2017 IEEE 56th Annual Conference on Decision and Control (CDC)
December 12-15, 2017, Melbourne, Australia

Distributed design of optimal structured feedback gains

Sepideh Hassan-Moghaddam and Mihailo R. Jovanović

Abstract— **We study the problem of designing optimal structured feedback gains for large-scale systems in a distributed manner. We quantify performance using the $\mathcal{H}_2$ norm and introduce regularization functions to promote desired structural properties in the controller. The key challenge is to evaluate the objective function and its gradient without solving the large-scale Lyapunov equations. We exploit the square-additive property of the $\mathcal{H}_2$ norm and provide an equivalent consensus-based characterization that is convenient for distributed implementation. The resulting constrained optimal control problem is in general non-convex and we solve it using an algorithm based on the alternating direction method of multipliers. This algorithm exploits separability of the objective function and is guaranteed to converge to a local minimum. For the optimal control of symmetric systems and undirected consensus networks we establish convexity and demonstrate how the underlying structure can be exploited to further simplify computations.**

Index Terms— **Alternating direction method of multipliers, consensus, distributed feedback design, large-scale systems, optimization, sparsity-promoting optimal control.**

I. Introduction

In large-scale systems, conventional control strategies that rely on centralized computation and implementation are often prohibitively expensive. For example, finding the optimal controller requires computation of the solution to the algebraic Riccati equations which is often infeasible because of the overwhelming computational requirements. This necessitates the development of theory and techniques that utilize distributed computing architectures to cope with large problem sizes.

Modern control applications impose additional requirements on controller design that cannot be addressed using standard optimal control tools. These requirements may arise from limited communication and computation resources or the size of the problem. The standard optimal control techniques typically induce an all-to-all communication requirements in the controller which is infeasible in large-scale setting. Recently, regularization has emerged as a promising tool for enhancing utility of standard optimal control techniques. In this approach, commonly used performance measures (e.g., $\mathcal{H}_2$ or $\mathcal{H}_\infty$) are augmented with regularization functions that are supposed to promote some desired structural features in the distributed controller, e.g., sparsity. Such an approach has received significant level of attention in recent years [1]–[7], but computing optimal solutions in large-scale problems still remains a challenge. Distributed computing techniques have been commonly used to cope with large problem sizes; for example, stability and synthesis of cooperative distributed model predictive controllers for linear systems have been recently studied in [8].

In this paper, for the regularized $\mathcal{H}_2$ optimal control problem, we exploit the square-additive property of the $\mathcal{H}_2$ norm to provide an equivalent consensus-based characterization that is convenient for distributed implementation. Furthermore, we demonstrate that the objective function and its gradients can be evaluated without solving large-scale Lyapunov equations. In part, our approach is inspired by the framework developed in [9], [10] where an iterative method for computing the structured linear quadratic regulator is proposed. Instead of solving the Lyapunov equations, the authors have developed a gradient algorithm that forms a search direction via numerical integration of the primal and adjoint systems. In contrast to [9], [10], we exploit separability of the objective function and utilize an ADMM-based consensus algorithm to solve the regularized optimal control problem in a distributed manner over multiple processors.

Even though the optimal control problem is in general non-convex, recent results can be utilized to show convergence to a local minimum [11]. The ADMM-based consensus algorithm that we use is standard (e.g., see [12]) but, to the best of our knowledge, it has not been previously used for distributed design of structured feedback gains. In [2], [13], ADMM was used as a general optimization tool to compute sparsity-promoting controllers in a centralized fashion. Herein, we exploit structure of the underlying optimal control problem to compute structured feedback gains in a distributed manner.

The rest of the paper is structured as follows. In Section II, we formulate regularized $\mathcal{H}_2$ optimal control problem, provide the decomposition of the $\mathcal{H}_2$ norm, and specialize the problem to symmetric systems and undirected networks. In Section III, we develop an ADMM-based consensus algorithm which is well-suited for large-scale systems and distributed computations. In Section IV, we provide examples and, in Section V, we offer concluding remarks.

II. Problem formulation

We consider the LTI systems

$$\begin{aligned} \dot{\psi} &= A\,\psi + B_1\,d + B_2\,u \\ \zeta &= \begin{bmatrix} Q^{1/2} \\ 0 \end{bmatrix}\psi + \begin{bmatrix} 0 \\ R^{1/2} \end{bmatrix} u \end{aligned}$$

where $d \in \mathbb{R}^p$ and $u \in \mathbb{R}^q$ are the disturbance and control inputs, $\psi \in \mathbb{R}^n$ is the state, and $\zeta \in \mathbb{R}^{n+q}$ is the performance

Financial support from the National Science Foundation under award ECCS-1739210 is gratefully acknowledged.

S. Hassan-Moghaddam and M. R. Jovanović are with the Ming Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089. E-mails: hassanmo@usc.edu, mihailo@usc.edu.

output. The dynamic matrix $A \in \mathbb{R}^{n\times n}$ determines interactions in the open-loop system, the matrices $B_1 \in \mathbb{R}^{n\times p}$ and $B_2 \in \mathbb{R}^{n\times q}$ determine how disturbances and controls enter into the dynamics, and symmetric matrices $Q \succeq 0$ and $R \succ 0$ specify the state and control weights in the performance output. For memoryless control laws, $u = -K\psi$ where $K \in \mathbb{R}^{q\times n}$ is the constant feedback gain matrix, the closed-loop system is given by

$$\begin{aligned} \dot{\psi} &= (A - B_2 K)\psi + B_1 d \\ \zeta &= \begin{bmatrix} Q^{1/2} \\ -R^{1/2}K \end{bmatrix} \psi. \end{aligned} \tag{1}$$

The matrix K that optimizes the closed-loop performance and has certain structural properties can be obtained by solving the regularized optimal control problem

$$\underset{K}{\text{minimize}} \quad f(K) + g(K). \tag{2}$$

Here, f is the function that quantifies the closed-loop performance and g is the regularization function that is introduced to promote certain structural properties of K. For example, when it is desired to design K with a specified pattern of zero elements, g is an indicator function of the set that characterizes this pattern [14]. When it is desired to promote sparsity of K, the ℓ_1 norm $g(K) = \gamma \sum_{i,j} |K_{ij}|$ can be used as a sparsity-enhancing regularizer, where γ is the positive parameter that characterizes emphasis on sparsity [2].

We quantify the closed-loop performance using the square of the $\mathcal{H}_2$ norm of system (1),

$$f(K) = \begin{cases} \text{trace}\left(B_1^T P B_1\right) & K \text{ stabilizing} \\ \infty & \text{otherwise} \end{cases}$$

where P is the closed-loop observability gramian,

$$(A - B_2K)^T P + P(A - B_2K) + Q + K^T R K = 0.$$

Over the set of stabilizing feedback gain matrices K, the gradient of f with respect to K is given by [15]

$$\nabla f(K) = 2(RK - B_2^T P)L$$

where L is the closed-loop controllability gramian,

$$(A - B_2K)L + L(A - B_2K)^T + B_1 B_1^T = 0$$

The objective function and its gradient depend on the controllability and observability gramians L and P of the closed-loop system. These matrices can be obtained by solving the corresponding Lyapunov equations but this is prohibitively expensive for most large-scale systems. Thus, the key challenge is to evaluate the objective function and the corresponding gradient without solving the large-scale Lyapunov equations.

A. Decomposition of the $\mathcal{H}_2$ norm

In this section, we exploit the square-additive property of the $\mathcal{H}_2$ norm to provide an equivalent representation that is convenient for large-scale and distributed optimization.

Using the definition of the matrix trace, for a stabilizing K, we can express the closed-loop $\mathcal{H}_2$ norm as

$$f(K) = \sum_{i=1}^{N} f_i(K)$$

where

$$f_i(K) = (B_1 \mathrm{e}_i)^T P B_1 \mathrm{e}_i = b_i^T P b_i.$$

Here, e_i is the ith canonical basis vector in $\mathbb{R}^N$ and b_i is the ith column of the matrix $B_1 \in \mathbb{R}^{n\times N}$. Furthermore, the closed-loop controllability gramian L and the gradient of f can be written as

$$L = \sum_{i=1}^{N} L_i, \quad \nabla f(K) = \sum_{i=1}^{N} \nabla f_i(K)$$

where

$$(A - B_2K)L_i + L_i(A - B_2K)^T + b_i b_i^T = 0$$

and

$$\nabla f_i(K) = 2(RK - B_2^T P)L_i.$$

We now show that $f_i(K)$ and $\nabla f_i(K)$ can be evaluated without the requirement to solve the corresponding Lyapunov equations for P and L_i (for similar developments in the structured LQR design see [9], [10]).

The function f_i is given by

$$f_i(K) = \text{trace}\left(P b_i b_i^T\right) = \text{trace}\left((Q + K^T R K) L_i\right)$$

where

$$L_i = \int_0^\infty \mathrm{e}^{A_{\mathrm{cl}} t} b_i b_i^T \mathrm{e}^{A_{\mathrm{cl}}^T t}\, \mathrm{d}t \tag{3}$$

and $A_{\mathrm{cl}} := A - B_2K$. Clearly, L_i can be obtained from

$$L_i = \int_0^\infty \psi_i(t)\, \psi_i^T(t)\, \mathrm{d}t$$

where ψ_i is the solution to

$$\dot{\psi}_i = A_{\mathrm{cl}}\, \psi_i, \quad \psi_i(0) = b_i. \tag{4}$$

Similarly,

$$KL_i = -\int_0^\infty u_i(t)\, \psi_i^T(t)\, \mathrm{d}t$$

where ψ_i is the solution to (4) and $u_i = -K\psi_i$. Finally, following [10], it can be shown that

$$PL_i = -\int_0^\infty \phi_i(t)\, \psi_i^T(t)\, \mathrm{d}t$$

where ϕ_i is obtained from the solution of the adjoint system

$$\begin{aligned} \dot{\phi}_i &= -A_{\mathrm{cl}}^T \phi_i + (Q + K^T R K)\psi_i \\ \phi_i(\infty) &= 0 \end{aligned} \tag{5}$$

and ψ_i is the solution to (4). Thus, numerical simulations of the primal and adjoint systems (4) and (5) along with numerical computations of the corresponding integrals can be used to evaluate $f_i(K)$ and $\nabla f_i(K)$.

Remark 1: We note that the integral in (3) can be com-

puted from the matrix exponential of the matrix [16]

$$\begin{bmatrix} A_{\rm cl}^T & 0 \\ b_i b_i^T & -A_{\rm cl}^T \end{bmatrix}.$$

In large-scale problems the computation of the entire matrix exponential is prohibitively expensive and we refrain from using this approach. Furthermore, since $b_i b_i^T$ is a rank-one matrix, efficient computation of L_i is possible [17]. However, since $Q+K^T R\, K$ is typically a full rank matrix, computation of the observability gramian P is challenging.

B. Systems with special structure

If the underlying system has additional structure, evaluation of both the objective function and the corresponding gradients can be further simplified. We next illustrate how structure of symmetric systems and undirected consensus networks can be exploited to simplify computations. In both cases, the closed-loop $\mathcal{H}_2$ norm has an explicit convex dependance on the optimization variable and there is no need to conduct simulations of the adjoint system and numerically evaluate the underlying integrals.

1) Symmetric systems: Let A and K in (1) be symmetric matrices and let $B_1 = B_2 = I$. For a stabilizing K, the square of the $\mathcal{H}_2$ norm of system (1) is determined by

$$f(K) \;=\; \frac{1}{2}\,\mathrm{trace}\left((K\,-\,A)^{-1}(Q\,+\,K\,R\,K)\right).$$

Convex dependence of f on K can be established via a straightforward use of the Schur complement. Furthermore, the design for symmetric systems provides a useful starting point for the design of non-symmetric systems [18].

Equivalently, $f(K)$ can be written as

$$\begin{aligned} f(K) \;&=\; \frac{1}{2}\,\mathrm{trace}\left((K\,-\,A)^{-1}(Q\,+\,A\,R\,A)\right)\,+ \\ &\qquad \frac{1}{2}\,\mathrm{trace}\,(R\,K)\,+\,\frac{1}{2}\,\mathrm{trace}\,(R\,A) \\ &=\; \frac{1}{2}\sum_{i\,=\,1}^{n} f_i(K) \end{aligned}$$

where

$$\begin{aligned} f_i(K) \;&=\; q_i^T(K\,-\,A)^{-1}q_i\,+\,a_i^T(K\,-\,A)^{-1}a_i\,+ \\ &\qquad r_i^T(K\,+\,A)\,r_i. \end{aligned}$$

The vectors indexed by i in this expression are given by

$$q_i \;=\; Q^{1/2}\,\mathrm{e}_i,\quad r_i \;=\; R^{1/2}\,\mathrm{e}_i,\quad a_i \;=\; A\,r_i$$

where e_i is the ith canonical basis vector in $\mathbb{R}^n$. Furthermore, the gradient of f_i with respect to K is given by

$$\begin{aligned} \nabla f_i(K) \;&=\; r_i\,r_i^T\,-\,(K\,-\,A)^{-1}q_i\,q_i^T(K\,-\,A)^{-1}\,- \\ &\qquad (K\,-\,A)^{-1}a_i\,a_i^T(K\,-\,A)^{-1} \end{aligned}$$

Thus, there is no need to compute and store the inverse of the matrix $K - A$ in order to evaluate $f_i(K)$ and $\nabla f_i(K)$; only actions of $(K - A)^{-1}$ on the vectors q_i and a_i is necessary. For example, the preconditioned conjugate gradients method can be used to compute them efficiently.

Remark 2: The vector $(K - A)^{-1}q_i$ represents the steady-state solution of a stable linear system

$$\dot{\psi}_i \;=\; (A\,-\,K)\,\psi_i\,+\,q_i$$

and it can be computed via numerical integration. This illustrates that the symmetric nature of the closed-loop system (1) allows us to avoid the need for numerical integration of the corresponding adjoint system.

2) Undirected consensus networks: For undirected consensus networks with n nodes, the matrices A and $K\in\mathbb{R}^{n\times n}$ in (1) determine the graph Laplacians of the plant and controller networks, respectively. In the absence of exogenous disturbances, the network converges to the average of the initial node values $\bar{\psi} \;=\; (1/n)\sum_i \psi_i(0)$ if and only if it is connected [19]. Let $B_1 \;=\; B_2 \;=\; I$ and let $Q \;:=\; I\,-\,(1/n)\mathbb{1}\mathbb{1}^T$ penalize the deviation of individual node values from average. The objective is to minimize the mean square deviation from the network average by adding a few additional edges, specified by the graph Laplacian K of a controller network. If E is the incidence matrix of the controller graph, K can be written as

$$K(x) \;=\; E\,\mathrm{diag}\,(x)\,E^T$$

where $\mathrm{diag}\,(x)$ is a diagonal matrix containing the optimization variable $x \in \mathbb{R}^m$ (i.e., the vector of the edge weights in the controller graph). Regularization terms may be used to promote sparsity of the controller network or to impose some additional constraints on the edge weights.

As shown in [7], up to an additive constant, the square of the $\mathcal{H}_2$ norm (from d to ζ) is determined by

$$\begin{aligned} f(x) \;=\; &\mathrm{trace}\left((E\,\mathrm{diag}\,(x)\,E^T\,-\,A)^{\dagger}(I\,+\,A\,R\,A)\right)\,+ \\ &\mathrm{diag}\left(E^T R\,E\right)^T x \end{aligned}$$

where the pseudo-inverse of the closed-loop graph Laplacian is given by

$$(E\,\mathrm{diag}\,(x)\,E^T - A)^{\dagger} = (E\,\mathrm{diag}\,(x)\,E^T + (1/n)\,\mathbb{1}\mathbb{1}^T - A)^{-1}.$$

It is easy to show that $f(x)$ can be written as

$$f(x) \;=\; \sum_{i\,=\,1}^{n} f_i(x)$$

where

$$\begin{aligned} f_i(x) \;&=\; \xi_i^T\left(E\,\mathrm{diag}\,(x)\,E^T + (1/n)\,\mathbb{1}\mathbb{1}^T - A\right)^{-1}\xi_i\,+ \\ &\qquad (1/n)\,\mathrm{diag}\left(E^T R\,E\right)^T x. \end{aligned} \tag{6}$$

Here, $\xi_i = (I + A\,R\,A)^{1/2}\,\mathrm{e}_i$ is the ith column of the square root of the matrix $(I + A\,R\,A)$. Moreover, it can be shown that the gradient of $f_i(x)$ is given by

$$\nabla f_i(x) \;=\; (1/n)\,\mathrm{diag}\left(E^T R\,E\right) \,-\, \nu_i(x)\circ\nu_i(x) \tag{7}$$

where $\circ$ is the elementwise multiplication and

$$\nu_i(x) \;=\; E^T\left(E\,\mathrm{diag}\,(x)\,E^T + (1/n)\,\mathbb{1}\mathbb{1}^T - A\right)^{-1}\xi_i. \tag{8}$$

As for symmetric systems, there is no need to compute and

store the inverse of the matrix in (8) and the preconditioned conjugate gradients method can be used to compute the gradient efficiently.

We next provide a proposition on the Lipschitz continuity of the gradient of f_i for connected resistive networks. In such networks, the plant graph is connected and all edge weights in both the plant and the controller graphs are non-negative.

Proposition 1: The gradient of f_i for connected resistive networks is Lipschitz continuous with the Lipschitz constant

$$L_i \;=\; \xi_i^T\, \hat{A}(0)^{-1}\, E\, E^T \hat{A}(0)^{-1}\, \xi_i\, \|E^T \hat{A}(0)^{-1}\, E\|_2 \quad (9)$$

where $\hat{A}(x) = E\,\mathrm{diag}\,(x)\,E^T + (1/n)\,\mathbb{1}\mathbb{1}^T - A$.

Proof: For a convex and twice differentiable function f_i, the gradient ∇f_i is Lipschitz continuous with Lipschitz constant L_i, if

$$\nabla^2 f_i(x) \;\preceq\; L_i\, I$$

or, equivalently, $z^T\, \nabla^2 f_i(x)\, z \le L_i\, \|z\|^2$. It can be shown that the second order derivative of f_i is given by

$$\nabla^2 f_i(x) \;=\; (\nu_i\, \nu_i^T) \circ (E^T\, \hat{A}(x)^{-1}\, E).$$

Thus, $z^T\, \nabla^2 f_i(x)\, z$ can be written as

$$\xi_i^T\, \hat{A}(x)^{-1}\, E\, D_z\, E^T\, \hat{A}(x)^{-1} E\, D_z\, E^T\, \hat{A}(x)^{-1} \xi_i$$

where $D_z := \mathrm{diag}(z)$. For positive definite matrices S and T with $S \preceq T$, we have $z^T S\, z \le z^T T\, z$ and

$$z^T S\, z \le \lambda_{\max}(S)\, \|z\|^2 \le \sqrt{\lambda_{\max}(S^T S)}\, \|z\|^2 = \|S\|_2\, \|z\|^2$$

for any vector z. Therefore,

$$D_z\, E^T\, \hat{A}(x)^{-1} E\, D_z \;\preceq\; \|E^T \hat{A}(0)^{-1}\, E\|_2\, \|z\|^2.$$

As a result,

$$z^T\, \nabla^2 f_i(x)\, z \;\le\; \nu_i(0)^T \nu_i(0)\, \|E^T \hat{A}(0)^{-1}\, E\|_2\, \|z\|^2$$

where $\nu_i(0) = E^T \hat{A}(0)^{-1}\, \xi_i$. ∎

III. An ADMM-based consensus algorithm

We next demonstrate that a standard consensus algorithm based on the Alternating Direction Method of Multipliers (ADMM) is well-suited for distributed design of optimal structured feedback gains. As shown in Section II, the regularized optimal $\mathcal{H}_2$ problem can be written as

$$\underset{K}{\text{minimize}} \quad \sum_{i=1}^{N} f_i(K) \;+\; g(K). \quad (10)$$

This characterization is suitable for distributed implementation in which each processors solves an optimization problem. By introducing N local variables K_i and a global variable K_0, problem (10) can be brought into a standard consensus form,

$$\begin{aligned} \underset{K_i,\, K_0}{\text{minimize}} \quad & \sum_{i=1}^{N} f_i(K_i) \;+\; g(K_0) \\ \text{subject to} \quad & K_i \;-\; K_0 \;=\; 0, \quad i \;=\; 1, \ldots, N. \end{aligned} \quad (11)$$

This formulation increases the number of optimization variables but it brings the objective function into a separable form and facilitates distributed computations. When implemented on a single machine, solving the reformulated problem is not necessarily more computationally efficient than solving the original problem using centralized algorithms. However, this formulation allows to work with a local objective function f_i that depends on a single local variable K_i. This is advantageous for large-scale systems where centralized algorithms cannot be afforded (e.g., because of the computational cost associated with solving the large-scale Lyapunov equations).

The augmented Lagrangian associated with (11) is

$$\begin{aligned} &\mathcal{L}(K_i, K_0; \Lambda) \;=\; g(K_0) \;+ \\ &\sum_{i=1}^{N} (\, f_i(K_i) \;+\; \langle \Lambda_i, K_i - K_0 \rangle \;+\; \frac{\rho_i}{2}\, \|K_i - K_0\|_F^2) \end{aligned} \quad (12)$$

where Λ_i's are the Lagrange multipliers and ρ_i's are positive parameters. The ADMM algorithm consists of the following iterative steps,

$$\begin{aligned} K_i^{k+1} \;&=\; \underset{K_i}{\mathrm{argmin}}\; f_i(K_i) \;+\; \frac{\rho_i}{2}\, \|K_i \;-\; U_i^k\|_F^2 \\ K_0^{k+1} \;&=\; \underset{K_0}{\mathrm{argmin}}\; g(K_0) \;+\; \sum_{i=1}^{N} \frac{\rho_i}{2}\, \|K_0 \;-\; V_i^k\|_F^2 \\ \Lambda_i^{k+1} \;&=\; \Lambda_i^k \;+\; \rho_i \left(K_i^{k+1} \;-\; K_0^{k+1}\right) \end{aligned}$$

where

$$\begin{aligned} U_i^k \;&:=\; K_0^k \;-\; (1/\rho_i)\, \Lambda_i^k \\ V_i^k \;&:=\; K_i^{k+1} \;+\; (1/\rho_i)\, \Lambda_i^k. \end{aligned}$$

The K_i-minimization step can be done via distributed computation by spreading subproblems to N different processors. On the other hand, the update of K_0 amounts to the evaluation of the proximal operator of the function g,

$$K_0^{k+1} \;=\; \mathbf{prox}_{g/(N\bar{\rho})} \left(\frac{1}{N\bar{\rho}} \sum_{i\,=\,1}^{N} \rho_i V_i^k \right).$$

where $\bar{\rho} := (1/N) \sum_i \rho_i$. Thus, the update of K_0 requires gathering each K_i^{k+1} and the associated Lagrange multipliers Λ_i^k in order to form V_i^k.

Remark 3: The above presented consensus algorithm is standard (e.g., see [12]) but to the best of our knowledge it has not been previously used for optimal design of structured feedback gains via distributed optimization. Recently, convergence of this algorithm was established even for problems with non-convex objective functions f_i [11]. The authors of [11] also show that additional computational advantage can be gained by updating only a subset of K_i's in each iteration and offer several alternative implementations to speed computations.

Remark 4: The update of each K_i amounts to the computation of the proximal operator associated with f_i,

$$K_i^{k+1} \;=\; \mathbf{prox}_{f_i/\rho_i}\big(U_i^k\big)$$

However, while the proximal operators of commonly used

regularization functions are easy to evaluate, the computation of $\mathbf{prox}_{f_i/\rho_i}$ is significantly more involved. In [2], the proximal operator associated with the closed-loop $\mathcal{H}_2$ norm $f(K)$ was computed using the Anderson-Moore algorithm. This approach requires computation of the closed-loop controllability and observability gramians. In contrast, the developments of Section II facilitate computation of $\mathbf{prox}_{f_i/\rho_i}$ via a proximal gradient algorithm that avoids the need for solving large-scale Lyapunov equations.

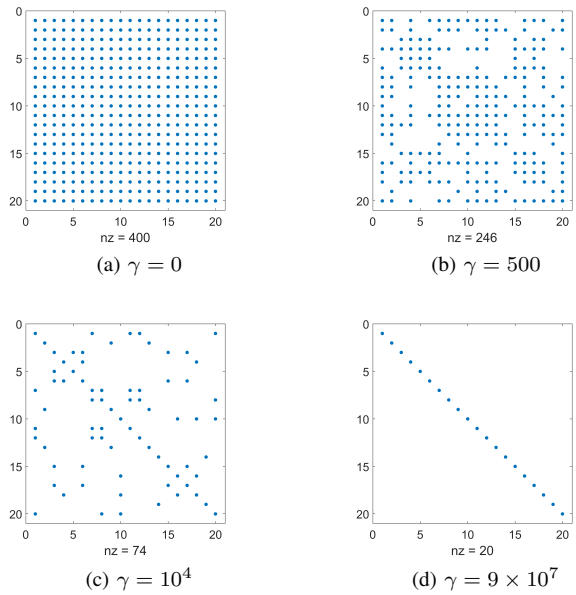
Remark 5: For the problem of growing connected resistive consensus networks [7], Algorithm 3 in [11] can be used to solve the K_i-minimization step in the ADMM-based algorithm explicitly. The key point of departure compared to the standard ADMM implementation is linearization in the K_i-minimization step of the function f_i with a Lipschitz continuous gradient around the current K_0 iterate. This offers a significant speed-up and enables an explicit update of K_i.

IV. Computational experiments

In this section, we employ the ADMM-based consensus algorithm to design optimal structured feedback gains. This algorithm is implemented in a distributed fashion by splitting the problem into N separate subproblems over N different cores. We have provided a parallel implementation in C++ using pthreads library and executed tests on a machine featuring an Intel Core i7-3770 with 16GB of RAM to measure the performance of the algorithm.

We first employ the algorithm to design a sparse optimal feedback controller for a symmetric system. We randomly generate a symmetric matrix A with $n = 20$. We set the control weight $R = I$ and the state weight matrix $Q = I$. Our goal is to find a sparse optimal controller that minimizes the $\mathcal{H}_2$ norm of the closed-loop system. The regularization function is $g(K) = \gamma \|K\|_1$ where γ is a sparsity-promoting parameter and the ℓ_1 norm is a proxy for inducing sparsity in the controller matrix. We start with $\gamma = 0$ and gradually increase the value of γ. Figure 1 shows how the sparsity pattern of the controller changes by increasing the value of γ. For $\gamma = 0$, the optimal feedback controller is given by a full matrix and as γ increases the controller becomes sparser. In particular, for $\gamma = 9 \times 10^7$, the controller is diagonal and by further increasing the value of γ, it remains diagonal to guarantee stability of the closed-loop system.

Next, we use our algorithm for growing a connected resistive network with $n = 20$ nodes. The plant graph is given by an Erdös-Rényi network with edge probability $1.05\log(n)/n$. We choose the control weight matrix $R = I$ and a state weight matrix that penalizes the mean-square deviation from the network average, $Q = I - (1/n)\,\mathbb{1}\mathbb{1}^T$. Moreover, the incidence matrix of the controller is such that there are no joint edges between the plant and the controller graphs. As discussed in Section II-B.2, for consensus networks, the controller can be written as a function of the vector of the edge weights x. Thus, the $\mathcal{H}_2$ norm of the closed-loop system is $f(x)$ and the regularization function is given by $g(x) = \gamma \|w \circ x\|_1$ where w is the vector of the



(a) $\gamma = 0$ (b) $\gamma = 500$

(c) $\gamma = 10^4$ (d) $\gamma = 9 \times 10^7$

Fig. 1: Topology of the controller graphs for a randomly generated symmetric system with $n = 20$ states.

weights. Since the plant network is resistive and connected, all the edge weights are nonnegative, thereby if the added edges have nonnegative edge weights, the closed-loop system is stable. We can write the smooth and nonsmooth parts of the objective function as $f(x) + \gamma\, w^T x$ and $g(x) = I_+(x)$ where $I_+(x)$ is an indicator function for nonnegative orthant.

We solve the problem (10) to find the controller graph for 500 logarithmically-spaced values of $\gamma \in [0.001, 0.3]$ using the path-following iterative reweighted algorithm as a proxy for inducing sparsity [20]. We set the weights to be inversely proportional to the magnitude of the solution x at the previous value of γ following by a polishing step that computes the optimal weights of identified edges; see [7].

As γ increases, the number of nonzero edges decreases and the closed-loop performance deteriorates. As shown in Fig. 2, relative to the optimal centralized vector of the edge weights, x_c, the $\mathcal{H}_2$ loss decreases as the sparsity of the vector of the edge weights x increases. In particular, for $\gamma = 0.3$, there is only one nonzero element in the vector of the edge weights. The identified sparse controller in this case uses only 0.62% of the edges, relative to the optimal centralized controller, i.e., $\mathbf{card}(x)/\mathbf{card}(x_c) = 0.62\%$ and achieves a performance loss of 23.47%, i.e., $(J - J_c)/J_c = 23.47\%$.

Next, we employ our algorithm for growing connected resistive networks with different number of nodes using multiple cores. The solve times are averaged over 10 trials and the speedup relative to a single core is displayed in Fig 3. This figure demonstrates that the algorithm is scalable. In particular, multi-core execution outperforms running just on a single core. The speed-up is even higher for larger networks since overheads of parallel execution are smaller and more time is spent on actual parallel computation.

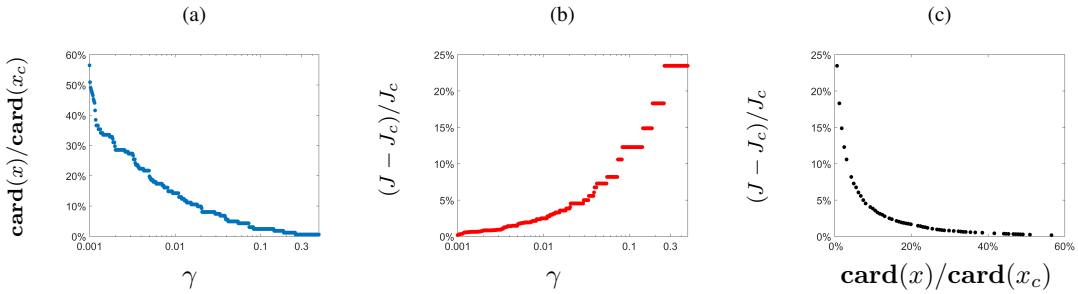


Fig. 2: (a) Sparsity level; (b) performance degradation; and (c) the optimal tradeoff between the performance degradation and the sparsity level of the optimal sparse x compared to the optimal centralized controller x_c. The results are obtained for a randomly generated Erdös-Rényi network with $n = 20$ nodes.

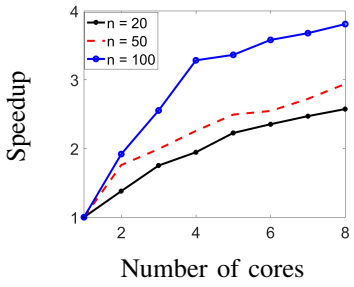


Fig. 3: Speedup ratio versus the number of the cores used for growing connected resistive networks with n nodes.

V. Concluding Remarks

We have considered the problem of designing optimal structured feedback controllers for large-scale systems. By exploiting the structure of the $\mathcal{H}_2$ norm, we have shown that this problem can be separated into N subproblems and that it can be solved efficiently using distributed optimization. We have utilized an ADMM-based consensus algorithm to design a sparse controller network that improves the performance of the closed-loop system. By splitting the problem into N separate subproblems over N different cores, we have implemented this algorithm in C++. Our parallel implementation can be used to solve structured optimal control problems for large-scale systems, e.g., power networks [21], [22]. Computational experiments are provided to demonstrate the utility of the developed approach.

References

[1] 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.

[2] 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.

[3] S. Schuler, P. Li, J. Lam, and F. Allgöwer, "Design of structured dynamic output-feedback controllers for interconnected systems," *International Journal of Control*, vol. 84, no. 12, pp. 2081–2091, 2011.

[4] N. Matni and V. Chandrasekaran, "Regularization for design," *IEEE Trans. Automat. Control*, vol. 61, no. 12, pp. 3991–4006, 2016.

[5] M. R. Jovanović and N. K. Dhingra, "Controller architectures: tradeoffs between performance and structure," *Eur. J. Control*, vol. 30, pp. 76–91, July 2016.

[6] S. Hassan-Moghaddam, X. Wu, and M. R. Jovanović, "Edge addition in directed consensus networks," in *Proceedings of the 2017 American Control Conference*, Seattle, WA, 2017, pp. 5592–5597.

[7] S. Hassan-Moghaddam and M. R. Jovanović, "Topology design for stochastically-forced consensus networks," *IEEE Trans. Control Netw. Syst.*, 2017, doi:10.1109/TCNS.2017.2674962.

[8] C. Conte, C. N. Jones, M. Morari, and M. N. Zeilinger, "Distributed synthesis and stability of cooperative distributed model predictive control for linear systems," *Automatica*, vol. 69, pp. 117–125, 2016.

[9] K. Mårtensson and A. Rantzer, "Gradient methods for iterative distributed control synthesis," in *Proceedings of the 48th IEEE Conference on Decision and Control*, 2009, pp. 549–554.

[10] K. Mårtensson and A. Rantzer, "A scalable method for continuous-time distributed control synthesis," in *Proceedings of the 2012 American Control Conference*, 2012, pp. 6308–6313.

[11] M. Hong, Z. Q. Luo, and M. Razaviyayn, "Convergence analysis of alternating direction method of multipliers for a family of nonconvex problems," *SIAM J. Optimiz.*, vol. 26, no. 1, pp. 337–364, 2016.

[12] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundation and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–124, 2011.

[13] X. Wu and M. R. Jovanović, "Sparsity-promoting optimal control of systems with symmetries, consensus and synchronization networks," *Syst. Control Lett.*, vol. 103, pp. 1–8, May 2017.

[14] F. Lin, M. Fardad, and M. R. Jovanović, "Augmented Lagrangian approach to design of structured optimal state feedback gains," *IEEE Trans. Automat. Control*, vol. 56, no. 12, pp. 2923–2929, 2011.

[15] T. Rautert and E. W. Sachs, "Computational design of optimal output feedback controllers," *SIAM J. Optim.*, vol. 7, no. 3, pp. 837–852, 1997.

[16] C. V. Loan, "Computing integrals involving the matrix exponential," *IEEE Trans. Automat. Control*, vol. 23, no. 3, pp. 395–404, 1978.

[17] P. Benner, J.-R. Li, and T. Penzl, "Numerical solution of large-scale Lyapunov equations, Riccati equations, and linear-quadratic optimal control problems," *Numerical Linear Algebra with Applications*, vol. 15, no. 9, pp. 755–777, 2008.

[18] N. K. Dhingra and M. R. Jovanović, "Convex synthesis of symmetric modifications to linear systems," in *Proceedings of the 2015 American Control Conference*, Chicago, IL, 2015, pp. 3583–3588.

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

[20] 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.

[21] F. Dörfler, M. R. Jovanović, M. Chertkov, and F. Bullo, "Sparsity-promoting optimal wide-area control of power networks," *IEEE Trans. Power Syst.*, vol. 29, no. 5, pp. 2281–2291, September 2014.

[22] X. Wu, F. Dörfler, and M. R. Jovanović, "Input-output analysis and decentralized optimal control of inter-area oscillations in power systems," *IEEE Trans. Power Syst.*, vol. 31, no. 3, pp. 2434–2444, May 2016.