2019 American Control Conference (ACC)
Philadelphia, PA, USA, July 10-12, 2019

Data-driven proximal algorithms for the design of structured optimal feedback gains

Sepideh Hassan-Moghaddam, Mihailo R. Jovanović, and Sean Meyn

***Abstract*— Distributed feedback design and complexity constrained control are examples of problems posed within the domain of structured optimal feedback synthesis. The optimal feedback gain is typically a non-convex function of system primitives. However, in recent years, algorithms have been proposed to obtain locally optimal solutions. In applications to large-scale distributed control, the major obstacle is computational complexity. This paper addresses complexity through a combination of linear-algebraic techniques and computational methods adapted from both machine learning and reinforcement learning. It is shown that for general classes of optimal control problems, the objective function and its gradient can be computed from data. Transformations borrowed from the theory of reinforcement learning are adapted to obtain simulation-based algorithms for computing the structured optimal H_2 feedback gain. Customized proximal algorithms based on gradient descent and incremental gradient are tested in computational experiments and their relative merits are discussed.**

***Index Terms*— Data-driven feedback design, large-scale systems, optimization, proximal algorithms, reinforcement learning, structured and sparsity-promoting optimal control.**

I. Introduction

Large-scale dynamical systems that combine sensing, computing, and communication devices are ubiquitous in modern technology. Systems of this type arise in applications ranging from distributed power generation, to coordination of autonomous vehicles, to control of fluid flows around wind turbines, to design of combination drug therapies for HIV and cancer treatments. One of the major challenges is the development of fast and scalable methods for their analysis and design. Such systems involve large-scale interconnections of components, have rapidly-evolving structure and limitations on communication/processing power, and require real-time distributed control actions. These requirements make control strategies that rely on centralized information processing infeasible and motivate new classes of optimal control problems. In these, standard performance metrics are augmented with typically nonsmooth regularizers to promote desired structural features (e.g., low communication requirements) in the optimal controller [1]–[5]. Moreover, in many applications, the dynamics of the plant are unknown and only limited input-output measurements are available. Designing optimal controllers for these systems is challenging even in the absence of structural constraints. The LQR problem in the model-free setup has been recently studied in [6], [7].

Structured optimal control and inverse problems, that arise when trying to identify and control dynamical representations of rapidly evolving systems in real-time, typically lead to optimization of functionals consisting of a sum of a smooth term and a nonsmooth regularizer. Such problems are of increasing importance in control and it is thus necessary to develop efficient and scalable algorithms for structured optimal control synthesis. This paper addresses complexity through a combination of linear-algebraic techniques and computational methods adapted from both machine learning and reinforcement learning. We approach the structured optimal control problem via a data-driven framework that does not require knowledge of the dynamical generator and avoids the need to solve large-scale matrical equations. For the structured optimal H_2 state-feedback problem, we show that the objective function and its gradient can be computed from data and develop customized proximal algorithms based on gradient descent and incremental gradient method.

Our presentation is organized as follows. In Section II, we describe the regularized structured H_2 optimal control problem. In Section III, a square-additive property of the H_2 norm is used to obtain a decomposition of the objective function and its gradient and duality arguments are utilized to transform these into quantities that can be estimated from simulations. In Section IV, proximal algorithms of tractable complexity for solving the structured optimal control problem are described and computational experiments are provided. In Section V, the paper is concluded with remarks and a summary of outstanding challenges.

II. Problem formulation

Consider the LTI control system in state-space form

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

where $\psi(t) \in \mathbb{R}^n$ is the state, $u(t) \in \mathbb{R}^m$ is the control input, and $\zeta(t) \in \mathbb{R}^{n+m}$ is the performance output. The matrices are all of compatible dimensions, and the standard assumptions are imposed: $Q \succeq 0$, $R \succ 0$, (A, B) is stabilizable, and $(A, Q^{1/2})$ is detectable.

The control input is defined by state-feedback,

$$u(t) = -K\,\psi(t)$$

Financial support from the National Science Foundation ECCS-1809833, Army Research Office W911NF1810334, and Air Force Office of Scientific Research FA9550-16-1-0009 is gratefully acknowledged.

S. Hassan-Moghaddam and M. R. Jovanović are with the Ming Hsieh Department of Electrical and Computer Engineering, University of Southern California, Los Angeles, CA 90089. S. Meyn is with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611. E-mails: hassanmo@usc.edu, mihailo@usc.edu, meyn@ece.ufl.edu.

with gain matrix $K \in \mathbb{R}^{m\times n}$. The closed-loop system is thus

$$\begin{aligned} \dot{\psi} &= A_{\rm cl}\,\psi \\ \zeta &= \left[\begin{array}{c} Q^{1/2} \\ -R^{1/2}K \end{array}\right]\psi \end{aligned} \tag{2}$$

where $A_{\rm cl} := A - BK$.

Closed-loop performance is quantified by the square of the L_2 norm of the impulse response for the closed-loop system, which is expressed using either of the following expressions:

$$\begin{aligned} f(K) &= \text{trace}\,(P) \\ &= \text{trace}\left((Q \,+\, K^T R\,K)\,L\right) \end{aligned} \tag{3}$$

where $P \succ 0$ is the closed-loop observability Gramian,

$$A_{\rm cl}^T P \,+\, P\,A_{\rm cl} \,+\, Q \,+\, K^T R\,K \,=\, 0 \tag{4}$$

and $L \succ 0$ is the closed-loop controllability Gramian,

$$A_{\rm cl}\,L \,+\, L\,A_{\rm cl}^T \,+\, I \,=\, 0. \tag{5}$$

Our objective is to promote certain structural properties of the feedback gain matrix K by solving the regularized optimal control problem

$$\underset{K}{\text{minimize}} \quad f(K) \,+\, g(K) \tag{6}$$

where f is given by (3) and g is the regularization function. When it is desired to design K to belong to a set $\mathcal{S}$ with a specified pattern of zero elements [1], g is an indicator function associated with the underlying sparsity constraints,

$$g(K) \,:=\, \left\{\begin{array}{ll} 0 & K \in \mathcal{S} \\ \infty & K \notin \mathcal{S}. \end{array}\right.$$

Similarly, the ℓ_1 norm and the nuclear norm are commonly used convex proxies for promoting sparsity, or for designing K with low rank. In this case, $g(K) \,=\, \gamma \sum_{i,\,j} |K_{ij}|$ or $g(K) \,=\, \gamma \|K\|_* \,=\, \gamma \sum_i \sigma_i(K)$, where γ is a positive regularization parameter and σ_i is the ith singular value.

In general, the optimization problem (6) is not convex because f is a non-convex function of K. The proximal augmented Lagrangian method [8] can be used to compute a local minimum of (6); for details, see [9]. The minimization (6) remains non-convex even in the absence of the regularization function g. Of course, in this special case it reduces to the standard LQR problem, and the globally optimal solution is given by

$$K^\star \,=\, R^{-1}B^T P$$

where P is the unique positive definite solution of the algebraic Riccati equation,

$$A^T P \,+\, PA \,+\, Q \,-\, P\,B\,R^{-1}B^T P \,=\, 0.$$

In spite of the lack of convexity, conditions for convergence of gradient descent methods for the standard LQR problem to the global minimum for discrete- and continuous-time systems were recently established in [7] and [10], respectively.

The goal of this paper is twofold: first is to obtain algorithms of tractable complexity for truly large-scale problems. A second goal is to solve the regularized, structured optimal control problem without knowledge of the underlying model. The approach to the second goal is a component of our approach to the first: a data-driven framework is proposed for computing the optimal feedback gain, with unknown dynamic matrix A. This approach also avoids the need to solve large-scale Lyapunov equations.

III. Computation from data

A square-additive property of the $\mathcal{H}_2$ norm is used to obtain a decomposition of the objective function f and its gradient ∇f. Duality arguments are then employed to transform each term into a quantity that can be estimated from simulations. The resulting algorithm obtains estimates of the optimal gain without solving a Lyapunov equation, and without knowledge of the matrix A.

The gradient of f with respect to K is given by [11],

$$\nabla f(K) \,=\, 2\,(R\,K \,-\, B^T P)\,L. \tag{7}$$

In what follows, we use the three decompositions

$$L \,=\, \sum_{k\,=\,1}^{n} L^k,\;\; f(K) \,=\, \sum_{k\,=\,1}^{n} f_k,\;\; \nabla f(K) \,=\, \sum_{k\,=\,1}^{n} \nabla f_k$$

where $L^k \succeq 0$ solves the Lyapunov equation

$$A_{\rm cl}\,L^k \,+\, L^k\,A_{\rm cl}^T \,+\, {\rm e}_k {\rm e}_k^T \,=\, 0, \tag{8}$$

${\rm e}_k$ is the kth unit vector in the canonical basis of $\mathbb{R}^n$, and

$$\begin{aligned} f_k(K) &= \text{trace}\left((Q \,+\, K^T R\,K)\,L^k\right) \\ \nabla f_k(K) &= 2\,(R\,K \,-\, B^T P)\,L^k. \end{aligned} \tag{9}$$

A. Computation of P and L from data

The solution to (8) can be expressed

$$\begin{aligned} L^k &= \int_0^\infty {\rm e}^{A_{\rm cl}t}\,{\rm e}_k {\rm e}_k^T\,{\rm e}^{A_{\rm cl}^T t}\,{\rm d}t \\ &= \int_0^\infty \psi^k(t)\,(\psi^k(t))^T\,{\rm d}t \end{aligned} \tag{10}$$

where $\psi^k(t) := {\rm e}^{A_{\rm cl}t}\,{\rm e}_k$ is the solution to

$$\frac{{\rm d}\psi^k}{{\rm d}t} \,=\, A_{\rm cl}\,\psi^k,\;\; \psi^k(0) \,=\, {\rm e}_k,\;\; k \,=\, 1,\ldots,n. \tag{11}$$

This leads to the first component of the data-driven architecture: for a given feedback gain K, the term KL^k that appears both in f and ∇f can be expressed as the integral

$$KL^k \,=\, \int_0^\infty u^k(t)\,(\psi^k(t))^T\,{\rm d}t \tag{12}$$

where $\psi^k(t)$ and $u^k(t) = K\psi^k(t)$ are obtained from simulations of system (11).

Similarly, the observability Gramian is given by

$$P \,=\, \int_0^\infty {\rm e}^{A_{\rm cl}^T t}\,(Q \,+\, K^T RK)\,{\rm e}^{A_{\rm cl}t}\,{\rm d}t$$

and its (i,j)th component is

$$P_{i,j} \;=\; \int_0^\infty (\psi^i(t))^T (Q \,+\, K^T R K)\, \psi^j(t)\, \mathrm{d}t \qquad (13)$$

where ψ^i and ψ^j are obtained from simulations of system (11) with the initial conditions e_i and e_j, respectively.

Thus, n forward-in-time simulations of (11) can be used to compute integrals (10), (12), and (13) and evaluate $\nabla f(K)$ without solving a Lyapunov equation. Overall, there are $n(n+5)/2$ integrals that involve either outer vector product or matrix-vector multiplication. Computation complexity can be further reduced by avoiding multiplication of P and L^k in (9). A data-driven approach to compute the product PL^k directly is introduced next.

B. Computation of PL from data

As showed in [12], [13], the product PL^k that appears in $\nabla f_k(K)$ can be expressed as

$$X^k \;:=\; PL^k \;=\; -\int_0^\infty \phi^k(t)\, (\psi^k(t))^T\, \mathrm{d}t \qquad (14)$$

where ϕ^k is obtained from the solution of the adjoint system

$$\begin{aligned} \dot{\phi}^k \;&=\; -\,A_{\mathrm{cl}}^T\, \phi^k \;+\; (Q \,+\, K^T R\, K)\, \psi^k \\ \phi^k(\infty) \;&=\; 0 \end{aligned} \qquad (15)$$

and ψ^k is the solution to (11). Thus, numerical simulations of the primal and adjoint systems (11) and (15) along with numerical evaluations of the corresponding integrals can be used to compute L^k, PL^k, f_k, and ∇f_k.

Next, we show that the matrix X^k in (14) can be computed without simulating the adjoint system (15). This is a critically important step in the model-free setup. Introduce the new variable

$$\eta^k \;:=\; (Q \,+\, K^T R\, K)\, \psi^k$$

so that (15) becomes

$$\dot{\phi}^k \;=\; -\,A_{\mathrm{cl}}^T\, \phi^k \;+\; \eta^k, \quad \phi^k(\infty) \;=\; 0 \qquad (16)$$

which admits the solution,

$$\begin{aligned} \phi^k(t) \;&=\; \left[H\eta^k\right](t) \\ &=\; -\int_t^\infty \mathrm{e}^{-A_{\mathrm{cl}}^T(t\,-\,\tau)}\, \eta^k(\tau)\, \mathrm{d}\tau. \end{aligned} \qquad (17)$$

The linear operator H is introduced here to facilitate an adjoint transformation below. This operator and its adjoint H^* are defined defined on $L_2([0,\infty)) \to \mathbb{R}^n$. For any functions $\xi, \chi \in L_2([0,\infty))$, we have by definition $\langle \chi, H\xi\rangle = \langle H^*\chi, \xi\rangle$; see, e.g., [14]. An explicit representation for $\mu = H^*\gamma$ is obtained using elementary calculus:

$$\mu(t) = \int_0^t \mathrm{e}^{A_{\mathrm{cl}}(t\,-\,\tau)}\gamma(\tau)\, \mathrm{d}\tau.$$

The (i,j)th element of the matrix X^k can be expressed

$$\begin{aligned} X^k_{i,j} \;:=\; \mathrm{e}_i^T X_k\, \mathrm{e}_j \;&=\; -\int_0^\infty \phi_i^k(t)\, \psi_j^k(t)\, \mathrm{d}t \\ &=\; -\left\langle \psi_j^k, \phi_i^k \right\rangle \end{aligned} \qquad (18)$$

where $\phi_i^k(t) := \mathrm{e}_i^T \phi^k(t)$ and $\psi_j^k(t) := \mathrm{e}_j^T \psi^k(t)$ are the ith and jth elements of the vectors $\phi^k(t)$ and $\psi^k(t)$, respectively, and the inner product is in $L_2([0,\infty))$. By substituting the expression (17) for ϕ^k in (18), we obtain

$$\begin{aligned} X^k_{i,j} \;&=\; -\left\langle \psi_j^k, \mathrm{e}_i^T H\eta^k \right\rangle \\ &=\; -\left\langle H^*\mathrm{e}_i\, \psi_j^k, \eta^k \right\rangle \\ &=:\; \left\langle \xi^k_{i,j}, \eta^k \right\rangle. \end{aligned} \qquad (19)$$

Applying the expression for H^* then gives

$$\begin{aligned} \xi^k_{i,j}(t) \;&:=\; -\left[H^*\mathrm{e}_i\mathrm{e}_j^T\psi^k\right](t) \\ &=\; \int_0^t \mathrm{e}^{A_{\mathrm{cl}}(t\,-\,\tau)}\, \mathrm{e}_i\, \mathrm{e}_j^T \psi^k(\tau)\, \mathrm{d}\tau. \end{aligned}$$

This implies that $\xi^k_{i,j}$ can be obtained as the solution to the LTI system,

$$\frac{\mathrm{d}\xi^k_{i,j}}{\mathrm{d}t} \;=\; A_{\mathrm{cl}}\, \xi^k_{i,j} \;+\; \mathrm{e}_i\, \mathrm{e}_j^T \psi^k, \qquad \xi^k_{i,j}(0) \;=\; 0. \qquad (20)$$

Thus, to compute $\nabla f(K)$ for a given gain K, the only systems that need to be simulated are the forward in time systems (11) and (20). The system (11) is unforced, and its solution determines the input to the forced system (20). To compute the n^2 elements of the matrix X^k via (19), n simulations of (11), n^2 simulations of (20), and n^2 inner products between $\xi^k_{i,j}$ and η^k are required. Moreover, a matrix-vector multiplication is needed to compute each η^k. Relative to Section III-A, the number of simulations has increased but the computation of integral (13) and the matrix multiplication of P and L^k have been avoided.

Remark 1: Similar adjoint techniques are used in analysis of reinforcement learning. The proof that the TD(1) algorithm solves a minimum norm problem is based on related adjoint transformations [15], [16]. And, a similar adjoint transformation is a crucial step in a Q-learning algorithm for deterministic continuous-time systems [17].

Remark 2: By writing the Lyapunov equation for the controllability Gramian as (8), unlike the requirement of stochastic simulations in [12], only deterministic simulations are needed in the present framework.

Remark 3: Since the optimal unstructured gain $K^\star$ only depends on the closed-loop observability Gramian P (which can be obtained from the solution of the algebraic Riccati equation), L does not influence $K^\star$ and its computation can be avoided. To promote structure (e.g., via proximal algorithms) we need to compute both P and L to form ∇f. In contrast, only computation of P is required to form natural policy gradient, which is defined as [18], [19],

$$\nabla h(K) \;=\; \nabla f(K)\, L^{-1} \;=\; 2\,(R\,K \,-\, B^T P). \qquad (21)$$

IV. Proximal algorithms

Algorithms of tractable complexity are introduced here for solving the regularized structured optimal control problem (6). A standard proximal gradient algorithm is considered as a starting point. It is argued however that computing the full gradient may be prohibitively expensive. Thus, we

propose a more efficient algorithm based on incremental proximal gradient that is well-suited for the design of structured optimal feedback gains in large-scale systems.

A. Proximal gradient algorithm

The proximal gradient algorithm is a first-order method that extends standard gradient descent to a class of non-smooth composite optimization problems. In our setup, it iteratively updates the feedback gain K via application of the proximal operator associated with the function g on the standard gradient descent update. The algorithm is initialized with a stabilizing K^0 and the iterates are determined by,

$$K^{l+1} \;=\; \mathbf{prox}_{\alpha_l g}\big(K^l \;-\; \alpha_l \nabla f(K^l)\big)$$

where l is the iteration index and α_l is the step-size. At each iteration, the step-size is selected via backtracking to guarantee sufficient descent of the objective function and stability of (2). To enhance practical performance, we use the Barzilai-Borwein (BB) step-size initialization [20].

For any matrix V and a positive scalar α, the proximal operator of the function g is defined as [21],

$$\mathbf{prox}_{\alpha g}(V) \;:=\; \underset{Z}{\operatorname{argmin}} \left(g(Z) \;+\; \frac{1}{2\alpha} \|Z \;-\; V\|_F^2 \right)$$

where $\|\cdot\|_F$ is the Frobenius norm. In particular, for $g(K) = \gamma \|K\|_1 = \gamma \sum_{i,\,j} |K_{ij}|$, we have

$$K^{l+1} \;=\; \mathcal{S}_{\gamma\alpha_l}\big(K^l \;-\; \alpha_l \nabla f(K^l)\big)$$

where $\mathcal{S}_\kappa(V)$ is the soft-thresholding function that acts on the individual entries V_{ij} of the matrix V according to $\mathcal{S}_\kappa(V_{ij}) = \operatorname{sign}(V_{ij}) \max(|V_{ij}| - \kappa, 0)$. In the absence of the regularization function g in (6), the proximal gradient algorithm simplifies to standard gradient descent with the updates $K^{l+1} = K^l - \alpha_l \nabla f(K^l)$.

Recall that estimates of $\nabla f(K^l)$ require the matrices KL and PL. These can be computed or estimated from data using the techniques described in the previous section.

Remark 4: The step-size α_l is adjusted via backtracking to ensure stability of closed-loop system (2) at each iteration. It is initialized using the BB method which provides a heuristic for approximating the Hessian of the function f via the scaled version of the identity matrix [20], $(1/\alpha_l)I$. At the lth iteration, the initial BB step-size

$$\alpha_{l,0} \;:=\; \frac{\|K^l \;-\; K^{l-1}\|_2^2}{\operatorname{trace}\left((K^{l-1} \;-\; K^l)^T \left(\nabla f(K^{l-1}) \;-\; \nabla f(K^l)\right)\right)} \tag{22}$$

is adjusted via backtracking until system (2) is stable and sufficient descent is achieved at each iteration by satisfying

$$\begin{aligned} f(K^{l+1}) \;<\; & f(K^l) + \operatorname{trace}\left((K^{l+1} \;-\; K^l)^T \nabla f(K^l)\right) \\ & + \tfrac{1}{2\alpha_l} \|K^{l+1} \;-\; K^l\|_2^2. \end{aligned} \tag{23}$$

The algorithm stops when the generalized gradient map becomes smaller than the given tolerance,

$$\Big\| \frac{K^l \;-\; \mathbf{prox}_{\alpha_l g}\big(K^l \;-\; \alpha_l \nabla f(K^l)\big)}{\alpha_l} \Big\| \;<\; \epsilon.$$

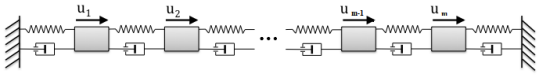

Fig. 1: A mass-spring-damper system.

B. Incremental proximal gradient

Next, we exploit the separable structure obtained in Section II to speed up computations and save memory. This formulation exploits a decomposition into n separable optimization problems, based on f_k and ∇f_k, which reduces computational complexity and allows implementation at scale.

We start with an initial stabilizing gain K^0, and at each iteration we solve the following optimization problem

$$\underset{K}{\text{minimize}} \quad f_k(K) \;+\; g(K). \tag{24}$$

The explicit solution to (24) is given by

$$K^{l+1} \;=\; \mathbf{prox}_{\alpha_l g}\left(K^l \;-\; \alpha_l\, \nabla f_k(K^l)\right).$$

Here, $\mathbf{prox}$ is the proximal operator of the function g, and α_l is a step-size. This is the incremental proximal gradient algorithm because, at each iteration, the optimization variable is updated in the negative direction of a single element of the gradient where the index k is selected in a random manner. At each iteration, we choose step-size to be $O(1/l)$ and adjust it to guarantee stability of closed-loop system (2).

Incremental and stochastic gradient based algorithms have recently found widespread use in large-scale optimization and machine learning. High variance that results from estimating the full gradient ∇f using samples of its entries can result in slow convergence. Moreover, to ensure convergence, the step-size has to decay to zero.

In the first few iterations, the objective function in the incremental proximal gradient decreases dramatically but it starts to oscillate after that. Thus, in large-scale optimization problems where having high accuracy may not be achievable, these methods are useful because of their fast initial convergence rate.

C. Computational experiments

We next provide examples to illustrate the performance of proximal algorithms. The acronym proxG represents the proximal gradient and proxIG is the incremental proximal gradient. We have implemented all algorithms in MATLAB. In all examples, we choose $R = I$ and $Q = I$.

The algorithms were tested for a mass-spring-damper (MSD) model with N masses, illustrated in Figure 1. Our goal is to find the local minimum of the problem (6) with $g(K) = \gamma \|K\|_1$, where γ is the sparsity-promoting parameter and the ℓ_1 norm is a proxy for inducing sparsity in the feedback gain matrix. The MSD system with N masses has $n = 2N$ states where the first N states denote the positions and the rest are velocities. We consider the case

where all masses, spring, and damping constants are equal to one, which results in the state-space model with

$$A = \begin{bmatrix} 0 & I \\ -T & -T \end{bmatrix}, \; B = \begin{bmatrix} 0 \\ I \end{bmatrix}$$

where T is an $N \times N$ tridiagonal symmetric Toeplitz matrix with 2 on the main diagonal and -1 on the first upper and lower sub-diagonals, and I and 0 are $N \times N$ identity and zero matrices.

Sparsity-promoting controllers were designed for this model with $N = 10$ masses and compared with the ADMM-based algorithm [3] that does not utilize the iterative re-weighting scheme. The algorithms were initialized with the (unstructured) optimal feedback gain $K^\star$. Figure 2 shows how the sparsity pattern of the controller changes as the value of the parameter γ is increased. It is worth to note that all algorithms give the same structure of K with very close feedback gain values. For $\gamma = 0$, the optimal feedback controller is given by a dense matrix and as γ increases the controller becomes sparser. It is diagonal when $\gamma = 1$, and for sufficiently large γ the gain is identically zero. This is a feasible solution because the open-loop system is stable.

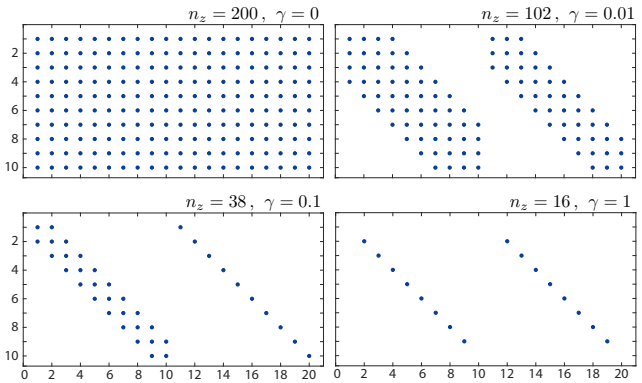

Fig. 2: Structure of feedback gains resulting from the solution to (6) for different values of the regularization parameter γ for an MSD system with $N = 10$ masses. The number of non-zero elements in K is denoted by n_z.

V. Concluding remarks

This paper provides a step towards a comprehensive theory for distributed and structured optimal control of LTI systems. Even within the linear setting there are many open questions, including

(i) Development of simulation-based methods in the presence of disturbances and uncertainty;

(ii) Construction of truly recursive algorithms, similar to those used in traditional reinforcement learning settings;

(iii) Techniques to avoid local minima;

(iv) Development of algorithms that combine proximal methods with natural policy gradient.

Extensions to nonlinear control will probably require more modest objectives since it is not an easy task to compute the performance of a given policy. However, both Q-learning and TD-learning are based on value function approximation. Consequently, as in the present work, these algorithms can be designed so that each value function approximation serves as a Lyapunov function for the current approximating policy. This provides bounds on performance, as well as stability. Consequently, algorithms for computation of structured stabilizing policies with good performance are not out of reach even for nonlinear control systems.

References

[1] 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.

[2] 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*, San Francisco, CA, 2011, pp. 2050–2055.

[3] 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.

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

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

[6] S. Dean, H. Mania, N. Matni, B. Recht, and S. Tu, "On the sample complexity of the linear quadratic regulator," 2017, arXiv:1710.01688.

[7] M. Fazel, R. Ge, S. M. Kakade, and M. Mesbahi, "Global convergence of policy gradient methods for linearized control problems," 2018, arXiv:1801.05039.

[8] N. K. Dhingra, S. Z. Khong, and M. R. Jovanović, "The proximal augmented Lagrangian method for nonsmooth composite optimization," *IEEE Trans. Automat. Control*, 2018, doi:10.1109/TAC.2018.2867589.

[9] N. K. Dhingra and M. R. Jovanović, "A method of multipliers algorithm for sparsity-promoting optimal control," in *Proceedings of the 2016 American Control Conference*, Boston, MA, 2016, pp. 1942–1947.

[10] H. Mohammadi, A. Zare, M. Soltanolkotabi, and M. R. Jovanović, "Nonconvex optimization meets convex reparametrization: convergence of gradient methods for the linear quadratic regulator," in *Proceedings of the 58th IEEE Conference on Decision and Control*, Nice, France, 2019, submitted.

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

[12] 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.

[13] S. Hassan-Moghaddam and M. R. Jovanović, "Distributed design of optimal structured feedback gains," in *Proceedings of the 56th IEEE Conference on Decision and Control*, Melbourne, Australia, 2017, pp. 6586–6591.

[14] D. G. Luenberger, *Optimization by vector space methods*. New York: John Wiley & Sons Inc., 1969, reprinted 1997.

[15] J. N. Tsitsiklis and B. Van Roy, "An analysis of temporal-difference learning with function approximation," *IEEE Trans. Automat. Control*, vol. 42, no. 5, pp. 674–690, 1997.

[16] S. P. Meyn, *Control Techniques for Complex Networks*. Cambridge University Press, 2007, pre-publication edition available online.

[17] P. G. Mehta and S. P. Meyn, "Q-learning and Pontryagin's minimum principle," in *Proceedings of the 48th IEEE Conference on Decision and Control*, 2009, pp. 3598–3605.

[18] S. Amari, "Natural gradient works efficiently in learning," *Neural Computation*, vol. 10, no. 2, pp. 251–276, 1998.

[19] S. M. Kakade, "A natural policy gradient," in *Advances in Neural Information Processing Systems*, 2002, pp. 1531–1538.

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

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