2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

Game-Theoretic Mixed H_2/H_∞ Control with Sparsity Constraint for Multi-Agent Control Systems*

Feier Lian[1], Aranya Chakrabortty[2] and Alexandra Duel-Hallen[3]

***Abstract*— A mixed H_2/H_∞ control problem under a sparsity constraint is investigated for multi-agent control systems (MAS) to provide robustness against model uncertainty and to reduce the communication cost. First, proximal alternating linearized minimization (PALM) is employed to develop a centralized social optimization algorithm, which is guaranteed to converge to a globally optimal sparse controller. Next, we investigate a noncooperative game that accommodates different control performance criteria of the agents and propose a best-response dynamics algorithm based on PALM. A special case of this game produces a partially distributed social optimization solution. We validate the proposed algorithms using a network with open-loop-unstable nodes and demonstrate superiority of the PALM-based method to a previously investigated sparsity-constrained mixed H_2/H_∞ controller.**

I. Introduction

Recently, sparse controller designs for model uncertainties have been proposed for multi-agent control systems (MAS) to reduce the communication cost, which can be very high in practical control systems due to exchange of large volumes of feedback data [1]–[3]. Moreover, differential games have been investigated where each player aims to optimize its individual objective using an associated control policy as in, e.g., power systems [4] or autonomous transportation networks [5]. However, combination of the sparsity constraint and H_2/H_∞ objectives to create a holistic framework for sparse control of uncertain multi-agent systems is missing, and existing designs cannot be readily extended to meet these combined objectives.

In this paper, we investigate controller designs that aim to reduce H_2 cost under H_∞ and sparsity constraints for MAS under norm-bounded parametric uncertainties [2], [6]. First, we present an algorithm to compute a *centralized, socially-optimal, sparse H_2/H_∞ controller*. Next, we develop a *noncooperative game*, where each agent designs its own part of the feedback matrix, and present a numerical algorithm to find an Approximate Local Equilibrium (ALE) of this game. Finally, we adopt the proposed game to obtain a *partially distributed implementation of the social optimization*. We validate the proposed algorithms using numerical simulations and *prove convergence of the centralized algorithm*. While existence of an ALE or convergence to it are not guaranteed due to the non-convex optimization objective, we provide numerical examples and conditions for convergence for the proposed noncooperative game algorithm.

In this paper, we utilize the proximal alternating linearized minimization (PALM) [7], which has been shown to be effective for optimization for nonconvex nonsmooth problems [8] and was used in [7] in a sparsity-constrained output-feedback co-design problem. We demonstrate that PALM significantly outperforms the greedy gradient support pursuit (GraSP), which was employed in our earlier work [6].

The rest of the paper is organized as follows. Section II presents the system model with parametric uncertainty and develops and analyzes a centralized PALM algorithm. Section III describes a MAS with parametric uncertainty, proposes a noncooperative game, develops partially distributed numerical algorithms for this game and for social optimization, and discusses their complexity and convergence properties. Section IV demonstrates effectiveness of the proposed algorithms using numerical simulations. Section V outlines future directions and concludes the paper. Table I contains frequently used notation.

II. PALM algorithm for centralized, sparsity-constrained mixed H_2/H_∞ control

A. System model

Consider the following linear time-invariant system with model uncertainty

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \underbrace{(\mathbf{A}+\Delta\mathbf{A})}_{\hat{\mathbf{A}}}\mathbf{x}(t) + \underbrace{(\mathbf{B}+\Delta\mathbf{B})}_{\hat{\mathbf{B}}}\mathbf{u}(t) + \mathbf{B}_2\mathbf{w}_2(t)\\ \mathbf{z}_2(t) &= \mathbf{C}_2\mathbf{x}(t) + \mathbf{D}_2\mathbf{u}(t)\\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t),\end{aligned} \tag{1}$$

where $\mathbf{x}(t)\in\mathbb{R}^{n\times 1}$ is the state vector, $\mathbf{u}(t)\in\mathbb{R}^{m\times 1}$ is the control input vector, $\mathbf{w}_2(t)\in\mathbb{R}^{m_2\times 1}$ is the exogenous input, $\mathbf{z}_2(t)\in\mathbb{R}^{p_2\times 1}$ is the performance output, $\mathbf{y}(t)\in\mathbb{R}^{p\times 1}$ is the measured output, $\mathbf{A}$ and $\mathbf{B}$ are the nominal state and control input matrices, respectively, while $\Delta\mathbf{A}$ and $\Delta\mathbf{B}$ model their respective uncertainties. Without loss of generality, we make the following assumptions:

Assumption 1: (i) The pair $(\mathbf{A},\mathbf{B})$ is stabilizable, $(\mathbf{C},\mathbf{A})$ is detectable.
(ii) $\Delta\mathbf{A}$ and $\Delta\mathbf{B}$ have the form [2]

$$[\Delta\mathbf{A}\ \Delta\mathbf{B}] = \mathbf{B}_1\Delta\boldsymbol{\delta}[\mathbf{C}_1\ \mathbf{D}_1], \tag{2}$$

where $\mathbf{B}_1\in\mathbb{R}^{n\times m_1}$, $\mathbf{C}_1\in\mathbb{R}^{p_1\times n}$, $\mathbf{D}_1\in\mathbb{R}^{p_1\times m}$ are known matrices, and $\Delta\boldsymbol{\delta}\in\mathbb{R}^{m_1\times p_1}$ is an unknown matrix, which is norm-bounded, satisfying $\Delta\boldsymbol{\delta}^T\Delta\boldsymbol{\delta} \preceq 1/\gamma^2\mathbf{I}$ for any scalar $\gamma > 0$.

*Financial support from the NSF Grant EECS 1544871 is gratefully acknowledged.

[1]Udemy, Inc, 600 Harrison Street, 3rd Floor San Francisco, CA 94107 flian2@alumni.ncsu.edu
[2]Electrical and Computer Engineering, North Carolina State University, Raleigh, NC 27695 USA aranya.chakrabortty@ncsu.edu
[3]Electrical and Computer Engineering, North Carolina State University, Raleigh, NC 27695 USA sasha@ncsu.edu

Assumption 2: Matrices $\mathbf{C}_2$, $\mathbf{D}_2$ have the following form:

$$\mathbf{C}_2 = \begin{bmatrix} \mathbf{C}_2^1 \\ \mathbf{0}_{p_2^u \times n} \end{bmatrix}, \mathbf{D}_2 = \begin{bmatrix} \mathbf{0}_{p_2^x \times m} \\ \mathbf{D}_2^2 \end{bmatrix}. \tag{3}$$

where $\mathbf{C}_2^1 \in \mathbb{R}^{p_2^x \times n}$, $\mathbf{D_2^2} \in \mathbb{R}^{p_2^u \times m}$.

Assumption 1(i) is necessary for the existence of an output feedback controller that stabilizes the closed loop system (1). Assumption 1(ii) specifies the structure of uncertainty we consider in this paper, i.e., the unknown but bounded uncertainty widely considered in other research work e.g. [2]. Assumption 2 limits our focus to a common LQR control scenario where the H_2 cost is determined only by the energies of the state signal and control input signal. Using the above assumptions, the system (1) can be expressed as the feedback interconnection of the following two subsystems:

$$\Sigma : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{B}_1\mathbf{w}_1(t) + \mathbf{B}_2\mathbf{w}_2(t) \\ \mathbf{z}_1(t) = \mathbf{C}_1\mathbf{x}(t) + \mathbf{D}_1\mathbf{u}(t) \\ \mathbf{z}_2(t) = \mathbf{C}_2\mathbf{x}(t) + \mathbf{D}_2\mathbf{u}(t) \\ \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) \end{cases} \tag{4}$$

$$\Sigma_K : \left\{ \mathbf{w}_1(t) = \Delta\boldsymbol{\delta}\mathbf{z}_1(t), \right. \tag{5}$$

where $\mathbf{z}_1(t) \in \mathbb{R}^{p_1 \times 1}$, $\mathbf{w}_1(t) \in \mathbb{R}^{m_1 \times 1}$.

B. Sparsity-constrained mixed H_2/H_∞ control

To find a sparse static controller $u(t) = -\mathbf{K}\mathbf{y}(t)$ that stabilizes the uncertain system in (4)-(5), we formulate the following sparsity-constrained mixed H_2/H_∞ problem

$$\begin{aligned} &\min_{\mathbf{K}} ||T_{z_2 w_2}(\mathbf{K})||_2, \\ &\text{s.t. } ||T_{z_1 w_1}(\mathbf{K})||_\infty < \gamma, \ \text{card}(\mathbf{K}) \le s, \end{aligned} \tag{6}$$

with the plant model satisfying (1), where $T_{z_i w_i}(\mathbf{K})$ represents the closed-loop transfer function from w_i to z_i.

We denote the H_2 objective function in (6) by $J(\mathbf{K})$. Using the impulse response definition of H_2 norm, the H_2 norm square in (6) can be written as

$$\begin{aligned} J(\mathbf{K}) &:= ||T_{z_2 w_2}(\mathbf{K})||_2^2 = \int_{t=0}^{\infty} z_2^T z_2 dt \\ &= \int_{t=0}^{\infty} \left[\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{u}^T\mathbf{R}\mathbf{u}\right] dt. \end{aligned} \tag{7}$$

where the last equation holds due to the Assumption 2, and

$$\mathbf{Q} = (\mathbf{C}_2^1)^T\mathbf{C}_2^1 \succeq 0, \ \mathbf{R} = (\mathbf{D}_2^2)^T\mathbf{D}_2^2 \succ 0 \tag{8}$$

C. Centralized PALM algorithm

First, we replace the variable $\mathbf{K}$ in problem (6) by two variables $\mathbf{K} \in \mathbb{R}^{m \times n}$ and $\mathbf{F} \in \mathbb{R}^{m \times n}$ where $\mathbf{K} = \mathbf{F}$ and $\mathbf{K}$ and $\mathbf{F}$ represent the feedback matrices satisfying the H_∞ and cardinality constraints, respectively. Second, we relax the assumption $\mathbf{K} = \mathbf{F}$ and solve the optimization problem:

$$\begin{aligned} &\min_{\mathbf{K},\mathbf{F}} J(\mathbf{K}) + \frac{\rho}{2}||\mathbf{K} - \mathbf{F}||_F^2, \\ &\text{s.t. } T_\infty(\mathbf{K}) < \gamma, \ \text{card}(\mathbf{F}) \le s, \end{aligned} \tag{9}$$

where the penalty term $\rho/2||\mathbf{K} - \mathbf{F}||_F^2$ is used to regularize the difference between $\mathbf{K}$ and $\mathbf{F}$. If the regularization parameter $\rho > 0$ is chosen large enough, then $\mathbf{F}$ (9) is sufficiently close to $\mathbf{K}$ (6) in Frobenius norm. In (9) and the rest of the paper, we use $T_\infty(\mathbf{K})$ as the short-hand notation for $||T_{z_1 w_1}(\mathbf{K})||_\infty$. Furthermore, the constrained optimization problem (9) can be transformed into the following unconstrained optimization problem by converting the inequality constraints into indicator functions:

$$\min_{\mathbf{K},\mathbf{F}} \Phi(\mathbf{K}, \mathbf{F}) := J(\mathbf{K}) + g(\mathbf{K}) + f(\mathbf{F}) + H(\mathbf{K}, \mathbf{F}). \tag{10}$$

where

$$g(\mathbf{K}) = \begin{cases} 0, & T_\infty(\mathbf{K}) < \gamma \\ +\infty, & otherwise. \end{cases} \tag{11}$$

$$f(\mathbf{F}) = \begin{cases} 0, & \text{card}(\mathbf{F}) \le s \\ +\infty, & otherwise. \end{cases} \tag{12}$$

$$H(\mathbf{K}, \mathbf{F}) = \frac{\rho}{2}||\mathbf{K} - \mathbf{F}||_F^2. \tag{13}$$

Algorithm 1 describes the proposed centralized PALM algorithm. In Steps 2 and 3 of Algorithm 1, $\mathbf{F}$-*minimization* (17) and $\mathbf{K}$-*minimization* (18) are performed, respectively. Plugging (12),(16) into (17), the F-minimization reduces to (see eq. (4.20)- (4.28) of [9] for detailed derivations)

$$\begin{aligned} \mathbf{F}^{k+1} = \arg\min_{\mathbf{F}} \ &||\mathbf{F} - \mathbf{Z}^k||_F^2 \\ \text{s.t. } \ &\text{card}(\mathbf{F}) \le s, \end{aligned} \tag{14}$$

TABLE I: Notation

Term	Definition
$\mathbf{M}\succ 0 (\succeq 0)$	Matrix $\mathbf{M}$ is positive definite (semidefinite)
$\mathbf{M}\prec 0 (\preceq 0)$	$\mathbf{M}$ is negative definite (semidefinite)
$\sigma_{\max}(\mathbf{M})$	maximum singular value of $\mathbf{M}$
$\|\|\mathbf{K}\|\|_F$	Frobenius norm of the matrix $\mathbf{K}$, defined by $\sqrt{\text{trace}(\mathbf{K}^{\text{T}}\mathbf{K})}$.
$\text{card}(\mathbf{K})$	Cardinality of matrix $\mathbf{K}$, defined by the number of nonzero elements in $\mathbf{K}$.
$\nabla_{\mathbf{K}} J(\mathbf{K})$	The gradient of the scalar function $J(\mathbf{K})$ with respect to the matrix $\mathbf{K}$. Assuming $\mathbf{K} \in \mathbb{R}^{m \times n}$, $\nabla_{\mathbf{K}} J(\mathbf{K})$ is given by a $m \times n$ matrix with the elements $[\nabla_{\mathbf{K}} J(\mathbf{K})]_{ij} = \partial J / \partial K_{ij}$.
$[\mathbf{K}]_s$	The matrix obtained by preserving only the s largest-magnitude entries of the matrix $\mathbf{K}$ and setting all other entries to zero.

Algorithm 1 PALM algorithm for the mixed H_2/H_∞ control algorithm with sparsity constraint

Given s: sparsity constraint, γ: H_∞-norm bound.
1. **Initialization**:
$\mathbf{K}^0$: any stabilizing feedback gain with $T_\infty(\mathbf{K}^0) < \gamma$.
$\mathbf{F}^0$: any stabilizing feedback gain $\mathbf{F}^0$.
γ_1, γ_2: constants greater than 1.
Compute $a := \gamma_1 \rho$, $b := \gamma_2 \rho$.
for $k = 1, 2, ...k_{\max}$ until $||\mathbf{K}^{k+1} - \mathbf{K}^k||_F < \epsilon_1$ or $||\mathbf{F}^{k+1} - \mathbf{F}^k||_F < \epsilon_2$ **do**
  // 2. $\mathbf{F}$-minimization step
  2.1 Compute $\mathbf{Z}^k := \mathbf{F}^k - \frac{1}{a}\nabla_{\mathbf{F}} H(\mathbf{K}^k, \mathbf{F}^k)$
  2.2 Prune $\mathbf{Z}^k$: $\mathbf{F}^{k+1} := [\mathbf{Z}^k]_s$.
  // 3. $\mathbf{K}$-minimization step
  3.1 Compute $\mathbf{X}^k := \mathbf{K}^k - \frac{1}{b}\nabla_{\mathbf{K}} H(\mathbf{K}^k, \mathbf{F}^{k+1})$.
  3.2 Update $\mathbf{K}^{k+1}$: $\mathbf{K}^{k+1} := \text{KPROXOP}(\mathbf{K}^k, \mathbf{X}^k, b)$.
end for

where $\mathbf{Z}^k$ is computed in 2.1 in Algorithm 1. As shown in [7], [8], the solution to (14) is $[\mathbf{Z}^k]_s$ (see Table I), which is Step 2.2 of Algorithm 1. The $\mathbf{K}$-minimization step reduces to

$$\mathbf{K}^{k+1} = \arg\min_{\mathbf{K}} \underbrace{(J(\mathbf{K}) + \frac{b_k}{2}||\mathbf{K} - \mathbf{X}^k||_F^2)}_{h(\mathbf{K})}$$
$$\text{s.t. } T_\infty(\mathbf{K}) < \gamma, \tag{15}$$

where $\mathbf{X}^k$ is computed in 3.1 in Algorithm 1. The minimization of (15) is performed in the subroutine KPROXOP in Algorithm 4.7 of [9], which utilizes the LMI condition in [6] to find an improving feasible direction of $\mathbf{K}$. The derivation of the Algorithm 1 and KPROXOP is contained in Chapters 4.2.2 to 4.2.3 in [9].

D. Proof of Global Convergence of Algorithm 1

Definition 1 (Proximal Map): Given a proper, lower semicontinuous function $\sigma : \mathbb{R}^d \to (-\infty, \infty], \mathbf{x} \in \mathbb{R}^d, t > 0$, the proximal map associated with σ at point $\mathbf{x}$ [10]:

$$\text{prox}_t^\sigma(\mathbf{x}) = \arg\min_{\mathbf{u}\in\mathbb{R}^d} \left\{ \sigma(\mathbf{u}) + \frac{t}{2}||\mathbf{u} - \mathbf{x}||^2 \right\}. \tag{16}$$

Algorithm 1 minimizes (10) by alternating the minimization on the proximal maps associated with the variables $\mathbf{K}$ and $\mathbf{F}$:

$$\mathbf{F}^{k+1} \in \text{prox}_{a_k}^f \left(\mathbf{F}^k - 1/a_k \nabla_{\mathbf{F}} H(\mathbf{K}^k, \mathbf{F}^k)\right) \tag{17}$$
$$\mathbf{K}^{k+1} \in \text{prox}_{b_k}^{J+g} \left(\mathbf{K}^k - 1/b_k \nabla_{\mathbf{K}} H(\mathbf{K}^k, \mathbf{F}^{k+1})\right) \tag{18}$$

where a_k and b_k are positive constants that are greater than the Lipschitz constants of $\nabla_{\mathbf{F}} H(\mathbf{K}^k, \mathbf{F})$ and $\nabla_{\mathbf{K}} H(\mathbf{K}, \mathbf{F}^{k+1})$, respectively. The definitions of *Lipschitz continuous/Lipschitz constant*, *proper function*, *lower-semicontinuous function* can be found in Definitions B.3.1, 1.2.2 and 1.2.3 of [9], respectively.

Theorem 1 (Convergence of Algorithm 1): The sequence $(\mathbf{F}^k, \mathbf{K}^k)$ generated by (17)-(18) globally converges to a critical point of Φ in (10).

Remark 1: The definition of a critical point is given in [8]. Achieving a critical point of Φ is a necessary condition for achieving optimality in (9).

The proof of Theorem 1 is based on Lemmas 1–2 and [8]. To simplify notation, we define $\tilde{g}(\mathbf{K}) \triangleq J(\mathbf{K}) + g(\mathbf{K})$ (see (7), (11)).

Lemma 1: $\tilde{g} : \mathbb{R}^{m\times n} \to (-\infty, \infty]$ and $f : \mathbb{R}^{m\times n} \to (-\infty, \infty]$ (12) are proper and lower semicontinuous functions, and $H : \mathbb{R}^{m\times n} \times \mathbb{R}^{m\times n} \to \mathbb{R}$ (13) is a continuously differentiable function, i.e., $H \in C^1$.

Proof: Since $|J(\mathbf{K})| < \infty$ if $\mathbf{K}$ is stabilizing, the function J is proper. In addition, since $J(\mathbf{K})$ is continuous in $\mathbf{K}$ [11], it is lower semicontinuous. Similar arguments show that $g(\mathbf{K})$ and $f(\mathbf{F})$ are proper and $g(\mathbf{K})$ is lower semicontinuous. Thus $\tilde{g}(\mathbf{K})$ is proper and lower semicontinuous. Moreover, $f(\mathbf{F})$ is lower semicontinuous [8]. Finally, the gradient of $H(\mathbf{K}, \mathbf{F})$ (see eq. (4.25) in [9]) is continuous in $\mathbf{K}, \mathbf{F}$. Thus, $H \in C^1$. ■

Assumption 3: Function J is a semi-algebraic function [8].

Lemma 2: The properties (i–iv), which are stated as assumptions in [8], hold for the proposed optimization (10): (i) $\inf_{\mathbb{R}^{m\times n},\mathbb{R}^{m\times n}} \Phi > -\infty$, $\inf_{\mathbb{R}^{m\times n}} f > -\infty$, and $\inf_{\mathbb{R}^{m\times n}} \tilde{g} > -\infty$, where Φ is given by (10).
(ii) The partial gradients $\nabla_{\mathbf{K}} H(\mathbf{K}, \mathbf{F})$ and $\nabla_{\mathbf{F}} H(\mathbf{K}, \mathbf{F})$ are globally Lipschitz with Lipschitz constants $L_1(\mathbf{F})$ and $L_2(\mathbf{K})$ [8].
(iii) There exist bounds λ_i^-, λ_i^+, $i = 1, 2$ such that

$$\inf\{L_1(\mathbf{F}^k) : k \in \mathbb{N}\} \geq \lambda_1^-, \inf\{L_2(\mathbf{K}^k) : k \in \mathbb{N}\} \geq \lambda_2^-$$
$$\sup\{L_1(\mathbf{F}^k) : k \in \mathbb{N}\} \leq \lambda_1^+, \sup\{L_2(\mathbf{K}^k) : k \in \mathbb{N}\} \leq \lambda_2^+$$

(iv) $\nabla H \triangleq (\nabla_{\mathbf{K}} H, \nabla_{\mathbf{F}} H)$ is Lipschitz continuous [10] on bounded subsets of $\mathbb{R}^{m\times p} \times \mathbb{R}^{m\times p}$.
(v) Under Assumption 3, the objective function Φ of (10) is a Kurdyka-Łojasiewicz (KL) function [8].

Proof: The proof is contained Lemma B.5.3–B.5.4 of [9] and is omitted due to space constraints. ■

Remark 2: The definitions of semi-algebraic set and function, and Kurdyka-Łojasiewicz (KL) function can be found in Definitions B.4.4 and B.4.3 of [9], respectively. Moreover, a broad class of functions satisfy the semi-algebraic property, which aids the proof of Lemma 2 and supports Assumption 3 [8].

Using the arguments in [8], we can show that if Lemmas 1–2 hold, then the sequence generated by PALM algorithm globally converges to a critical point [8] of Φ. This confirms convergence of Algorithm 1 to a sparsity-constrained mixed H_2/H_∞ controller, which corresponds to a critical point of Φ under mild assumptions on the functions J and g.

III. SPARSITY-CONSTRAINED NONCOOPERATIVE GAMES FOR MULTI-AGENT CONTROL

A. Multi-agent model and the noncooperative game

Consider a network of N agents, where agent i employs control strategy $\mathbf{u}_i(t)\in\mathbb{R}^{q_i\times 1}$, corresponding to its sub-block of the overall feedback matrix $\mathbf{K}$, i=1...N. Thus, (1) can be represented as

$$\dot{\mathbf{x}}(t)=(\mathbf{A}+\Delta\mathbf{A})\mathbf{x}(t)+\sum_{i=1}^{N}\left(\mathbf{B}_{(i)}+\Delta\mathbf{B}_{(i)}\right)\mathbf{u}_i(t)+\mathbf{B}_2\mathbf{w}_2(t)$$
$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t)$$
$$\mathbf{u}_i(t) = -\mathbf{K}_i\mathbf{y}(t),\ i = 1, ..., N. \tag{19}$$

where $\mathbf{A}\in\mathbb{R}^{n\times n}$, $\mathbf{B}_{(i)}\in\mathbb{R}^{n\times q_i}$ represent the nominal values of the state and control matrix, respectively, for the i-th control input. Due to the offline design of the sparse controller, we assume all agents know $\mathbf{A}$ and $\mathbf{B}_{(i)}$ for i=1, ..., N. The uncertain matrices $\Delta\mathbf{A} \in \mathbb{R}^{n\times n}$ and $\Delta\mathbf{B} := [\Delta\mathbf{B}_{(1)}, \Delta\mathbf{B}_{(2)}, ..., \Delta\mathbf{B}_{(N)}]$ satisfy (2), where $\Delta\mathbf{B}_{(i)}\in\mathbb{R}^{q_i\times n}$. Note that $\mathbf{B}_{(i)}$ is the column block of $\mathbf{B}$ in (1), with $\sum_{i=1}^{N}\mathbf{B}_{(i)}\mathbf{u}_i$=$\mathbf{B}\mathbf{u}$, and $\mathbf{K}_i \in \mathbb{R}^{q_i\times p}$ is the row block of $\mathbf{K}$ associated with the rows corresponding to the control inputs for agent i. Thus, (19) can be expressed in the form (4–5), with the first equation in (4) replaced by

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \sum_{i=1}^{N}\mathbf{B}_{(i)}\mathbf{u}_i(t) + \mathbf{B}_1\mathbf{w}_1(t) + \mathbf{B}_2\mathbf{w}_2(t). \tag{20}$$

Let $\mathbf{K}_{-i}$ denote the set of strategies $j \neq i, j = 1, ..., N$. When agent i chooses its strategy $\mathbf{K}_i$ in (19) given $\mathbf{K}_{-i}$, we refer to the resulting feedback gain matrix $\mathbf{K}$ as $\{\mathbf{K}_i; \mathbf{K}_{-i}\}$.

In (20), the single performance output $\mathbf{z}_2$ in (1) is replaced by N individual performance outputs of the agents $\mathbf{z}_{2,(i)}$. Assuming that each performance output $\mathbf{z}_{2,(i)} = \mathbf{C}_{2,(i)}\mathbf{x} + \mathbf{D}_{2,(i)}\mathbf{u}_i$ has a form that satisfies (3), the H_2-cost from $\mathbf{w}_2$ to agent i's performance output can equivalently be defined as the individual LQR cost of agent i:

$$J_i(\mathbf{K}) = \int_{t=0}^{\infty} \left[\mathbf{x}^T(t)\mathbf{Q}_i\mathbf{x}(t) + \mathbf{u}_i^T(t)\mathbf{R}_i\mathbf{u}_i(t)\right] dt$$
$$\text{s.t. } \mathbf{w}_1(t) = \mathbf{0}, \mathbf{w}_2(t) = \boldsymbol{\delta}(t) \tag{21}$$

where $\mathbf{Q}_i \in \mathbb{R}^{n\times n} \succeq 0$ and $\mathbf{R}_i \in \mathbb{R}^{q_i\times q_i} \succ 0$ are weight matrices for state and control input of agent i, respectively, and $\mathbf{w}_2(t)$ is an impulse disturbance.

Given other players' strategies $\mathbf{K}_{-i}$, the set of feasible strategies for player i, which guarantee stability of the uncertain system (19) with at most s communication links overall, must satisfy

$$\mathcal{G}_i(\mathbf{K}_{-i}) = \{\mathbf{K}_i | \ \text{card}(\{\mathbf{K}_i; \mathbf{K}_{-i}\}) \leq s, \quad T_\infty(\{\mathbf{K}_i; \mathbf{K}_{-i}\}) < \gamma\}. \tag{22}$$

Given $\mathbf{K}_{-i}$, player i solves the following optimization:

$$\min_{\mathbf{K}_i} J_i(\{\mathbf{K}_i; \mathbf{K}_{-i}\}) \quad \text{s.t. } \mathbf{K}_i \in \mathcal{G}_i(\mathbf{K}_{-i}). \tag{23}$$

We define a relaxed version of the Nash Equilibrium termed *Approximate Local Equilibrium (ALE)*, which is similar to the smoothed local equilibrium concept in [12]. In ALE, any player i cannot make ϵ-deviations to improve its utility given other players' strategies. ALE is computationally tractable since it is quantified by the projected gradient, which is suitable for constrained optimization problems. Thus ALE is a more practical equilibrium concept for the proposed game than the Generalized Nash Equilibrium (GNE) [13].

Definition 2 (Projected Gradient [12]): Let $\eta > 0$. The projected gradient of cost J_i onto the constraint set $\mathcal{G}_i(\mathbf{K}^*_{-i})$ of player i is defined as

$$\nabla_{\mathcal{G}_i(\mathbf{K}^*_{-i}),\eta} J_i(\{\mathbf{K}_i; \mathbf{K}^*_{-i}\}) = \frac{1}{\eta}(\mathbf{K}_i - \Pi_{\mathcal{G}_i(\mathbf{K}^*_{-i})}[\mathbf{K}_i - \eta\nabla_{\mathbf{K}_i} J_i(\{\mathbf{K}_i, \mathbf{K}^*_{-i}\})]), \tag{24}$$

where the operator $\Pi_{\mathcal{K}}(\cdot)$ denotes projection onto set $\mathcal{K}$.

Definition 3 (Approximate Local Equilibrium): A set of strategies $(\mathbf{K}^*_1, \mathbf{K}^*_2, ..., \mathbf{K}^*_N))$ is an ϵ-approximate local equilibrium if

$$||\nabla_{\mathcal{G}_i(\mathbf{K}^*_{-i}),\eta} J_i(\{\mathbf{K}_i; \mathbf{K}^*_{-i}\})|| < \epsilon, \ \forall i = 1, 2, ..., N. \tag{25}$$

Note that when ϵ=0 in (25), the ALE point achieves a necessary condition for GNE [12].

B. PALM algorithm for computing an ALE

Algorithm 2 employs the best-response (BR) dynamics (23) to find an ALE. Recall Algorithm 1 where the tuple $\mathbf{K}, \mathbf{F}$ was iteratively optimized to solve the penalized optimization (10). Similarly, given $\mathbf{K}_{-i}$, player i's optimization (23) can be written in the penalized form using indicator functions as

$$\min_{\mathbf{K}_i, \mathbf{F}} \Phi_i(\mathbf{K}_i, \mathbf{F}; \mathbf{K}_{-i}), \ \text{with} \tag{26}$$

$$\Phi_i(\mathbf{K}_i, \mathbf{F}; \mathbf{K}_{-i}) := J_i(\{\mathbf{K}_i; \mathbf{K}_{-i}\}) + h(\{\mathbf{K}_i; \mathbf{K}_{-i}\}) + f(\mathbf{F}) + H(\{\mathbf{K}_i; \mathbf{K}_{-i}\}, \mathbf{F}) \tag{27}$$

where $h(\cdot)$ and $f(\cdot)$ are given by (11,12) and the matrix $\{\mathbf{K}_i; \mathbf{K}_{-i}\}$ is defined after (20). In (26), $\mathbf{K}_i \in \mathbb{R}^{q_i\times n}$ represents the feedback gain of agent i that satisfies $T_\infty(\{\mathbf{K}_i; \mathbf{K}_{-i}\}) < \gamma$ and $\mathbf{F} \in \mathrm{R}^{m\times n}$ is the system-wide sparse feedback gain matrix that satisfies the global sparsity constraint. The global constraint is appropriate since the agents share the communication network. In the BR dynamics, in each round the players take turns to minimize their own respective Φ_i functions over $\mathbf{K}_i$ and $\mathbf{F}$. The equilibrium point is achieved when no player can improve its Φ_i using $\mathbf{K}_i$ and $\mathbf{F}$ while $\mathbf{K}_j$ is fixed for $j \neq i$.

The minimization (26) is similar to (10). Thus, modified Algorithm 1 is used in line 9 of Algorithm 2 to solve (26). Given its $\mathbf{K}_i^l$, $\mathbf{F}^l$ at iteration l, the following proximal operators are employed by player i in the minimization of line 9 in Algorithm 2.

$\mathbf{F}$*-minimization:* Compute the proximal point $\mathbf{Z}^k$ for $\mathbf{F}^k$:

$$\mathbf{Z}^k = \mathbf{F}^k - \frac{1}{a}\nabla_{\mathbf{F}} H(\mathbf{K}^k, \mathbf{F}^k) = \mathbf{F}^k - \frac{\gamma}{a}(\mathbf{F}^k - \{\mathbf{K}_i^k; \mathbf{K}_{-i}\}) \tag{28}$$

Solve for the proximal operator:

$$\mathbf{F}^{k+1} = \arg\min_{\mathbf{F}} \frac{a}{2}||\mathbf{F} - \mathbf{Z}^k||_F^2, \ \text{s.t.card}(\mathbf{F}) \leq s, \tag{29}$$

and get $\mathbf{F}^{k+1} = [\mathbf{Z}^k]_s$, similarly to Step 2 of Algorithm 1.

$\mathbf{K}$*-minimization:* Compute the proximal point $\mathbf{X}_i^k$ for $\mathbf{K}_i$:

$$\mathbf{X}_i^k = \mathbf{K}_i^k - \frac{1}{b}\nabla_{\mathbf{K}_i} H(\mathbf{K}_i^k - (\mathbf{F}^{k+1})_i) = \mathbf{K}_i^k - \frac{\rho}{b}(\mathbf{K}_i^k - (\mathbf{F}^{k+1})_i). \tag{30}$$

Solve for the proximal operator:

$$\mathbf{K}_i^{k+1} = \arg\min_{\mathbf{K}_i} \left\{ J_i(\{\mathbf{K}_i; \mathbf{K}_{-i}\}) + \frac{b}{2}||\mathbf{K}_i - \mathbf{X}_i^k||_F^2 \right\} \quad \text{s.t. } T_\infty(\{\mathbf{K}_i; \mathbf{K}_{-i}\}) < \gamma. \tag{31}$$

Similarly to (15), we solve (31) by applying the KPROXOP subroutine (see Algorithm 1). In the player i's optimization to solve line 9 of Algorithm 2, its individual control strategy is first updated in the $\mathbf{K}$-minimization step. Then this player informs all players to jointly update their strategies in the $\mathbf{F}$-minimization. Each player sends the updated control strategies to the other players. In summary, Algorithm 2 has *partially distributed computation* where the players update their control strategies individually but require the nominal system knowledge and communicate with the other players.

Finally, we note that partially distributed implementation of the social optimization (6) is obtained using the noncooperative game above where each agent's utility (23) is replaced with the objective $J(\mathbf{K})$ in (7) expressed as (32), where $\mathbf{R}(i)$ denotes the submatrix of $\mathbf{R}$ that represents the weight matrix for $\mathbf{u}_i(t)$ and $\mathbf{Q}_i = \mathbf{Q} + \mathbf{C}^T(\sum_{j\neq i} \mathbf{K}_j^T \mathbf{R}_{(j)} \mathbf{K}_j)\mathbf{C}$.

$$J(\mathbf{K}) = J(\{\mathbf{K}_i, \mathbf{K}_{-i}\}) = \int_0^\infty [\mathbf{x}^T(\mathbf{Q}+\mathbf{C}^T(\sum_{j\neq i} \mathbf{K}_j^T \mathbf{R}_{(j)} \mathbf{K}_j)\mathbf{C})\mathbf{x}+\mathbf{u}_i^T \mathbf{R}_{(i)} \mathbf{u}_i] dt. \tag{32}$$

Similarly to GNE, existence of ALE and convergence of Algorithm 2 are not guaranteed due to the nonconvex optimization objectives [13]. If Algorithm 2 converges to an ALE (25) at iteration l, ϵ is approximated by $\frac{1}{\eta}||\Delta \mathbf{K}_i^l||$, where η is the step size determined in the KPROXOP subroutine in Algorithm 1. Finally, the *numerical complexity* of Algorithms 1 and 2 is dominated by the $\mathbf{K}$-minimization step (Step 3 of Algorithm 1 and line 9 of Algorithm 2), which has *polynomial* complexity on the number of variables in the feedback matrix [14].

IV. Numerical results

We consider a standard example of an uncertain network model [15], [16] that consists of N connected nodes distributed randomly on a L by L square units area. Each node is an unstable second-order system coupled with other nodes through an exponentially decaying function of the Euclidean distance $\hat{l}(i,j)$ where $i, j = 1, ..., N$ [15]. The state-space representation of node i is given as:

$$\begin{bmatrix} \dot{x}_{1i} \\ \dot{x}_{2i} \end{bmatrix} = \hat{\mathbf{A}}_{ii} \begin{bmatrix} x_{1i} \\ x_{2i} \end{bmatrix} + \sum_{j\neq i} e^{-\hat{l}(i,j)} \begin{bmatrix} x_{1j} \\ x_{2j} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} (u_i + w_i). \tag{33}$$

Similarly to the system (1), the state matrix $\hat{\mathbf{A}} \in \mathbb{R}^{2N\times 2N}$ includes parametric uncertainty, with matrix $\mathbf{A}$ denoting the known nominal value of the state matrix. The state matrices within each node are given by $\hat{\mathbf{A}}_{ii}$, $i = 1, ..., N$ and the Euclidean distances between nodes i and j, $\hat{l}(i,j)$ are modeled as

$$\begin{aligned} \hat{\mathbf{A}}_{ii} &= \mathbf{A}_{ii} + \mathbf{A}_{ii} \odot \begin{bmatrix} \theta_{11} & \theta_{12} \\ \theta_{21} & \theta_{22} \end{bmatrix} \\ \hat{l}(i,j) &= l(i,j) \cdot (1 + \delta_{i,j}), \forall i, j = 1, ..., N \end{aligned} \tag{34}$$

Algorithm 2 PALM algorithm for computing ALE (25)

1: **Given** s: global sparsity constraint, γ: H_∞-norm bound.
2: **Initialization**:
3: $\mathbf{K}^0$: any stabilizing feedback gain with $T_\infty(\mathbf{K}^0) < \gamma$.
4: $\mathbf{F}^0$: any stabilizing feedback gain $\mathbf{F}^0$.
5: **for** $l = 1...l_{\max}$ until $||\mathbf{F}^1 - \mathbf{F}^{l-1}||_F < \epsilon_3$ **do**
6: $\quad\mathbf{K}^l := \mathbf{K}^{l-1}$, $\mathbf{F}^l := \mathbf{F}^{l-1}$
7: $\quad$**for** $i = 1...N$ **do**
8: $\quad\quad$// Solve using (28–31) with $\mathbf{K}_i^l$, $\mathbf{F}^l$ as the initial values:
9: $\quad\quad\hat{\mathbf{K}}_i, \hat{\mathbf{F}} = \arg\min_{\mathbf{K}_i, \mathbf{F}} \Phi_i(\mathbf{K}_i, \mathbf{F}; \mathbf{K}_{-i}^l)$
10: $\quad\quad$// Update $\mathbf{K}^l$ and $\mathbf{F}^l$:
11: $\quad\quad\mathbf{K}^l = \{\hat{\mathbf{K}}_i; \mathbf{K}_{-i}^l\}$
12: $\quad\quad\mathbf{F}^l = \hat{\mathbf{F}}$
13: $\quad$**end for**
14: **end for**
15: **Output:** $\mathbf{K}^{\mathrm{ALE}}(s) := \mathbf{F}^l$.

where $\mathbf{A}_{ii}$ and $l(i,j)$ are the nominal values, and δ_{ij} and $\theta_{mn} \forall m, n = 1, 2$ are independent random perturbations, uniformly distributed in the range $\pm 20\%$. The operator $\odot$ denotes element-wise multiplication. The input matrix $\hat{\mathbf{B}}$ is known exactly, i.e., $\hat{\mathbf{B}} = \mathbf{B} = \mathbf{1}_N \otimes [0\ 1]^T$, where $\otimes$ denotes the Kronecker product. In this simulation study, we collected 200 random samples of $\hat{\mathbf{A}}$. To guarantee closed-loop stability of (33), we numerically compute the worst-case $\hat{\boldsymbol{A}}$ as $\hat{\boldsymbol{A}}_{\mathrm{worst}} = \arg\max_{\hat{\boldsymbol{A}}} \sigma_{\max}(\hat{\boldsymbol{A}} - \boldsymbol{A})$. Using the singular value decomposition, we obtain $\boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T = \hat{\boldsymbol{A}}_{\mathrm{worst}} - \boldsymbol{A}$. Normalizing $\boldsymbol{S}$ by $\sigma_{\max}(\boldsymbol{S})$, we set $\boldsymbol{B}_1 = \sqrt{\sigma_{\max}(\boldsymbol{S})}\boldsymbol{U}$, $\boldsymbol{C}_1 = \sqrt{\sigma_{\max}(\boldsymbol{S})}\boldsymbol{V}^T$ in (2). Due to this normalization, $\gamma = 1$.

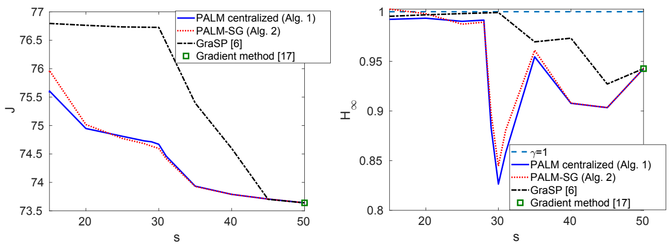
We set $L = 2$, $N = 5$. Thus, $\mathbf{A} \in \mathbb{R}^{10\times 10}$, $\mathbf{B} \in \mathbb{R}^{10\times 5}$. The output matrix $\mathbf{C} = \mathbf{I}_{10}$. The dense feedback matrix $\mathbf{K}$ has $\mathrm{card}(\mathbf{K}) = 50$ while the completely decentralized feedback controller has $\mathrm{card}(\mathbf{K}) = 10$. We set $\mathbf{Q} = 100 \cdot \mathbf{I}$ and $\mathbf{R} = \mathbf{I}$ in (8). The *noncooperative game* with the agents' utilities in (21) has two players, where player 1 is in charge of the control inputs in nodes 1 and 3 and player 2 is in charge of the control inputs in nodes 2, 4, 5. The parameters $\rho = 1000$ (9), and $\epsilon_1 = \epsilon_2 = 10^{-4}$, $\epsilon_3 = 10^{-3}$ in Algorithms 1 and 2 were chosen experimentally to aid accuracy and convergence. The performance index matrices $\mathbf{Q}_i, \mathbf{R}_i$, $i = 1, 2$ for the LQR cost in (21) satisfy:

$$\begin{aligned} \mathbf{x}^T \mathbf{Q}_1 \mathbf{x} + \mathbf{u}_1^T \mathbf{R}_1 \mathbf{u}_1 &= 100 \sum_{i=1,2} (x_{i1} - x_{i3})^2 + \sum_{j=1,3} u_j^2 \\ \mathbf{x}^T \mathbf{Q}_2 \mathbf{x} + \mathbf{u}_2^T \mathbf{R}_2 \mathbf{u}_2 &= 100 \sum_{j=2,4,5} (x_{1j}^2 + x_{2j}^2) + \sum_{j=2,4,5} u_j^2. \end{aligned} \tag{35}$$

First, we present simulation results for the social optimization problem (6). We compare the following controllers: $\mathbf{K}^*_{\mathrm{PALM}}(s)$ computed by Algorithm 1, $\mathbf{K}^*_{\mathrm{PALM-SG}}(s)$ computed by Algorithm 2 with the agents' objectives (32), $\mathbf{K}^*_{\mathrm{GraSP}}(s)$ computed from the GraSP algorithm in [6], and the dense mixed H_2/H_∞ controller using the simple gradient method in [17].

Figure 1 illustrates the optimal LQR cost J in problem (6) and the associated H_∞ norm vs. sparsity constraint s. From Figure 1(a), we observe that the LQR costs of all sparsity-constrained methods decrease as s is relaxed and approach that of the dense controller. However, the PALM-based methods have similar LQR costs and *outperform significantly the GraSP algorithm* in [6]. At convergence, GraSP algorithm determines the sparsity structure given by the *greedy* selection step and does not necessarily find the critical point of the problem in (6) achieved by the PALM method (Theorem 1).

Next, we investigate performance of the noncooperative game solved using Algorithm 2. Let $\mathbf{K}^{\mathrm{ALE}}(s) = \{\mathbf{K}_1^{\mathrm{ALE}}(s), \mathbf{K}_2^{\mathrm{ALE}}(s)\}$ denote the feedback gain matrices for the two players produced by Algorithm 2 when the sparsity constraint is given by s. Figure 2(a) shows the errors in consecutive steps of player i's strategic variables $\mathbf{K}_i, \mathbf{F}_i$ for $i = 1, 2$ vs iteration l in Algorithm 2. We observe that both $||\Delta \mathbf{K}_i||_F$ and $||\Delta \mathbf{F}_i||_F$ decrease significantly within the first 10 iterations, and saturate to approximately 10^{-2} as l grows,



(a) J vs. sparsity constraint s. (b) H_∞ norm v.s. sparsity constraint s.

Fig. 1: Performance of social optimization methods.

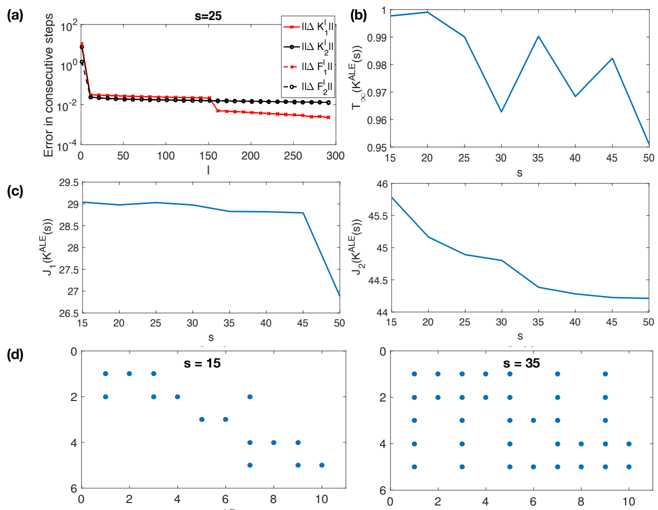


Fig. 2: Performance of Algorithm 2 for noncooperative games. (a) Errors in consecutive steps of $\mathbf{K}_i^l$ and $\mathbf{F}_i^l$ for players $i=1,2$ vs. step l for $s=25$. (b) $T_\infty(\mathbf{K}^{\mathrm{ALE}}(s))$ vs. the sparsity constraint s at ALE. (c) $J_i(\mathbf{K}^{\mathrm{ALE}}(s))$ vs. sparsity constraint s at ALE. (d) The sparsity pattern (nonzero elements) of $\mathbf{K}^{\mathrm{ALE}}(s)$ for different s values.

resulting in the saturation of the penalized cost function Φ_i in line 9 of Algorithm 2. While $||\Delta\mathbf{K}_i||_F$ does not decrease to zero to reach a necessary condition for a GNE, the saturation point represents an ALE (25). Figure 2(b) shows that $T_\infty(\mathbf{K}^{\mathrm{ALE}}(s)) < 1$ for $15 \le s \le 45$, indicating that the strategies in $\mathbf{K}^{\mathrm{ALE}}(s)$ are guaranteed to stabilize the uncertain system (33). Figure 2(c) illustrates the individual LQR costs J_i (21) when $\mathbf{K}^{\mathrm{ALE}}(s)$ is applied. Note that the LQR cost of each player achieved at the equilibrium point decreases with s, demonstrating the trade-off between the selfish LQR cost and the global shared sparsity constraint. Finally, Figure 2(d) illustrates the non-zero elements of the feedback matrix for several s-values, demonstrating that the feedback matrix tends to that of the decentralized controller but still retains some "important" links between the nodes as the communication cost grows.

V. Conclusion

The PALM method was employed to solve the sparsity-constrained mixed H_2/H_∞ control problem for multi-agent systems. First, a centralized social-optimization algorithm was investigated. Second, we developed a noncooperative game with partially distributed computation. The proposed algorithms were validated using an open-loop-unstable network dynamic system. It was demonstrated that the centralized PALM method outperforms the GraSP-based method for most sparsity levels and converges both theoretically and in simulation results. Moreover, our numerical results illustrate convergence of the proposed best-response-dynamics algorithm for the noncooperative game to an approximate local equilibrium point for most sparsity constraint values. Finally, the performance of the game-based partially distributed algorithm for social optimization closely approximates that of the centralized algorithm.

References

[1] C. Lidström and A. Rantzer, "Optimal H_∞ state feedback for systems with symmetric and hurwitz state matrix," in *American Control Conference*, 2016, pp. 3366–3371.

[2] M. Bahavarnia and N. Motee, "Sparse memoryless LQR design for uncertain linear time-delay systems," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 10 395–10 400, 2017.

[3] E. J. LoCicero and L. Bridgeman, "Sparsity promoting H_2-conic control," *IEEE Control Systems Letters*, vol. 5, no. 4, pp. 1453–1458, 2020.

[4] C. Colson, M. Nehrir, and R. Gunderson, "Distributed multi-agent microgrids: a decentralized approach to resilient power system self-healing," in *2011 4th International Symposium on Resilient Control Systems*. IEEE, 2011, pp. 83–88.

[5] J. Kamel, R. Vosooghi, J. Puchinger, F. Ksontini, and G. Sirin, "Exploring the impact of user preferences on shared autonomous vehicle modal split: A multi-agent simulation approach," *Transportation Research Procedia*, vol. 37, pp. 115–122, 2019.

[6] F. Lian, A. Chakrabortty, F. Wu, and A. Duel-Hallen, "Sparsity-constrained mixed H_2/H_∞ control," in *American Control Conference (ACC)*, 2018.

[7] F. Lin and V. Adetola, "Co-design of sparse output feedback and row/column-sparse output matrix," in *American Control Conference (ACC), 2017*. IEEE, 2017, pp. 4359–4364.

[8] J. Bolte, S. Sabach, and M. Teboulle, "Proximal alternating linearized minimization or nonconvex and nonsmooth problems," *Mathematical Programming*, vol. 146, no. 1-2, pp. 459–494, 2014.

[9] F. Lian, "Communication-cost-constrained algorithms and games for multi-agent control systems," Ph.D. dissertation, North Carolina State University, Raleigh, NC, USA, 2019.

[10] N. Parikh, S. Boyd *et al.*, "Proximal algorithms," *Foundations and Trends® in Optimization*, vol. 1, no. 3, pp. 127–239, 2014.

[11] 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.

[12] E. Hazan, K. Singh, and C. Zhang, "Efficient regret minimization in non-convex games," *arXiv preprint arXiv:1708.00075*, 2017.

[13] D. Paccagnan, B. Gentile, F. Parise, M. Kamgarpour, and J. Lygeros, "Distributed computation of generalized Nash equilibria in quadratic aggregative games with affine coupling constraints," in *Decision and Control (CDC), 2016 IEEE 55th Conference on*. IEEE, 2016, pp. 6123–6128.

[14] P. Gahinet, A. Nemirovskii, A. J. Laub, and M. Chilali, "The LMI control toolbox," in *Proceedings of 1994 33rd IEEE Conference on Decision and Control*, vol. 3. IEEE, 1994, pp. 2038–2041.

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

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

[17] Y. Kami and E. Nobuyama, "A gradient method for the static output feedback mixed H_2/H_∞ control," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 7838–7842, 2008.