2016 American Control Conference (ACC)
Boston Marriott Copley Place
July 6-8, 2016. Boston, MA, USA

A method of multipliers algorithm for sparsity-promoting optimal control

Neil K. Dhingra and Mihailo R. Jovanović

***Abstract*— We develop a customized method of multipliers algorithm to efficiently solve a class of regularized optimal control problems. By exploiting the problem structure, we transform the augmented Lagrangian into a form which can be efficiently minimized using proximal methods. We apply our algorithm to an ℓ_1-regularized state-feedback optimal control problem and compare its performance with a proximal gradient algorithm and an alternating direction method of multipliers algorithm. In contrast to other methods, our algorithm has both a theoretical guarantee of convergence and fast computation speed in practice.**

***Index Terms*— Augmented Lagrangian, method of multipliers, non-smooth optimization, proximal methods, sparsity-promoting optimal control, structure identification.**

I. Introduction

The design of state-feedback controllers which balance performance with sparsity has been the subject of considerable attention in recent years [1]–[10]. Research efforts have focused on the identification of classes of tractable problems and the development of efficient algorithms for sparse synthesis.

In [1], ℓ_1-regularization was applied to the $\mathcal{H}_2$-optimal state-feedback control problem to promote the design of sparse and block sparse feedback gains. It was shown that, for a class of systems, this problem can be cast as a semidefinite program (SDP). An alternating direction method of multipliers method (ADMM) was utilized in [2] to design feedback gains which balance closed-loop $\mathcal{H}_2$ performance and sparsity. Sparse controllers have also been designed by solving series of convex problems based on SDP relaxations [9] and via techniques based on polynomial optimization and rank minimization [10]. In [3], an LMI-based approximation was used to design structured dynamic output-feedback controllers subject to a given $\mathcal{H}_\infty$ performance criterion. In [4], an exact LMI characterization was introduced for the design of row-sparse controllers and in [5] an efficient ADMM algorithm was developed. A more general framework for regularization in the context of control using atomic norms was provided in [6]–[8].

Our work builds on the sparsity-promoting framework developed in [1], [2]. In spite of its good performance in practice, ADMM lacks convergence guarantees for nonconvex problems and may converge slowly to high accuracy solutions even in convex problems. We thus investigate the use of the method of multipliers [11]–[13] in combination with proximal algorithms [14]. Similar to ADMM, this method utilizes the augmented Lagrangian but provides a guarantee of convergence to a local minimum [11]–[13].

By exploiting the structure of the regularized optimal control problem, we are able to transform the augmented Lagrangian from a nondifferentiable function to a continuously differentiable function. Due to local convexity of the augmented Lagrangian [11]–[13] and Lipschitz continuity of the gradient of the smooth part of the objective function [15], proximal methods can be used to obtain a local minimum [16]. We initialize step-size using the Barzilai-Borwein (BB) method [17] and employ a backtracking line search to ensure convergence [18].

The paper is organized as follows. In Section II, we describe the sparsity-promoting optimal control problem and form the associated augmented Lagrangian. In Section III, we summarize the method of multipliers, transform the augmented Lagrangian into a continuously differentiable function, and develop a customized proximal algorithm for minimization of the augmented Lagrangian. In Section IV, we provide examples to illustrate the effectiveness of our approach. In Section V, we conclude with a brief summary of our work and an overview of ongoing research directions.

II. Problem formulation and background

We consider the control problem for a linear time-invariant system,

$$\begin{aligned} \dot{x} &= (A - B_2 F)\, x + B_1\, d \\ z &= \begin{bmatrix} Q^{1/2} \\ -R^{1/2} F \end{bmatrix} x \end{aligned} \tag{1}$$

where d is an exogenous disturbance, z is the performance output, and $Q = Q^T \succeq 0$ and $R = R^T \succ 0$ are the state and control performance weights. System (1) describes closed-loop dynamics under the state-feedback control law,

$$u = -Fx, \quad F \in \mathbb{R}^{m \times n}.$$

We make the standard assumptions that (A, B_2) is stabilizable and $(A, Q^{1/2})$ is detectable.

We are interested in the problem of structure identification and optimal design of a state-feedback matrix F to minimize the steady-state variance amplification (i.e., the $\mathcal{H}_2$ norm)

$$\lim_{t \to \infty} \mathbf{E}\left(x^T(t)\, Q\, x(t) \;+\; u^T(t)\, R\, u(t)\right)$$

of the closed-loop system, where $\mathbf{E}$ is the expectation operator. The square of the $\mathcal{H}_2$ norm can be expressed as a

Financial support from the National Science Foundation under award ECCS-1407958, the University of Minnesota Informatics Institute Transdisciplinary Faculty Fellowship, the NASA Harriet G. Jenkins Predoctoral Fellowship Program, and the MnDRIVE Graduate Scholars Program is gratefully acknowledged.

N. K. Dhingra and M. R. Jovanović are with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455. E-mails: dhin0008@umn.edu, mihailo@umn.edu.

function of the feedback gain F as

$$J(F) := \begin{cases} \text{trace}\left(PB_1B_1^T\right), & F \text{ stabilizing} \\ +\infty, & \text{otherwise} \end{cases}$$

where P is the closed-loop observability gramian,

$$(A - B_2F)^T P + P(A - B_2F) = -\left(Q + F^T RF\right).$$

An equivalent characterization in terms of the closed-loop controllability gramian X,

$$(A - B_2F)X + X(A - B_2F)^T = -B_1B_1^T \quad (2)$$

is given by

$$J(F) := \begin{cases} \text{trace}\left(X(Q + F^T RF)\right), & F \text{ stabilizing} \\ +\infty, & \text{otherwise.} \end{cases}$$

A. Sparsity-promoting optimal control

The state-feedback gain F which minimizes the closed-loop $\mathcal{H}_2$ norm is, in general, a dense matrix. In [1], [2], the authors studied the problem of designing feedback gain matrices which balance $\mathcal{H}_2$ performance with the sparsity of F. This was achieved by considering a regularized optimal control problem,

$$\text{minimize } J(F) + \gamma\, g(F) \quad \text{(SP)}$$

where $g(F)$ encodes some structural constraint or penalty on F, and $\gamma > 0$ encodes the emphasis on this penalty relative to the $\mathcal{H}_2$ performance. For $\gamma = 0$, the problem simplifies to the $\mathcal{H}_2$ state-feedback problem whose solution is given by the standard linear quadratic regulator. A typical approach is to solve (SP) for a series of different γ and to generate a set of feedback gains with different levels of sparsity. From this set, a sparse feedback gain can be selected or γ can be refined to yield sparser or denser controllers.

In (SP), the regularization term $g(F)$ is introduced as a proxy for managing complexity of the controller. A weighted ℓ_1 penalty on the individual elements $F_{ij} \in \mathbb{R}$ of the feedback gain matrix F,

$$g_1(F) := \sum_{i,j} w_{ij} |F_{ij}| \quad (3)$$

promotes elementwise sparsity. Similarly, the sum of the Frobenius norm of the submatrices $F_{ij} \in \mathbb{R}^{m_i \times n_j}$,

$$g_2(F) = \sum_{i,j} w_{i,j} \|F_{ij}\|_F \quad (4)$$

can be used to promote sparsity at the level of submatrices. Here, the feedback gain F can be partitioned into submatrices that need not have the same size and the weights $w_{ij} \geq 0$ specify the emphasis on sparsity of individual elements (blocks). Alternative regularization terms $g(F)$ can promote the limited use of sensors or actuators [4], [5], enforce the communication of only relative information [19], [20], or penalize more advanced measures of controller complexity [6], [7].

Problem (SP) is difficult to solve directly because J is, in general, a nonconvex function of F and g is typically not differentiable. In the absence of a regularization term (i.e., for $\gamma = 0$), the change of variables $Y := FX$ in (2) can be used to express the square of the $\mathcal{H}_2$ norm as,

$$J(X, Y) := \text{trace}(Q\,X) + \text{trace}\left(X^{-1}\,Y^T RY\right)$$

and to formulate the $\mathcal{H}_2$ optimal control problem as a semidefinite program (SDP). However, for $\gamma > 0$, such a nonlinear change of coordinates in general introduces a non-convex dependence of the regularization term on the optimization variables X and Y. One exception occurs for promoting row-sparsity of F because of the equivalence between the row-sparsities of F and Y [4]. In this case, a penalty on the row-sparsity of Y can be used as a proxy for promoting row-sparsity of F, which leads to a convex characterization. However, even when the resulting optimal control problem is convex, solving SDPs with standard solvers is computationally expensive and customized algorithms are required for large-scale systems.

B. Alternating direction method of multipliers

By introducing an additional optimization variable G, (SP) can be equivalently written as

$$\begin{aligned} \text{minimize} \quad & J(F) + \gamma\, g(G) \\ \text{subject to} \quad & F - G = 0. \end{aligned} \quad \text{(SP1)}$$

This separates the objective function into two parts, the $\mathcal{H}_2$ performance index J and the sparsity-promoting term g. The augmented Lagrangian associated with (SP1) is

$$\mathcal{L}_\rho(F, G; \Lambda) = J(F) + \gamma\, g(G) + \langle \Lambda, F - G \rangle + \frac{\rho}{2} \|F - G\|_F^2$$

where Λ is the Lagrange multiplier, ρ is a positive scalar, and $\langle \cdot, \cdot \rangle$ is the standard matricial inner product. Relative to the regular Lagrangian, the augmented Lagrangian has an additional term that introduces a quadratic penalty on the violation of the linear constraint.

In [2], the alternating direction method of multipliers (ADMM) was used to compute a solution to (SP1) via a sequence of iterations [21]. The ADMM algorithm involves minimization of $\mathcal{L}_\rho$ separately over F and G and an update of the Lagrange multiplier Λ,

$$\begin{aligned} F^{k+1} &= \underset{F}{\text{argmin}}\ \mathcal{L}_\rho(F,\, G^k;\, \Lambda^k) \\ G^{k+1} &= \underset{G}{\text{argmin}}\ \mathcal{L}_\rho(F^{k+1},\, G;\, \Lambda^k) \\ \Lambda^{k+1} &= \Lambda^k + \rho\,(F^{k+1} - G^{k+1}). \end{aligned} \quad (5)$$

C. Background on proximal operators

Since our customized algorithms utilize properties of proximal operators, we next provide a brief overview; for additional information, see [14].

The proximal operator associated with the function f is defined by,

$$\mathbf{prox}_{\mu f}(V) := \underset{F}{\text{argmin}}\ f(F) + \frac{1}{2\mu} \|F - V\|_F^2$$

and the optimal value determines its Moreau envelope,

$$M_{\mu f}(V) \;:=\; \inf_F \; f(F) \;+\; \frac{1}{2\mu}\,\|F \,-\, V\|_F^2.$$

The Moreau envelope is a continuously differentiable function, even when f is not, and its gradient is given by [14],

$$\nabla M_{\mu f}(V) \;=\; \frac{1}{\mu}\left(V \;-\; \mathbf{prox}_{\mu f}(V)\right).$$

In fact, the ADMM algorithm (5) can be equivalently expressed as [14],

$$\begin{aligned} F^{k+1} \;&=\; \mathbf{prox}_{\rho^{-1}J}\big(G^k \,-\, (1/\rho)\,\Lambda^k\big) \\ G^{k+1} \;&=\; \mathbf{prox}_{\rho^{-1}g}\big(F^{k+1} \,+\, (1/\rho)\,\Lambda^k\big) \\ \Lambda^{k+1} \;&=\; \Lambda^k \,+\, \rho\,(F^{k+1} \,-\, G^{k+1}). \end{aligned}$$

Since the regularization term g typically has a proximal operator that is easy to evaluate, the challenging aspect of ADMM for (SP) lies in the F-minimization step, i.e., in the evaluation of the proximal operator of the function J. This operator is not known explicitly and it is determined by solving a smooth nonconvex optimization problem [2].

III. The method of multipliers

Recently, ADMM has found wide-spread use in distributed optimization problems because it can exploit separability in the components of the objective function regardless of the form of linear constraint [21]. However, in [2], ADMM was utilized as a general purpose algorithm for solving the sparsity-promoting optimal control problem (SP1).

The method of multipliers is the most widely used algorithm for solving constrained nonlinear programing problems [11]–[13], [22]. While it requires a joint minimization of the augmented Lagrangian over F and G, in contrast to ADMM, it is guaranteed to converge to a local minimum even for nonconvex problems. Furthermore, the parameter ρ in the augmented Lagrangian $\mathcal{L}_\rho$ can be systematically adjusted and the minimization of $\mathcal{L}_\rho$ can be inexact up to *a priori* specified tolerances [22].

In this paper, we exploit the special structure of the linear constraint in (SP1) to utilize the separability of the optimality conditions with respect to G in order to eliminate it from the augmented Lagrangian. This leads to an optimization problem with a continuously differentiable objective function.

For the remainder of this paper, we will restrict our attention to the case where the regularization function g in (SP1) is the weighted ℓ_1-norm of the feedback gain,

$$g(G) \;:=\; \sum_{i,\,j} w_{ij}\,|G_{ij}|.$$

A. Elimination of G

In contrast to ADMM, each iteration of the method of multipliers requires *joint* minimization of the augmented Lagrangian with respect to F and G,

$$(F^{k+1}, G^{k+1}) \;=\; \operatorname*{argmin}_{F,\,G}\; \mathcal{L}_\rho(F,\,G;\,\Lambda^k)$$

followed by the update of the Lagrange multiplier Λ and the penalty parameter ρ.

The proximal operator associated with g can be used to eliminate the optimization variable G from the augmented Lagrangian. Minimization of $\mathcal{L}_\rho$ with respect to G gives,

$$G^\star \;=\; \mathbf{prox}_{\kappa g}(F \,+\, (1/\rho)\Lambda)$$

where $\kappa := \gamma w_{ij}/\rho$. Substitution of $G^\star$ into the augmented Lagrangian yields a formulation in terms of the Moreau envelope,

$$\mathcal{L}_\rho(F;\Lambda) \;=\; J(F) \,+\, \gamma M_{\kappa g}(F + (1/\rho)\Lambda) \,-\, \frac{1}{2\rho}\,\|\Lambda\|_F^2 \quad (6)$$

where $\mathcal{L}_\rho(F;\Lambda) := \mathcal{L}_\rho(F,\,G^\star;\,\Lambda)$. Properties of proximal operators summarized in Section II-C imply that this expression for $\mathcal{L}_\rho(F;\Lambda)$ is at least once continuously differentiable. This means that minimization of the augmented Lagrangian with respect to F and G, which is a nondifferentiable function, can be achieved by minimizing the differentiable function (6) over F.

B. The method of multipliers algorithm

The method of multipliers alternates between minimization of the augmented Lagrangian with respect to F (for a fixed value of the parameter ρ_k and the Lagrange multiplier Λ^k) and the update of ρ and Λ. In particular, a solution to (SP1) can be found using

$$F^{k+1} \;=\; \operatorname*{argmin}_F\; \mathcal{L}_{\rho_k}(F;\Lambda^k) \quad (7a)$$

$$\Lambda^{k+1} \;=\; \Lambda^k \,+\, \rho_k\, C^{k+1} \quad (7b)$$

where

$$C^{k+1} \;:=\; F^{k+1} \,-\, \mathbf{prox}_{(\gamma/\rho_k)g}(F^{k+1} \,+\, (1/\rho_k)\Lambda^k) \quad (7c)$$

denotes the difference between F^{k+1} and $G^\star$ at (F^{k+1}, Λ^k).

An efficient procedure for solving (SP1) via the method of multipliers is summarized in Algorithm 1. This algorithm closely follows [22, Algorithm 17.4] where a method for adaptively adjusting parameter ρ_k is provided and, compared to (7), a more refined update of the Lagrange multiplier Λ is used. In Algorithm 1, η^* and ω^* are convergence tolerances, and $\rho_{\max}$ is a maximum value for the penalty parameter ρ.

C. Minimization of the augmented Lagrangian

The main computational burden in the method of multipliers lies in finding a solution to the optimization problem (7a). Although the differentiability of $\mathcal{L}_\rho$ implies that gradient descent may be employed to update F, we utilize the proximal gradient method to exploit the structure of the Moreau envelope associated with g.

To avoid clutter, we suppress the superscripts k as well as the dependence of $\mathcal{L}_\rho$ on Λ^k and use the notation $\{F^m\}$ to denote the sequence of inner iterates that converge to a solution of (7a).

We first recall the gradient of the smooth part of the objective function in (SP); for additional details see [15].

input: Initial point F^0 and Lagrange multiplier Λ^0
initialize: $\rho_0 = 10$, $\omega_0 = 1/\rho_0$, and $\eta_0 = 1/\rho_0^{0.1}$.
for $k = 0, 1, 2, \ldots$

Solve (7a) using proximal gradient such that

$$\|\nabla \mathcal{L}_\rho(F^{k+1}, \Lambda^k)\|_F \;\le\; \omega_k$$

if $\|C_{k+1}\|_F \le \eta_k$

if $\|C_{k+1}\|_F \le \eta^*$ and $\|\nabla \mathcal{L}_\rho(F^{k+1}, \Lambda^k)\|_F \le \omega^*$

stop with approximate solution F^{k+1}

else

$$\Lambda^{k+1} = \Lambda^k + \rho_k C_{k+1}, \qquad \rho_{k+1} = \rho_k$$
$$\eta_{k+1} = \eta_k/\rho_{k+1}^{0.9}, \qquad \omega_{k+1} = \omega_k/\rho_{k+1}$$

endif

else

$$\Lambda^{k+1} = \Lambda^k, \qquad \rho_{k+1} = \min\{5\rho_k, \rho_{\max}\}$$
$$\eta_{k+1} = 1/\rho_{k+1}^{0.1}, \qquad \omega_{k+1} = 1/\rho_{k+1}$$

endif
endfor

Algorithm 1: Method of Multipliers for (SP1).

Proposition 1: The gradient of the $\mathcal{H}_2$ norm with respect to F is given by

$$\nabla J(F) \;=\; 2(RF \;-\; B_2^T P)L$$

where P and L are observability and controllability gramians of the closed-loop system,

$$\begin{aligned} A_{\rm cl}^T P \;+\; P A_{\rm cl} \;&=\; -(Q \;+\; F^T R F) \\ A_{\rm cl} L \;+\; L A_{\rm cl}^T \;&=\; -B_1 B_1^T \end{aligned}$$

and $A_{\rm cl} := A \;-\; B_2 F$. Furthermore, $\nabla J(F)$ is a Lipschitz continuous function on the set of stabilizing feedback gains.

1) Proximal gradient descent: Proximal gradient descent provides a generalization of standard gradient descent which can be applied to nonsmooth optimization problems. The standard gradient descent update $F^{m+1} = F^m + \tilde{F}^m$ where $\tilde{F}^m = -\alpha_m \nabla f(F^m)$ can be interpreted as the minimizer to a simple quadratic approximation of f around the current iterate F^m,

$$\widetilde{F}^m \;=\; \operatorname*{argmin}_{\widetilde{F}} \; f(F^m) \;+\; \left\langle \nabla J(F^m), \widetilde{F} \right\rangle \;+\; \frac{1}{2\alpha_m} \|\widetilde{F}\|_F^2$$

where α_m is the step-size. If f can be expressed as,

$$f(F) \;:=\; f_1(F) \;+\; f_2(F)$$

where f_1 is differentiable, the proximal gradient update $\widetilde{F}^m$ is derived from a quadratic approximation of f_1,

$$\begin{aligned} \widetilde{F}^m \;=\; \operatorname*{argmin}_{\widetilde{F}} \; & f_1(F^m) \;+\; \left\langle \nabla f_1(F^m), \widetilde{F} \right\rangle \;+\; \frac{1}{2\alpha_m} \|\widetilde{F}\|_F^2 \\ & +\; f_2(F^m \;+\; \widetilde{F}). \end{aligned}$$

The update [14] is given by the proximal operator,

$$F^{m+1} \;=\; \mathbf{prox}_{\alpha_m f_2}(F^m \;-\; \alpha_m \nabla f_1(F^m)).$$

Clearly, the proximal gradient method is most effective when the proximal operator of the function f_2 is easy to evaluate.

2) Proximal gradient step for minimizing $\mathcal{L}_\rho$: When g is the weighted ℓ_1-norm, the Moreau envelope $M_{\kappa g}$ is given by the Huber function

$$h_\kappa(v) \;=\; \begin{cases} \frac{1}{2}\, v^2, & |v| \;\le\; \kappa \\ \kappa\,(|v| \;-\; \frac{1}{2}\,\kappa), & |v| \;\ge\; \kappa \end{cases}$$

which acts on each element of its matricial argument. Its gradient is the elementwise saturation operator,

$$\nabla h_\kappa(v) \;=\; \mathrm{sat}_\kappa(v) \;:=\; \begin{cases} v, & |v| \;\le\; \kappa \\ \kappa\,\mathrm{sign}\,(v), & |v| \;\ge\; \kappa. \end{cases}$$

The proximal gradient update $F^{m+1} \;=\; F^m + \widetilde{F}^m$ for minimizing the $\mathcal{L}_\rho$ over F in subproblem (7a) is given by,

$$\widetilde{F}_{ij}^m \;=\; \begin{cases} -\alpha_m((\nabla J)_{ij} + \rho\kappa), & U_{ij}^m \;\ge\; \kappa\,(\alpha_m\rho + 1) \\ -\dfrac{\alpha_m((\nabla J)_{ij} + \rho V_{ij}^m)}{1 \;+\; \alpha_m\rho}, & |U_{ij}^m| \;\le\; \kappa\,(\alpha_m\rho + 1) \\ -\alpha_m((\nabla J)_{ij} - \rho\kappa), & U_{ij}^m \;\le\; -\kappa\,(\alpha_m\rho + 1) \end{cases} \tag{8}$$

where

$$\begin{aligned} V^m \;&:=\; F^m \;+\; (1/\rho)\Lambda \\ U^m \;&:=\; V^m \;-\; \alpha_m \nabla J(F^m). \end{aligned}$$

This expression for the update of F follows from the separability of the Moreau envelope. Minimizing of the sum of a quadratic function and a Huber function has an explicit solution. By defining $a := \widetilde{F}_{ij}^m$, $b := (\nabla J(F^m))_{ij}$, and $c := V_{ij}^m$, optimization over each element of $\widetilde{F}$ can be expressed as,

$$\operatorname*{minimize}_a \; \frac{1}{2\alpha}\, a^2 \;+\; b\, a \;+\; \rho\, h_\kappa(a + c).$$

Setting the gradient to zero yields,

$$a \;+\; \alpha\,(b \;+\; \rho\,\mathrm{sat}_\kappa(a \;+\; c)) \;=\; 0.$$

Considering the separate cases when $\mathrm{sat}_\kappa(a+c) = \kappa$, $a+c$, and $-\kappa$ yields the optimal a,

$$a^\star \;=\; \begin{cases} -\,\alpha(b \;+\; \rho\kappa), & \alpha b - c \;\ge\; \kappa(\alpha\rho \;+\; 1) \\ -\,\dfrac{\alpha}{1 \;+\; \alpha\rho}(b \;+\; \rho c), & |\alpha b - c| \le\; \kappa(\alpha\rho \;+\; 1) \\ -\,\alpha(b \;-\; \rho\kappa), & \alpha b - c \;\le -\kappa(\alpha\rho \;+\; 1) \end{cases}$$

which is equivalent to (8).

3) Step-size selection: Since the objective function is not smooth, an Armijo backtracking rule cannot be used. Instead, we backtrack from $\alpha_{m,0}$ by selecting the smallest nonnegative integer r such that $\alpha_m = c^r \alpha_{m,0}$ with $c \in (0, 1)$

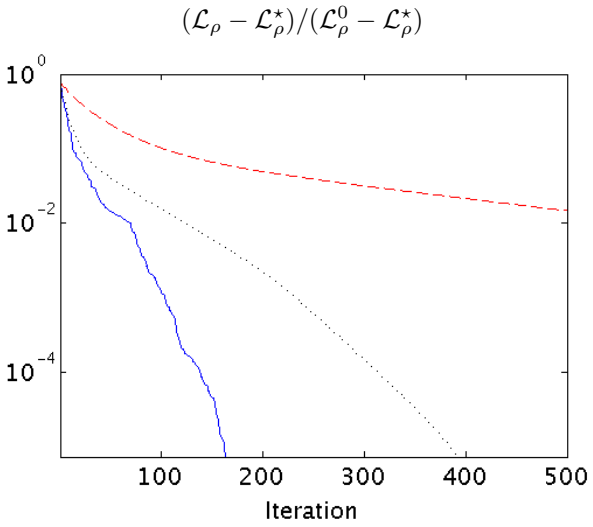

Fig. 1. Comparison of proximal gradient with BB step-size selection (solid blue), proximal gradient without BB step-size selection (black dotted) and gradient descent with BB step-size selection (red dashed) for the F-minimization step (7a) for an unstable network with 20 subsystems, $\gamma = 0.0844$, and $\rho = 10$. The y-axis shows the distance from the optimal objective value relative to initial distance, $(\mathcal{L}_\rho - \mathcal{L}_\rho^\star)/(\mathcal{L}_\rho^0 - \mathcal{L}_\rho^\star)$.

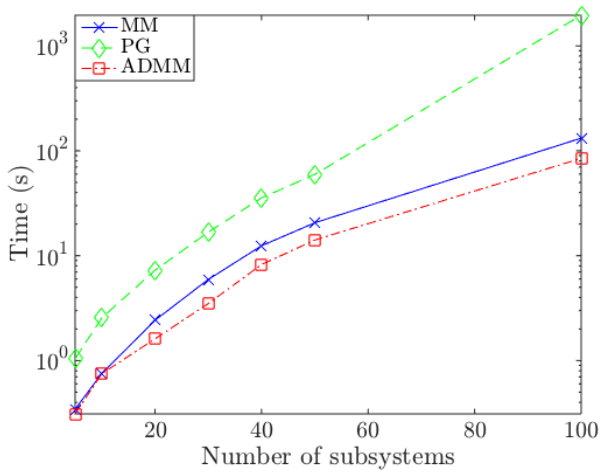

Fig. 2. Computation time required to solve (SP) for 10 evenly spaced values of γ from 0.001 to 1.0 for a mass-spring example with $N = 5, 10, 20, 30, 40, 50, 100$ masses. Performance of direct proximal gradient (green dashed $\diamond$), the method of multipliers (blue solid $\times$) and ADMM (red dash-dot $\square$) is displayed. All algorithms use BB step-size initialization.

such that F^{m+1} is stabilizing and,

$$J(F^{m+1}) \;\leq\; J(F^m) \;+\; \left\langle \nabla J(F^m), F^{m+1} \;-\; F^m \right\rangle \;+\; \frac{1}{2\alpha_m} \|F^{m+1} \;-\; F^m\|_F^2.$$

This backtracking rule adaptively estimates the Lipschitz constant of $\nabla J(F)$ to ensure convergence [18].

To improve the speed of the proximal gradient algorithm, we initialize the step-size using the Barzilai-Borwein (BB) method [17],

$$\alpha_{m,0} \;=\; \frac{\|F^m \;-\; F^{m-1}\|_F^2}{\langle F^{m-1} \;-\; F^m, \nabla J(F^{m-1}) \;-\; \nabla J(F^m)\rangle}.$$

Figure 1 illustrates the utility of the proximal gradient method over standard gradient descent and the advantage of BB step-size initialization.

D. Proximal gradient applied to (SP)

It is also possible to solve (SP) directly using proximal gradient descent. This algorithm is guaranteed to converge to a local optimal point [23], but we find that in practice it takes longer to find a solution than the method of multipliers. The proximal operator for the weighted ℓ_1-norm is the elementwise softhresholding operator,

$$\mathcal{S}_\beta(v) \;:=\; \begin{cases} 0, & |v| \;\leq\; \beta \\ v \;-\; \beta\,\mathrm{sign}(v), & |v| \;\geq\; \beta \end{cases}$$

and the update for solving (SP) directly is given by

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

where $\beta := \gamma w_{ij} \alpha_k$ and α_k is the step-size. The backtracking and BB step-size initialization rules described in Section III-C.3 are also used here.

IV. Examples

We next illustrate the utility of our approach using two examples. We compare our method of multipliers algorithm with the ADMM algorithm from [2] and a direct application of the proximal gradient method.

A. Mass-spring system

Consider a series of N masses connected by linear springs. The dynamics of each mass are described by

$$\ddot{p}_i \;=\; -(p_i \;-\; p_{i+1}) \;-\; (p_i \;-\; p_{i-1}) \;+\; d_i \;+\; u_i$$

where p_i is the position of the ith mass. When the first and last masses are affixed to rigid bodies, the aggregate dynamics are given by

$$\begin{bmatrix} \dot{p} \\ \dot{v} \end{bmatrix} \;=\; \begin{bmatrix} 0 & I \\ -T & 0 \end{bmatrix} \begin{bmatrix} p \\ v \end{bmatrix} \;+\; \begin{bmatrix} 0 \\ I \end{bmatrix} d \;+\; \begin{bmatrix} 0 \\ I \end{bmatrix} u$$

where p, v, and d are the position, velocity and disturbance vectors, and T is a Toeplitz matrix with 2 on the main diagonal and -1 on the first super- and sub-diagonals.

In Figure 2, we compare the time required to compute a series of sparse feedback gains for 10 values of γ, linearly spaced between 0.001 and 1.0. Taking $\gamma = 1.0$ corresponds to roughly 6% nonzero elements in the feedback gain matrix.

Among the three algorithms, ADMM is the fastest; however, the method of multipliers is comparable and scales at the same rate. Direct proximal gradient was the slowest and exhibited the worst scaling. Since the mass-spring system has benign dynamics, we next consider an unstable network.

B. Unstable network

Let N nodes be uniformly randomly distributed in a box. Each node is an unstable second order system coupled with

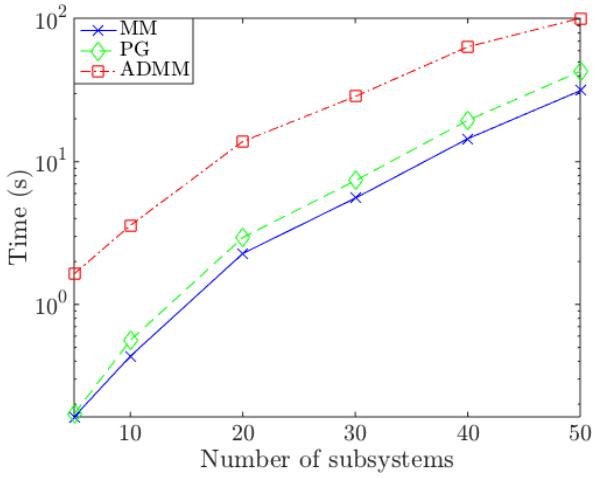

Fig. 3. Computation time required to solve (SP) for 20 evenly spaced values of γ from 0.001 to 0.05 for unstable network examples with $N = 5, 10, 20, 30, 40, 50$ nodes. Performance of direct proximal gradient (green dashed $\diamond$), the method of multipliers (blue solid $\times$) and ADMM (red dash-dot $\square$) is displayed. All algorithms use BB step-size initialization.

nearby nodes via an exponentially decaying function of the Euclidean distance $\delta(i,j)$ between them [24],

$$\begin{bmatrix} \dot{x}_{1i} \\ \dot{x}_{2i} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x_{1i} \\ x_{2i} \end{bmatrix} + \sum_{j \neq i} \mathrm{e}^{-\delta(i,j)} \begin{bmatrix} x_{1j} \\ x_{2j} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} (d_i + u_i)$$

where Q and R are taken to be the identity. Note that simple truncation of the centralized controller could result in a non-stabilizing feedback matrix [24]. We solve (SP1) for γ varying from 0.001 to 0.05 in 20 linearly spaced increments. On average, $\gamma = 0.05$ corresponds to approximately 25% nonzero entries in the feedback gain matrix.

Computation times for N varying from 5 to 50, are shown in Figure 3. Since networks are randomly generated, we average the computation time for 5 networks of each size. For this more complicated example, the method of multipliers algorithm is the fastest and ADMM is the slowest.

V. Concluding Remarks

We have developed a customized algorithm for a sparsity-promoting optimal control problem. Our approach combines the method of multipliers with proximal algorithms. We have provided a comparison with the ADMM algorithm [2] and a direct proximal gradient method. Our algorithm consistently outperforms direct proximal gradient, is competitive with ADMM for a simple example and faster than ADMM for a more complicated example. Our method of multipliers algorithm is appealing because it both provides a guarantee of convergence and performs well in practice.

Our ongoing effort focuses on developing accelerated proximal gradient methods [18] for minimizing the augmented Lagrangian in (7a). We are also exploring the utility of the method of multipliers for other classes of regularized optimal control problems including sensor/actuator selection and topology identification in dynamic networks.

References

[1] M. Fardad, F. Lin, and M. R. Jovanovic, "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.

[2] F. Lin, M. Fardad, and M. R. Jovanovic, "Design of optimal sparse feedback gains via the alternating direction method of multipliers," *IEEE Trans. Automat. Control*, vol. 58, no. 9, pp. 2426–2431, September 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] B. Polyak, M. Khlebnikov, and P. Shcherbakov, "An LMI approach to structured sparse feedback design in linear control systems," in *Proceedings of the 2013 European Control Conference*, 2013, pp. 833–838.

[5] N. K. Dhingra, M. R. Jovanovic, and Z. Q. Luo, "An admm algorithm for optimal sensor and actuator selection," in *Proceedings of the 53rd IEEE Conference on Decision and Control*, Los Angeles, CA, 2014, pp. 4039–4044.

[6] N. Matni and V. Chandrasekaran, "Regularization for design," in *Proceedings of the 53rd IEEE Conference on Decision and Control*, Los Angeles, CA, 2014, pp. 1111–1118.

[7] N. Matni, "Communication delay co-design in H_2 distributed control using atomic norm minimization," *IEEE Trans. Control Netw. Syst.*, 2016, accepted; also arXiv:1404.4911.

[8] N. Matni and V. Chandrasekaran, "Regularization for design," *IEEE Trans. Automat. Control*, 2016, accepted; also arXiv:1404.1972.

[9] M. Fardad and M. R. Jovanovic, "On the design of optimal structured and sparse feedback gains using semidefinite programming," in *Proceedings of the 2014 American Control Conference*, Portland, OR, 2014, pp. 2438–2443.

[10] Y. Wang, J. Lopez, and M. Sznaier, "Sparse static output feedback controller design via convex optimization," in *Proceedings of the 53rd IEEE Conference on Decision and Control*, 2014, pp. 376–381.

[11] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*. New York: Academic Press, 1982.

[12] D. P. Bertsekas, *Nonlinear programming*. Athena Scientific, 1999.

[13] D. G. Luenberger and Y. Ye, *Linear and Nonlinear Programming*. Springer, 2008.

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

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

[16] A. Kaplan and R. Tichatschke, "Proximal point methods and nonconvex optimization," *Journal of Global Optimization*, vol. 13, no. 4, pp. 389–406, 1998.

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

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

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

[20] 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.

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

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

[23] A. Patrascu and I. Necoara, "Efficient random coordinate descent algorithms for large-scale structured nonconvex optimization," *Journal of Global Optimization*, vol. 61, no. 1, pp. 19–46, 2015.

[24] N. Motee and A. Jadbabaie, "Optimal control of spatially distributed systems," *IEEE Trans. Automat. Control*, vol. 53, no. 7, pp. 1616–1629, 2008.