \mathbb{1} \; - \; \mathrm{diag}\, \big(E^T (Y(\bar{x}) \, - \, R) \, E\big) \; - \; \bar{y}. \tag{28}$$

The search direction is obtained by expressing $\tilde{y}$ in terms of $\tilde{x}$,

$$\tilde{y} \; = \; -D_{\bar{x}}^{-1} D_{\bar{y}} \, \tilde{x} \; - \; \bar{y} \tag{29a}$$

and solving the linearized system of central path equations,

$$A \, \tilde{x} \; = \; b \tag{29b}$$

with $A := H + D_{\bar{x}}^{-1} D_{\bar{y}}$, $b := -\left(\bar{y} + r_d(\bar{x}, \bar{y})\right)$, and $H(\bar{x}) := 2\left(H_1(\bar{x}) \circ H_2(\bar{x})\right)$, where the matrices H_1 and H_2 are given in Proposition 1. Positive definiteness of $D_{\bar{x}}^{-1} D_{\bar{y}}$ and H (elementwise product of two positive definite matrices is positive definite) implies positive definiteness of A. Thus, Cholesky factorization of A followed by back solve operations can be used to to determine $(\tilde{x}^a, \tilde{y}^a)$ and $(\tilde{x}, \tilde{y})$. For large problems, the search direction is obtained via an inexact method based on the PCG algorithm with diagonal preconditioner, $\Pi := I \circ A$.

B. Proximal gradient method

Using a simple quadratic approximation of the smooth part of the objective function f around the current iterate x^k

$$f(x) \;\approx\; f(x^k) \;+\; \nabla f(x^k)^T(x \,-\, x^k) \;+\; \frac{1}{2\alpha_k}\,\|x \,-\, x^k\|_2^2$$

the optimal solution of optimization problem (23) is determined by the proximal operator of the function $g(x) = I_+(x)$,

$$x^{k+1} \;=\; \big(x^k \,-\, \alpha_k \nabla f(x^k)\big)_+$$

where $(\cdot)_+$ is the projection on the non-negative orthant. Thus, the action of the proximal operator is given by the projected gradient.

As in Section IV-B, we initialize α_k using the BB heuristics but we skip the backtracking step here and employ a non-monotone BB scheme [48], [49]. The effectiveness of this strategy has been established on quadratic problems [31], [48], [50], but its convergence in general is hard to prove. In Section VI, we demonstrate efficiency of this approach.

C. Proximal Newton method

We next adjust the customized algorithm based on proximal Newton method for growing connected resistive networks. We approximate the smooth part of the objective function f in (23) using the second-order Taylor series expansion around the current iterate $\bar{x}$,

$$f(\bar{x} + \tilde{x}) \;\approx\; f(\bar{x}) \;+\; \nabla f(\bar{x})^T\,\tilde{x} \;+\; \frac{1}{2}\,\tilde{x}^T\,\nabla^2 f(\bar{x})\,\tilde{x}$$

and rewrite (23) as

$$\begin{array}{ll} \underset{\tilde{x}}{\text{minimize}} & \nabla f(\bar{x})^T\,\tilde{x} \;+\; \dfrac{1}{2}\,\tilde{x}^T\,\nabla^2 f(\bar{x})\,\tilde{x} \\ \text{subject to} & \bar{x} \;+\; \tilde{x} \;\geq\; 0. \end{array} \tag{30}$$

By perturbing $\tilde{x}$ in the direction of the ith standard basis vector e_i in $\mathbb{R}^m$, $\tilde{x} + \delta_i\,\mathrm{e}_i$, the objective function in (30) becomes

$$\nabla f(\bar{x})^T(\tilde{x} \,+\, \delta_i\,\mathrm{e}_i) \;+\; \frac{1}{2}\,(\tilde{x} \,+\, \delta_i\,\mathrm{e}_i)^T\,\nabla^2 f(\bar{x})\,(\tilde{x} \,+\, \delta_i\,\mathrm{e}_i)\,.$$

Elimination of constant terms allows us to bring (30) into

$$\begin{array}{ll} \underset{\delta_i}{\text{minimize}} & \dfrac{1}{2}\,a_i\,\delta_i^2 \;+\; b_i\,\delta_i \\ \text{subject to} & \bar{x}_i \;+\; \tilde{x}_i \;+\; \delta_i \;\geq\; 0. \end{array} \tag{31}$$

The optimization variable is the scalar δ_i and a_i, b_i, $\bar{x}_i$, and $\tilde{x}_i$ are the problem data with

$$\begin{aligned} a_i \;&:=\; \mathrm{e}_i^T\,\nabla^2 f(\bar{x})\,\mathrm{e}_i \\ b_i \;&:=\; \big(\nabla^2 f(\bar{x})\,\mathrm{e}_i\big)^T\tilde{x} \;+\; \mathrm{e}_i^T\,\nabla f(\bar{x}) \end{aligned}$$

The explicit solution to (31) is given by

$$\delta_i \;=\; \begin{cases} \quad -b_i/a_i, & \bar{x}_i \,+\, \tilde{x}_i \,-\, b_i/a_i \;\geq\; 0 \\ -\,(\bar{x}_i \,+\, \tilde{x}_i)\,, & \text{otherwise}\,. \end{cases}$$

After the Newton direction $\tilde{x}$ has been computed, we determine the step-size α via backtracking. This guarantees positivity of the updated vector of the edge weights, $\bar{x} + \alpha\tilde{x}$, and sufficient decrease of the objective function, $f(\bar{x} + \alpha\tilde{x}) \leq f(\bar{x}) + \alpha\,\sigma\,\nabla f(\bar{x})^T\tilde{x}$.

Remark 6: As in Section IV-C, we use an active set strategy to identify the directions that do not need to be updated in the coordinate descent algorithm. For $g(x) = I_+(x)$, the ith search direction is inactive if

$$\bar{x}_i \;=\; 0 \;\;\text{and}\;\; \mathrm{e}_i^T\,\nabla f(\bar{x}) \;\geq\; 0$$

and it is active otherwise.

Stopping criteria: The norm of the dual residual, r_d, and the duality gap, η, are used as stopping criteria. For proximal algorithms, the dual variable y is obtained from (25) where $\hat{Y}$ is given by (13a) and β satisfies (26). At each iteration, η and r_d are evaluated using (24) and (28).

VI. Computational experiments

Herein, we provide examples and compare performance of our customized algorithms. The direct primal-dual IP algorithm uses Cholesky factorization to compute the search direction, and the indirect algorithm uses PCG that does not store the matrix A. Algorithm proxBB represents proximal gradient method with BB step-size initialization and proxN identifies proximal Newton method in which the search direction is found via coordinate descent. We have implemented all algorithms in MATLAB. All tests were executed on a 3.4 GHz Core(TM) i7-3770 Intel(R) machine with 16GB RAM.

In all examples, we set $R = I$ and choose the state weight that penalizes the mean-square deviation from the network average, $Q \,=\, I \,-\, (1/n)\,\mathbb{1}\mathbb{1}^T$. The absolute value of the dual residual, r_d, and the duality gap, η, are used as stopping criteria. We set the tolerances for r_d and η to 10^{-3} and 10^{-4}, respectively. Finally, for connected plant networks

$$\gamma_{\max} \;:=\; \|\,\mathrm{diag}\,(E^T\,G_p^{-1}\,Q\,G_p^{-1}\,E)\,\|_\infty$$

identifies the value of the regularization parameter γ for which all edge weights in the controller graph are equal to zero.

Additional information about our computational experiments, along with MATLAB source codes, can be found at:

www.ece.umn.edu/∼mihailo/software/graphsp/

A. Performance comparison

We first solve the problem (P1) for growing connected resistive Erdös-Rényi networks with different number of nodes ($n = 5$ to $n = 100$). The generator of the plant dynamics is given by an undirected unweighted graph with the edge probability $1.05\log(n)/n$. The incidence matrix of the controller graph is selected to satisfy the following requirements: (i) in the absence of the sparsity-promoting term, the closed-loop network is given by a complete graph; and (ii) there are no joint edges between the plant and the controller graphs. As the size of the network increases, the implementation of the IP method based on PCG is significantly more efficient than the implementation based on Cholesky factorization. These algorithms are, on average, about 206 and 41 times faster than CVX (for sizes that can be handled by CVX), respectively.

Table I compares our customized algorithms in terms of speed and the number of iterations. For $n = 300$, the direct IP method runs out of memory and the indirect method computes the optimal solution in about 16 seconds (these results are

TABLE I: Comparison of our algorithms (solve times in seconds/number of iterations) for the problem of growing connected resistive Erdös-Rényi networks with different number of nodes n, edge probability $1.05\log(n)/n$, and $\gamma = 0.8\,\gamma_{\max}$.

number of nodes	$n = 300$	$n = 700$	$n = 1000$	$n = 1300$	$n = 1500$
number of edges	$m = 43986$	$m = 242249$	$m = 495879$	$m = 839487$	$m = 1118541$
IP (PCG)	$16.499/8$	$394.256/13$	$1014.282/13$	$15948.164/13$	$179352.208/14$
proxBB	$1.279/11$	$15.353/11$	$55.944/13$	$157.305/16$	$239.567/16$
proxN	$1.078/4$	$11.992/4$	$34.759/4$	$82.488/4$	$124.307/4$

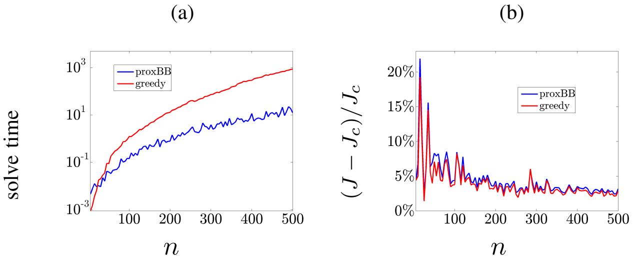


Fig. 1: (a) Solve times (in seconds); and (b) performance degradation of proximal gradient and greedy algorithms relative to the optimal centralized controller.

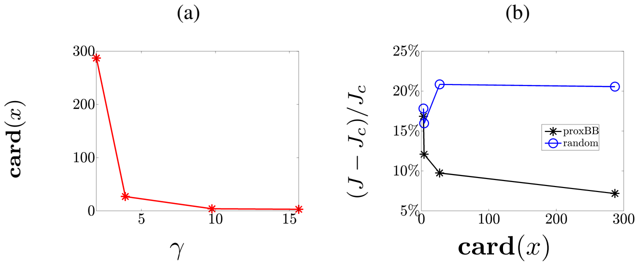


Fig. 2: (a) Sparsity level; and (b) optimal tradeoff curves resulting from the application of proximal gradient algorithm and a heuristic strategy for the Facebook network.

not shown in Table I). Even for small networks, proximal methods are significantly faster than the IP method and proxN takes smaller number of iterations and converges quicker than proxBB. For a larger network (with 1500 nodes and 1118541 edges in the controller graph), it takes about 50 hours for the PCG-based IP method to solve the problem. In contrast, proxN and proxBB converge in about 2 and 4 minutes, respectively.

Figure 1 compares our proximal gradient algorithm with the fast greedy algorithm of [19]. We solve problem (P1) for Erdös-Rényi networks with different number of nodes ($n = 5$ to 500) and $\gamma = 0.4\,\gamma_{\max}$. After proxBB identifies the edges in the controller graph, we use the greedy method to select the same number of edges. Finally, we polish the identified edge weights for both methods. Figure 1a shows the solve times (in seconds) versus the number of nodes. As the number of nodes increases the proximal algorithm significantly outperforms the fast greedy method. Relative to the optimal centralized controller, both methods yield similar performance degradation of the closed-loop network; see Fig. 1b.

B. Large-scale Facebook network

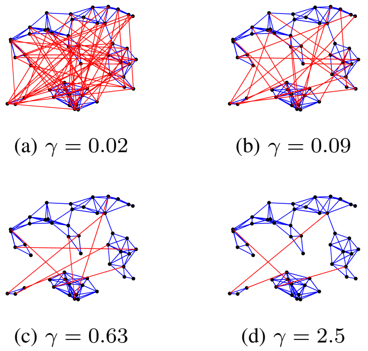
To evaluate effectiveness of our algorithms on large networks, we solve the problem of growing a network of friendships. In such social networks, nodes denote people and edges denote friendships. There is an edge between two nodes if the two people are friends [51]. The ego-Facebook network is undirected and unweighted with 4039 nodes and 88234 edges; the data is available at http://snap.stanford.edu/data/. Our objective is to improve performance by adding a small number of extra edges. We assume that people can only form friendships with friends of their friends. This restricts the number of potential edges in the controller graph to 1358067.

To avoid memory issues, we have implemented our algorithms in C++. For $\gamma = c\,\gamma_{\max}$ with $c = \{0.1, 0.2, 0.5, 0.8\}$ and $\gamma_{\max} = 19.525$, the proximal gradient algorithm computes the solution in about 10, 2.6, 0.87, and 0.43 hours, respectively. After identifying the topology of the controller graph, we compute the optimal edge weights via polishing.

Figure 2a shows that the number of nonzero elements in the vector x decreases as γ increases. Moreover, Fig. 2b illustrates that the $\mathcal{H}_2$ performance deteriorates as the number of nonzero elements in x decreases. In particular, for $\gamma = 0.8\,\gamma_{\max}$, the identified sparse controller has only 3 nonzero elements (it uses only 0.0002% of the potential edges). Relative to the optimal centralized controller, this controller degrades performance by 16.842%, $(J - J_c)/J_c = 16.842\%$.

The Facebook network consists of 10 ego nodes, $\{1, 108, 349, 415, 687, 699, 1685, 1913, 3438, 3981\}$. All other nodes are friends to at least one of these ego nodes [51]. In all of our experiments, the added links with the largest edge weights connect either the ego nodes to each other or three non-ego nodes, $\{429, 564, 568\}$, to the ego nodes. Thus, our method identifies non-ego nodes that play an important role in improving the network performance.

We compare performance of the identified controller to a heuristic strategy that is described next. Controller graph contains 16 potential edges between ego nodes. If the number of edges identified by our method is smaller than 16, we randomly select the desired number of edges between ego nodes. Otherwise, we connect all ego nodes and select the remaining edges in the controller graph randomly. We then use polishing to find the optimal edge weights. The performance of resulting random controller graphs are averaged over 10 trials and the performance loss relative to the optimal centralized controller is displayed in Fig. 2b. We see that our algorithm always performs better than the heuristic strategy. On the other hand, the heuristic strategy outperforms the strategy that adds edges


(a) $\gamma = 0.02$ (b) $\gamma = 0.09$

(c) $\gamma = 0.63$ (d) $\gamma = 2.5$

Fig. 3: Topologies of the plant (blue lines) and controller graphs (red lines) for an unweighted random network with three disconnected subgraphs.

randomly (without paying attention to ego nodes). Unlike our method, the heuristic strategy does not necessarily improve the performance by increasing the number of added edges. In fact, the performance deteriorates as the number of edges in the controller graph increases from 4 to 27; see Fig. 2b.

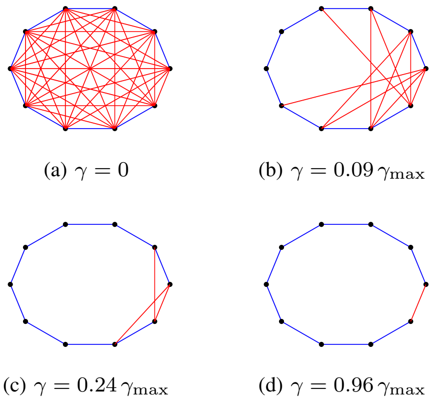
C. Random disconnected network

The plant graph (blue lines) in Fig. 3 contains 50 randomly distributed nodes in a region of 10×10 units. Two nodes are neighbors if their Euclidean distance is not greater than 2 units. We examine the problem of adding edges to a plant graph which is not connected and solve the sparsity-promoting optimal control problem (SP) for controller graph with $m = 1094$ potential edges. This is done for 200 logarithmically-spaced values of $\gamma \in [10^{-3}, 2.5]$ using the path-following iterative reweighted algorithm as a proxy for inducing sparsity [41]. As indicated by (8), we set the weights to be inversely proportional to the magnitude of the solution x to (SP) at the previous value of γ. We choose $\varepsilon = 10^{-3}$ in (8) and initialize weights for $\gamma = 10^{-3}$ using the solution to (SP) with $\gamma = 0$ (i.e., the optimal centralized vector of the edge weights). Topology identification is followed by the polishing step that computes the optimal edge weights; see Section II-B.

As illustrated in Fig. 3, larger values of γ yield sparser controller graphs (red lines). In contrast to all other examples, the plant graph is not connected and the optimal solution is obtained using the algorithms of Section IV. Note that greedy method [19] cannot be used here. Since the plant graph has three disconnected subgraphs, at least two edges in the controller are needed to make the closed-loop network connected.

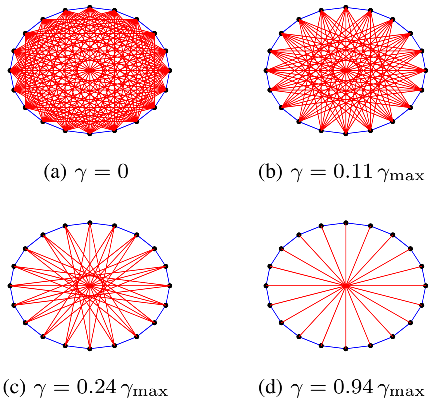
Figure 4 shows that the number of nonzero elements in the vector of the edge weights x decreases and that the closed-loop performance deteriorates as γ increases. In particular, Fig. 4c illustrates the optimal tradeoff curve between the $\mathcal{H}_2$ performance loss (relative to the optimal centralized controller) and the sparsity of the vector x. For $\gamma = 2.5$, only four edges are added. Relative to the optimal centralized vector of the controller edge weights x_c, the identified sparse controller in this case uses only 0.37% of the edges, and achieves a performance loss of 82.13%, i.e.,

$$\mathbf{card}(x)/\mathbf{card}(x_c) \;=\; 0.37\%, \;\; (J \,-\, J_c)/J_c \;=\; 82.13\%.$$


(a) $\gamma = 0$ (b) $\gamma = 0.09\,\gamma_{\max}$

(c) $\gamma = 0.24\,\gamma_{\max}$ (d) $\gamma = 0.96\,\gamma_{\max}$

Fig. 5: The problem of growing unweighted path network. Blue lines identify edges in the plant graph, and red lines identify edges in the controller graph.


(a) $\gamma = 0$ (b) $\gamma = 0.11\,\gamma_{\max}$

(c) $\gamma = 0.24\,\gamma_{\max}$ (d) $\gamma = 0.94\,\gamma_{\max}$

Fig. 6: The problem of growing unweighted ring network. Blue lines identify edges in the plant graph, and red lines identify edges in the controller graph.

Here, x_c is the solution to (SP) with $\gamma = 0$ and the pattern of non-zero elements of x is obtained by solving (SP) with $\gamma = 2.5$ via the path-following iterative reweighted algorithm.

D. Path and ring networks

For path networks, our computational experiments show that for a large enough value of the sparsity-promoting parameter γ a single edge, which generates the longest cycle, is added; see Fig. 5. This is in agreement with [16] where it was proved that the longest cycle is most beneficial for improving the $\mathcal{H}_2$ performance of tree networks. Similar observations can be made for the spatially-invariant ring network with nearest neighbor interactions. For large values of γ, each node establishes a link to the node that is farthest away in the network; see Fig. 6. It is worth noting that this is in agreement with recent theoretical developments [11] where perturbation analysis was used to identify optimal week links in edge-transitive consensus networks. Thus, for these regular networks and large enough values of the regularization parameter, our approach indeed provides the globally optimal solution to the original non-convex cardinality minimization problem.

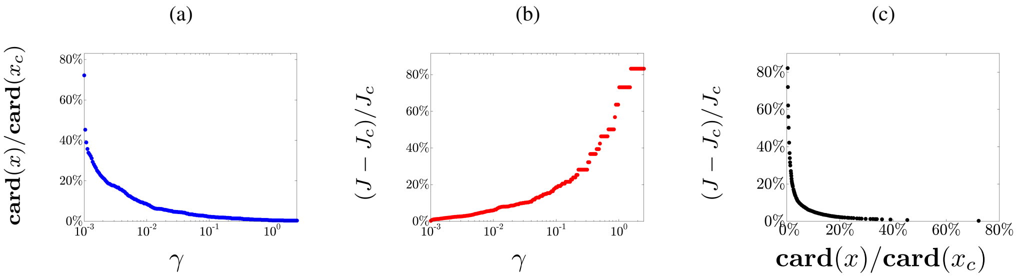


Fig. 4: (a) Sparsity level; (b) performance degradation; and (c) the optimal tradeoff curve between the performance degradation and the sparsity level of optimal sparse x compared to the optimal centralized vector of the edge weights x_c. The results are obtained for unweighted random disconnected plant network with topology shown in Fig. 3.

VII. Concluding remarks

We have examined the problem of optimal topology identification and design of the corresponding edge weights for undirected consensus networks. Our approach uses convex optimization to balance performance of stochastically-forced networks with the number of edges in the distributed controller. For ℓ_1-regularized minimum variance optimal control problem, we have derived a Lagrange dual and exploited structure of the optimality conditions for undirected networks to develop three customized algorithms that are well-suited for large problems. These are based on the infeasible primal-dual interior-point, the proximal gradient, and the proximal Newton methods. The proximal gradient algorithm is a first-order method that updates the controller graph Laplacian via the use of the soft-thresholding operator. In the IP method, the Newton direction is obtained using an inexact iterative procedure based on the preconditioned conjugate gradients and, in the sequential quadratic approximation method, it is computed using cyclic coordinate descent over the set of active variables. Examples are provided to demonstrate utility of our algorithms. We have shown that proximal algorithms can solve the problems with millions of edges in the controller graph in several minutes, on a PC. Furthermore, we have specialized our algorithm to the problem of growing connected resistive networks. In this, the plant graph is connected and there are no joint edges between the plant and the controller graphs. We have exploited structure of such networks and demonstrated how additional edges can be systematically added in a computationally efficient manner.

Acknowledgments

We thank J. W. Nichols for his feedback on earlier versions of this manuscript, T. H. Summers for useful discussion, and M. Sanjabi for his help with C++ implementation.

References

[1] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 2010.

[2] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Syst. Control Lett.*, vol. 53, pp. 65–78, 2004.

[3] Y. Kim and M. Mesbahi, "On maximizing the second smallest eigenvalue of a state-dependent graph Laplacian," *IEEE Trans. Automat. Control*, vol. 51, no. 1, pp. 116–120, 2006.

[4] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2508–2530, 2006.

[5] L. Xiao, S. Boyd, and S.-J. Kim, "Distributed average consensus with least-mean-square deviation," *J. Parallel Distrib. Comput.*, vol. 67, no. 1, pp. 33–46, 2007.

[6] P. Barooah and J. P. Hespanha, "Estimation on graphs from relative measurements: Distributed algorithms and fundamental limits," *IEEE Control Syst. Mag.*, vol. 27, no. 4, pp. 57–74, 2007.

[7] A. Ghosh, S. Boyd, and A. Saberi, "Minimizing effective resistance of a graph," *SIAM Rev.*, vol. 50, no. 1, pp. 37–66, 2008.

[8] G. F. Young, L. Scardovi, and N. E. Leonard, "Robustness of noisy consensus dynamics with directed communication," in *Proceedings of the 2010 American Control Conference*, 2010, pp. 6312–6317.

[9] D. Zelazo and M. Mesbahi, "Edge agreement: Graph-theoretic performance bounds and passivity analysis," *IEEE Trans. Automat. Control*, vol. 56, no. 3, pp. 544–555, 2011.

[10] B. Bamieh, M. R. Jovanović, P. Mitra, and S. Patterson, "Coherence in large-scale networks: dimension dependent limitations of local feedback," *IEEE Trans. Automat. Control*, vol. 57, no. 9, pp. 2235–2249, 2012.

[11] M. Fardad, X. Zhang, F. Lin, and M. R. Jovanović, "On the properties of optimal weak links in consensus networks," in *Proceedings of the 53rd IEEE Conference on Decision and Control*, 2014, pp. 2124–2129.

[12] I. Poulakakis, G. Young, L. Scardovi, and N. E. Leonard, "Information centrality and ordering of nodes for accuracy in noisy decision-making networks," *IEEE Trans. Automat. Control*, vol. 61, no. 4, pp. 1040–1045, 2016.

[13] A. Ghosh and S. Boyd, "Growing well-connected graphs," in *Proceedings of the 45th IEEE Conference on Decision and Control*, 2006, pp. 6605–6611.

[14] R. Dai and M. Mesbahi, "Optimal topology design for dynamic networks," in *50th IEEE Conference on Decision and Control*, 2011, pp. 1280–1285.

[15] F. Lin, M. Fardad, and M. R. Jovanović, "Identification of sparse communication graphs in consensus networks," in *Proceedings of the 50th Annual Allerton Conference on Communication, Control, and Computing*, Monticello, IL, 2012, pp. 85–89.

[16] D. Zelazo, S. Schuler, and F. Allgöwer, "Performance and design of cycles in consensus networks," *Syst. Control Lett.*, vol. 62, no. 1, pp. 85–96, 2013.

[17] M. Fardad, F. Lin, and M. R. Jovanović, "Design of optimal sparse interconnection graphs for synchronization of oscillator networks," *IEEE Trans. Automat. Control*, vol. 59, no. 9, pp. 2457–2462, 2014.

[18] X. Wu and M. R. Jovanović, "Sparsity-promoting optimal control of consensus and synchronization networks," in *Proceedings of the 2014 American Control Conference*, Portland, OR, 2014, pp. 2948–2953.

[19] T. H. Summers, I. Shames, J. Lygeros, and F. Dörfler, "Topology design for optimal network coherence," in *Proceedings of the 2015 European Control Conference*, 2015, pp. 575–580.

[20] M. Fardad, F. Lin, and M. R. Jovanović, "Sparsity-promoting optimal control for a class of distributed systems," in *Proceedings of the 2011 American Control Conference*, 2011, pp. 2050–2055.

[21] F. Lin, M. Fardad, and M. R. Jovanović, "Design of optimal sparse feedback gains via the alternating direction method of multipliers," *IEEE Trans. Automat. Control*, vol. 58, no. 9, pp. 2426–2431, 2013.

[22] J. F. Sturm, "Using SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones," *Optim. Methods Softw.*, vol. 11, no. 1-4, pp. 625–653, 1999.

[23] K.-C. Toh, M. J. Todd, and R. H. Tütüncü, "SDPT3 – A Matlab software package for semidefinite programming, version 1.3," *Optim. Methods Softw.*, vol. 11, no. 1-4, pp. 545–581, 1999.

[24] S. Hassan-Moghaddam and M. R. Jovanović, "An interior point method for growing connected resistive networks," in *Proceedings of the 2015 American Control Conference*, Chicago, IL, 2015, pp. 1223–1228.

[25] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An interior-point method for large-scale ℓ_1-regularized least squares," *IEEE J. Sel. Topics Signal Process.*, vol. 1, no. 4, pp. 606–617, 2007.

[26] K. Koh, S.-J. Kim, and S. P. Boyd, "An interior-point method for large-scale ℓ_1-regularized logistic regression." *J. Mach. Learn. Res.*, vol. 8, no. 8, pp. 1519–1555, 2007.

[27] L. Li and K.-C. Toh, "An inexact interior point method for ℓ_1-regularized sparse covariance selection," *Math. Prog. Comp.*, vol. 2, no. 3-4, pp. 291–315, 2010.

[28] G. Al-Jeiroudi and J. Gondzio, "Convergence analysis of the inexact infeasible interior-point method for linear optimization," *J. Optim. Theory Appl.*, vol. 141, no. 2, pp. 231–247, 2009.

[29] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imaging Sci.*, vol. 2, no. 1, pp. 183–202, 2009.

[30] N. Parikh and S. Boyd, "Proximal algorithms," *Foundations and Trends in Optimization*, vol. 1, no. 3, pp. 123–231, 2013.

[31] J. Barzilai and J. M. Borwein, "Two-point step size gradient methods," *IMA J. Numer. Anal.*, vol. 8, no. 1, pp. 141–148, 1988.

[32] J. D. Lee, Y. Sun, and M. A. Saunders, "Proximal Newton-type methods for minimizing composite functions," *SIAM J. Optim.*, vol. 24, no. 3, pp. 1420–1443, 2014.

[33] P. Tseng, "Convergence of a block coordinate descent method for nondifferentiable minimization," *J. Optim. Theory Appl.*, vol. 109, no. 3, pp. 475–494, 2001.

[34] S. Shalev-Shwartz and A. Tewari, "Stochastic methods for ℓ_1 regularized loss minimization," *J. Mach. Learn. Res.*, vol. 12, pp. 1865–1892, June 2011.

[35] A. Saha and A. Tewari, "On the non-asymptotic convergence of cyclic coordinate descent methods," *SIAM J. Optim.*, vol. 23, no. 1, pp. 576–601, 2013.

[36] C.-J. Hsieh, M. A. Sustik, I. S. Dhillon, and P. Ravikumar, "QUIC: Quadratic approximation for sparse inverse covariance estimation," *J. Mach. Learn. Res.*, vol. 15, pp. 2911–2947, 2014.

[37] E. J. Candès and T. Tao, "Near optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406–5425, 2006.

[38] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[39] E. J. Candès, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Commun. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, 2006.

[40] T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning: data mining, inference, and prediction*. Springer, 2009.

[41] E. J. Candès, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *J. Fourier Anal. Appl*, vol. 14, pp. 877–905, 2008.

[42] J. Nocedal and S. J. Wright, *Numerical Optimization*. Springer, 2006.

[43] L. Vandenberghe, "Lecture 15: Primal-dual interior-point method," http://www.ee.ucla.edu/ee236a/lectures/mpc.pdf, Lecture notes for EE236A – Linear Programming; Fall Quarter 2013-14.

[44] S. Mehrotra, "On the implementation of a primal-dual interior point method," *SIAM J. Optim.*, vol. 2, no. 4, pp. 575–601, 1992.

[45] Y. Saad, *Iterative methods for sparse linear systems*. SIAM, 2003.

[46] S. Yun and K.-C. Toh, "A coordinate gradient descent method for ℓ_1 regularized convex minimization," *Computational Optimization and Applications*, vol. 48, no. 2, pp. 273–307, 2011.

[47] P. Tseng and S. Yun, "A coordinate gradient descent method for nonsmooth separable minimization," *Math. Prog.*, vol. 117, no. 1-2, pp. 387–423, 2009.

[48] Y.-H. Dai and R. Fletcher, "Projected Barzilai-Borwein methods for large-scale box-constrained quadratic programming," *Numerische Mathematik*, vol. 100, no. 1, pp. 21–47, 2005.

[49] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2479–2493, 2009.

[50] T. Serafini, G. Zanghirati, and L. Zanni, "Gradient projection methods for quadratic programs and applications in training support vector machines," *Optim. Methods Softw.*, vol. 20, no. 2-3, pp. 353–378, 2005.

[51] J. J. McAuley and J. Leskovec, "Learning to discover social circles in ego networks." in *Adv. Neural Inf. Process. Syst.*, 2012, pp. 539–547.