Communication Delay Co-Design in $\mathcal{H}_2$ Decentralized Control Using Atomic Norm Minimization

Nikolai Matni

***Abstract*— It has been shown that $\mathcal{H}_2$ decentralized control subject to delay constraints induced by a strongly connected communication graph can be solved by decomposing the controller into a centralized but delayed component, and a decentralized finite impulse response component, the latter of which can be solved for via a linearly constrained quadratic program. In this paper, we propose an atomic norm minimization based variant of this quadratic program that can be used for the co-design of a communication graph that is well suited for the control task.**

I. INTRODUCTION

Decentralized control problems arise when several decision makers, or controllers, need to determine their actions based only on a subset of the total information available about the system. These types of problems arise in areas as diverse as physiology, economics and the power grid.

In the past decade, this field has seen an explosion of advances at the theoretical, algorithmic and practical levels. We provide a brief survey of results most relevant to our paper in the following, and refer the reader to the excellent tutorial paper [1] for a thorough and timely presentation of the current state of the art in optimal decentralized control subject to information constraints.

A particular class of decentralized control problems that has received a significant amount of attention is that of optimal $\mathcal{H}_2$ (or LQG) control subject to delay constraints. In this case, the information constraints can be interpreted as arising from a communication graph, in which edge weights between nodes correspond to the delay required to transmit information between them. In the case where the underlying communication graph induces a quadratically invariant (QI) delay pattern, this problem has been solved via vectorization [2], semi-definite programming (SDP) [3], [4] , and most recently using an extension of spectral factorization [5].

As promising and impressive as the above results for controller synthesis have been, they all make several assumptions on the communication graph. The first is that bit-rate limits and quantization have negligible effects on performance, and that the defining characteristics of a communication network, at least from a controls perspective, are its end to end transmission and encoding/decoding delays. Results from Networked Control Systems (NCS) theory indicate that indeed this is in general a reasonable assumption (c.f [6] for a recent survey of these results).

Another assumption is that these delays are *fixed*, which necessarily introduces a level of conservatism in the control design procedure. In particular, to ensure that the delays with which controllers can exchange information do not vary, worst case delay times must be used in the control design. In [7], we make a first step towards relaxing this assumption in a decentralized optimal control setting by explicitly accounting for varying delays in the control design procedure.

Stronger still, however, is the very assumption that a communication graph has already been designed that is well suited for decentralized optimal control. Very little results informing the design of communication graphs for control can be found in the literature. Solving for the optimal (with respect to graph complexity and control performance) communication network is inherently combinatorial in nature, and tractable methods for computing exact solutions are unlikely, if not impossible. An approach which has seen much success in similar problems in other fields has been to employ convex relaxations in order to approximately recover such solutions. Even this approach has few representatives in the control literature as applied to communication design, with the closest in spirit being: (i) [8] which uses dual variables for communication graph augmentation, and (ii) [9], which uses ℓ_1-regularization techniques to induce sparsity when computing a state-feedback controller.

The idea of using convex relaxations in optimization problems, in particular in the setting of attempting to recover structured solutions, has a rich and fruitful history in the machine learning and statistics communities. In particular, it is often known *a priori* that the solution to an optimization problem should be "simple." It has been shown that this simple structure can often be approximately, and sometimes exactly, recovered by minimizing an appropriately chosen convex penalty function. Well known examples include the ℓ_1-norm to induce sparse solutions, and the nuclear norm to induce low-rank solutions [10], [11], [12], [13], [14]. In [15], this notion of "simplicity" was formalized and generalized in terms of *atomic norms*. In addition to [9], these ideas have also been successfully applied in the context of system identification [16].

In order to apply these types of techniques, the problem to be solved must ultimately be reduced to a finite dimensional convex program. In [5], it is shown that the decentralized $\mathcal{H}_2$ problem subject to delay constraints induced by a *strongly connected* communication graph can in fact be reduced to a finite dimensional linearly constrainted quadratic program. In particular, the optimal controller is solved for by decomposing it into a centralized, but delayed, component (thus the need for strong connectivity) and a decentralized finite impulse response (FIR) component. It is this FIR component

N. Matni is with the Department of Control and Dynamical Systems, California Institute of Technology, Pasadena, CA. nmatni@caltech.edu.

that is solved for in the quadratic program.

Noting that the *entire decentralized nature* of the problem is captured in this FIR element, we borrow ideas from atomic norm mininization in order to propose a convex programming based method for inducing sparsity patterns that are *consistent* with how information propagates through communication graphs. In particular, we identify the appropriate atomic norm for inducing the desired structure in the FIR component, and formulate the decentralized controller and communication graph co-design problem as a finite dimensional second order cone program (SOCP). Additionally, we present two tractable algorithms for generating the atoms required to construct this information propagation based atomic norm.

This article is structured as follows: Section II provides a brief summary of the necessary concepts from quadratic invariance [2], the spectral factorization solution to the distributed control problem [5], and atomic norm regularization theory [15]. Section III then formally defines the problem to be solved, and presents our main result. Section IV presents two algorithms for generating the atomic sets needed to implement our algorithm. We then present two application examples in Section V, and Section VI finishes with conclusions and directions for future work.

II. Preliminaries

A. Notation

If $\mathcal{M}$ is a subspace of an inner product space, we denote the orthogonal projection onto $\mathcal{M}$ by $\mathbb{P}_{\mathcal{M}}$. We let $\otimes$ denote the Kronecker product, $\text{vec}(\cdot) : \mathbb{R}^{p\times q} \to \mathbb{R}^{pq}$ be the vectorization operator that maps a matrix to a vector through the stacking of its columns, and $\text{mirror}(\cdot) : \mathbb{R}^{p\times q} \to \mathbb{R}^{p\times q}$ as $\{\text{mirror}(A)\}_{ij} = A_{p-(i-1),q-(i-1)}$, i.e. the operator that flips elements of a matrix about the vertical and horizontal axes. We denote the support operator by $\text{supp}(\cdot) : \mathbb{R}^{p\times q} \to \mathbb{R}^{p\times q}$, where $\{\text{supp}(A)\}_{ij} = 1$ if $A_{ij} \neq 0$, and 0 otherwise.

We let $E_{ij} \in \mathbb{R}^{p\times q}$ denote the matrix with element (i, j) set to 1, and all other elements set to 0. For any set Ω, we denote by 2^Ω the power set of Ω, that is to say the set of all subsets of Ω. Finally, for $x \in \mathbb{R}$ we denote by $\lfloor x\rfloor$ and $\lceil x\rceil$ the floor and ceiling operators, respectively.

Example:

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 5 & 6 \\ 7 & 8 & 0 \end{bmatrix}, \ \text{mirror}(A) = \begin{bmatrix} 0 & 8 & 7 \\ 6 & 5 & 0 \\ 3 & 2 & 1 \end{bmatrix},$$
$$\text{supp}(A) = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

B. $\mathcal{H}_2$ Preliminaries and Notation

Let $\mathbb{D} =\{z \in \mathbb{C} : |z| < 1\}$ be the unit disc of complex numbers. A function $G : (\mathbb{C} \bigcup\{\infty\})\backslash\mathbb{D} \to \mathbb{C}^{p\times q}$ is in $\mathcal{H}_2$ if it can be expanded as

$$G(z) = \sum_{i=0}^{\infty} \frac{1}{z^i} G_i$$

where $G_i \in \mathbb{C}^{p\times q}$ and $\sum_{i=0}^{\infty} \text{Tr}(G_i G_i^*) < \infty$. Define the conjugate of G by

$$G(z)^\sim = \sum_{i=0}^{\infty} z^i G_i^*$$

$\mathcal{H}_2$ is a Hilbert space with inner product given by

$$\begin{aligned} < G, H > &= \frac{1}{2\pi}\int_{-\pi}^{\pi} \text{Tr}(G(e^{j\theta})H(e^{j\theta})^\sim)d\theta \\ &= \sum_{i=0}^{\infty} \text{Tr}(G_i H_i^*), \end{aligned}$$

where the last equality follows from Parseval's identity.

C. $\mathcal{H}_2$ optimal control subject to delays

Let P be a stable discrete-time plant given by

$$P = \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & 0 & D_{12} \\ C_2 & D_{21} & 0 \end{array}\right] = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \tag{1}$$

with inputs of dimension p_1, p_2 and outputs of dimension q_1, q_2. We restrict attention to stable plants for simplicity. These methods could also be applied to an unstable plant if a stable stabilizing nominal controller can be found, as in [2]. We note that this task may be non-trivial, with strong guarantees existing only in the sparsity constrained setting [17].

To ensure the existence of stabilizing solutions to the appropriate Riccati equations (note that stabilizability and detectability of (A, B_2, C_2) is implied by the assumption of a stable plant), we assume (i) $D_{12}^T D_{12} > 0$, (ii) $D_{21} D_{21}^T > 0$, (iii) $C_1^T D_{12} = 0$, and (iv) $B_1 D_{21}^T = 0$

For $N \geq 1$, define the space of strictly proper FIR transfer matrices by $\mathcal{X} = \oplus_{i=1}^{N} \frac{1}{z^i}\mathbb{C}^{p_2\times q_2}$. We can therefore decompose $\frac{1}{z}\mathcal{H}_2$ into orthogonal subspaces as

$$\frac{1}{z}\mathcal{H}_2 = \mathcal{X} \oplus \frac{1}{z^{N+1}}\mathcal{H}_2,$$

In this paper, we are concerned with *designing* controller constraints describing delay patterns that correspond to a *strongly connected* communication graph. In particular, if we assume that the longest path between nodes in the communicaiton graph is of length N, we may restrict our attention to constraint sets $\mathcal{S} \subset \frac{1}{z}\mathcal{R}_p$ of the form

$$\mathcal{S} = \mathcal{Y} \oplus \frac{1}{z^{N+1}}\mathcal{R}_p \tag{2}$$

with $\mathcal{R}_p$ the space of proper real rational transfer matrices, and $\mathcal{Y} = \oplus_{i=1}^{N} \frac{1}{z^i}\mathcal{Y}_i \subset \oplus_{i=1}^{N} \frac{1}{z^i}\mathbb{R}^{p_2\times q_2} \subset \mathcal{X}$.

In this way $\mathcal{Y}$ encodes the information propagation between nodes, which is governed by the underlying communication graph, and $\frac{1}{z^{N+1}}\mathcal{R}_p$ corresponds to a centralized component in which global, but delayed, information is known. It is precisely this structure that will be exploited in order to reduce this problem to a finite dimensional convex program.

The decentralized control problem of interest is to design a controller $K \in \mathcal{S}$ so as to minimize the closed loop $\mathcal{H}_2$ norm of the system:

$$\begin{aligned} \underset{K}{\text{minimize}} \quad & \|P_{11} + P_{12}K(I - P_{22}K)^{-1}P_{21}\|_{\mathcal{H}_2} \\ & \text{s.t.}\, K \in \mathcal{S} \end{aligned} \tag{3}$$

In [2], it was shown that if the constraint set $\mathcal{S}$ is *quadratically invariant*, then one may pass to the Youla parameterization $Q = K(I - P_{22}K)^{-1}$ in (3) without loss.

Definition 1: A set $\mathcal{S}$ is *quadratically invariant* under P_{22} if

$$KP_{22}K \in \mathcal{S} \text{ for all } K \in \mathcal{S}$$

Since K is assumed to be strictly proper and stabilizing, Q must be strictly proper and stable: thus (3) can be reduced to the following model matching problem:

$$\begin{aligned} \underset{Q}{\text{minimize}} \quad & \|P_{11} + P_{12}QP_{21}\|_{\mathcal{H}_2} \\ & \text{s.t.}\, Q \in \mathcal{S} \textstyle\bigcap \frac{1}{z}\mathcal{H}_2 \end{aligned} \tag{4}$$

For technical simplicity, all controllers in this paper are assumed to be strictly proper – the results extend to non-strictly proper controllers, but the resulting formulas are more complicated. Although this problem admits several solutions [18], [2], [3], [4], we follow the one presented in [5], as it has structure that we exploit in the sequel.

D. Reduction to a Quadratic Program

Let X, Y be the stabilizing solutions to the following Riccati Equations

$$\begin{aligned} X &= C_1^T C_1 + A^T XA - (A^T XB_2 + C_1^T D_{12}) \times \\ & \quad \Omega^{-1}(A^T XB_2 + C_1^T D_{12})^T \\ Y &= B_1^T B_1 + AYA^T - (AYC_2^T + B_1 D_{21}^T) \times \\ & \quad \Psi^{-1}(AYC_2^T + B_1 D_{21}^T)^T \end{aligned}$$

where $\Omega := D_{12}^T D_{12} + B_2^T XB_2$, and $\Psi := D_{21}D_{21}^T + C_2YC_2^T$. Define the regulator and filter gains, respectively, as

$$\begin{aligned} K &= -\Omega^{-1}(B_2^T XA + D_{12}^T C_1) \\ L &= -(AYC_2^T + B_1 D_{21}^T)\Psi^{-1} \end{aligned}$$

and the auxiliary matrix T by

$$T = \Omega^{1/2} \left[\begin{array}{c|c} A & L \\ \hline K & 0 \end{array} \right] \Psi^{1/2}. \tag{5}$$

Finally, let W_L and W_R be left and right spectral factors for $P_{12}^{\sim}P_{12}$ and $P_{21}P_{21}^{\sim}$ such that

$$\begin{aligned} P_{12}^{\sim}P_{12} &= W_L^{-\sim}W_L^{-1} \\ P_{21}P_{21}^{\sim} &= W_R^{-1}W_R^{-\sim}. \end{aligned}$$

We first present the classical solution to the delayed model matching problem, from which the decentralized solution is then constructed.

Theorem 1: The optimal solution to the delayed model matching problem

$$\begin{aligned} \text{minimize}_Q \quad & \|P_{11} + P_{12}QP_{21}\|_{\mathcal{H}_2} \\ & \text{s.t.}\, Q \in \tfrac{1}{z^{N+1}}\mathcal{H}_2 \end{aligned} \tag{6}$$

is given by

$$Q_N = -W_L \mathbb{P}_{\frac{1}{z^{N+1}}\mathcal{H}_2}(T)W_R$$

Theorem 2: [5] The optimal solution to (4) is given by

$$Q^* = U^* + V^*$$

where $V^* \in \mathcal{Y}$ is the unique minimizer of

$$\|G(V)\|_{\mathcal{H}_2}^2 + 2 < G(V), T > \tag{7}$$

with $G(V) = \mathbb{P}_{\mathcal{X}}(W_L^{-1}VW_R^{-1})$, and

$$U^* = Q_N - W_L \mathbb{P}_{\frac{1}{z^{N+1}}\mathcal{H}_2}(W_L^{-1}V^*W_R^{-1})W_R \in \frac{1}{z^{N+1}}\mathcal{H}_2. \tag{8}$$

The optimal cost is then given by

$$\|P_{11} + P_{12}Q_N P_{21}\|_{\mathcal{H}_2}^2 + \|G(V^*)\|_{\mathcal{H}_2}^2 + 2 < G(V^*), T > \tag{9}$$

We now present the quadratic optimization problem that is solved to obtain the FIR component V^* of the decentralized controller. For ease of notation, let $G_i(V) = G_i$, and $H = W_L^{-1}$, $J = W_R^{-1}$. Note that H and J can be expanded as $H = \sum_{i=0}^{\infty} \frac{1}{z^i}H_i$ and $J = \sum_{i=0}^{\infty} \frac{1}{z^i}J_i$. Similarly, T and V admit the expansions $T = \sum_{i=1}^{\infty} \frac{1}{z^i}T_i$, and $V = \sum_{i=1}^{N} \frac{1}{z^i}V_i \in \mathcal{Y}$, with each $V_i \in \mathcal{Y}_i$.[1]

Lemma 1: $(Reduction\ to\ a\ Quadratic\ Program)$

The FIR transfer matrix $G(V)$ can be written as

$$G(V) = \sum_{i=1}^{N} \frac{1}{z^i}G_i, \text{ with } G_i = \sum_{\substack{j,l \geq 0, k \geq 1 \\ j+k+l=i}} H_j V_k J_l \tag{10}$$

and, applying Parseval's identity to (7), we can formulate the optimization problem as

$$\begin{aligned} \text{minimize}_V \quad & \textstyle\sum_{i=1}^{N} \text{Tr}G_iG_i^T + 2\sum_{i=1}^{N} \text{Tr}G_iT_i^T \\ \text{s.t.} \quad & V_i \in \mathcal{Y}_i \end{aligned} \tag{11}$$

Remark 1: It was shown in [5] that (11) is a convex quadratic program with a unique solution.

Thus, we see that in solving the decentralized control problem in this manner, *the entire decentralized nature* of the controller is captured in the FIR filter V – furthermore this filter is solved for via a *finite dimensional convex program.*

As formulated, the objective function of (11) corresponds to the *improvement* over the delayed centralized controller (6) due to the addition of an FIR filter V. It will be more convenient to us to formulate the problem in terms of a deviation from the (classical) centralized optimal controller. To do so, we observe that the optimal FIR filter V^* is unchanged if a constant is added to the objective of optimization (11). Thus, adding $\sum_{i=1}^{N} \text{Tr}T_iT_i^T$ yields the following equivalent formulation

$$\begin{aligned} \text{minimize}_V \quad & \textstyle\sum_{i=1}^{N} \text{Tr}(G_i + T_i)(G_i + T_i)^T \\ \text{s.t.} \quad & V_i \in \mathcal{Y}_i \end{aligned} \tag{12}$$

[1] The component matrices H_i, J_i and T_i can be easily computed via state space methods, c.f. [5]

As shown in the following lemma, this objective function is precisely the deviation from the optimal centralized closed loop norm due to the decentralized constraints $\mathcal{Y}$.

Lemma 2: The optimal cost to (3) is given by $(N_c^2 + \sum_{i=1}^N \mathrm{Tr}(G_i(V^*) + T_i)(G_i(V^*) + T_i)^T)^{\frac{1}{2}}$, where N_c is the closed loop norm of the classical optimal centralized system, and V^* is the solution to (11) and (12).

Proof: From (9), we have that the square of the optimal cost to (3) is given by

$$\begin{aligned} &||P_{11} + P_{12}Q_N P_{21}||_{\mathcal{H}_2}^2 + ||G(V^*)||_{\mathcal{H}_2}^2 + 2 < G(V^*), T > \\ &= ||P_{11} + P_{12}Q_N P_{21}||_{\mathcal{H}_2}^2 + ||G(V^*)||_{\mathcal{H}_2}^2 \\ &\qquad\qquad\qquad + 2 < G(V^*), \mathbb{P}_{\mathcal{X}}(T) > \\ &= ||P_{11} + P_{12}Q_N P_{21}||_{\mathcal{H}_2}^2 - ||\mathbb{P}_{\mathcal{X}}(T)||_{\mathcal{H}_2}^2 \\ &\qquad\qquad\qquad + ||G(V^*) + \mathbb{P}_{\mathcal{X}}(T)||_{\mathcal{H}_2}^2 \end{aligned}$$

where the last equality follows from adding and subtracting $||\mathbb{P}_{\mathcal{X}}(T)||_{\mathcal{H}_2}^2 = \sum_{i=1}^N \mathrm{Tr} T_i T_i^T$. We note that applying Parseval's identity to $\|G(V^*) + \mathbb{P}_{\mathcal{X}}(T)\|_{\mathcal{H}_2}^2$ yields the objective function of (12), and so it suffices to prove that $||P_{11} + P_{12}Q_N P_{21}||_{\mathcal{H}_2}^2 - ||\mathbb{P}_{\mathcal{X}}(T)||_{\mathcal{H}_2}^2 = N_c^2$.

This identity can be derived via spectral factor methods similar to those used in [5], but we choose to present here an optimization based proof of this fact. We first observe that the optimal solution V_C^* to (11) with the affine constraints removed coresponds to the first N elements of the impulse response of the optimal centralized controller. These can be solved for analytically as $V_C^* = -G^{-1}(\mathbb{P}_{\mathcal{X}}(T))$, where G^{-1} is the inverse of the linear operator $G(V)$ defined in (10) (note that under our assumptions, G is invertible, c.f. Section III.A.3 [8]). This yields an optimal value of $-||\mathbb{P}_{\mathcal{X}}(T)||_{\mathcal{H}_2}^2 = -\sum_{i=1}^N \mathrm{Tr} T_i T_i^T$, which, when combined with (9), gives the desired result. ∎

E. Atomic norms and structured solutions

As mentioned in the introduction, it is often known *a priori* that the solution to an optimization problem should be "simple," and that this simple structure can be promoted through the use of an appropriate *convex* function. This notion of solutions with simple structure, in the context of linear inverse problems, has been formalized and generalized in terms of *atomic norms* [15].

In particular, if it is known that the true solution X_* to a set of linear equations

$$y = Ax + \nu, \tag{13}$$

for some bounded noise term $||\nu||_2 \leq \delta$, should consist of a linear combination of a small number of "atoms", then it is shown that one should seek the solution that minimizes a so-called atomic norm, subject to consistency constraints. Specifically, if one assumes that

$$X_* = \sum_{i=1}^{r} c_i a_i, \ a_i \in \mathcal{A}, \ c_i \geq 0$$

for $\mathcal{A}$ a set of appropriately scaled and centered "atoms," and r a small number relative to the ambient dimension, then solving

$$\mathrm{minimize}_X \|X\|_{\mathcal{A}} \tag{14}$$

$$s.t. \|y - AX\|_2^2 \leq \delta^2 \tag{15}$$

with the atomic norm $\|\cdot\|_{\mathcal{A}}$ given by the gauge function

$$\begin{aligned} ||X||_{\mathcal{A}} : &= \inf\{t \mid X \in t\mathrm{conv}(\mathcal{A})\} \\ &= \inf\{\textstyle\sum_{a\in\mathcal{A}} |c_a| \mid X = \sum_{a\in\mathcal{A}} c_a a\} \end{aligned} \tag{16}$$

results in solutions that both satisfy the consistency constraint (15), and are sparse at the atomic level (i.e. are a linear combination of a small number of elements $a \in \mathcal{A}$).

The geometric justification behind the success of these methods is that the unit-ball of these norms is "pointy" in high dimensions, and thus solutions are likely to be at singularities (i.e. edges or corners) of the norm-ball, inducing the desired simple structure.

In contrast to [16] and results typical of the machine learning and statistics community, which attempt to identify an underlying *true model*, we, much as in [9], use atomic norm minimization as a tool for *design*.

III. Communication Delay Co-Design Via Atomic Norm Minimization

We begin by formally defining the information propagation pattern, or path, induced by a graph. We then impose additional constraints on the set of paths we allow such that they are compatible with both QI conditions and the physical restrictions of the design problem. Given such a collection of paths, which we term *implementable*, we show that they can be used to construct an atomic norm that induces simple communication patterns. With these tools at our disposal, we formulate the communication delay pattern co-design problem as a convex program that allows the designer an explicit trade off between closed loop performance and communication graph complexity.

For ease of presentation, we assume that the communication delay between neighboring nodes is 1, and that the nodes are single-input single-output (SISO) plants – the results and definitions extend to the general case in a straightforward, if not notationally cumbersome, way. We provide some brief comments on how this can be accomplished at the end of the section.

A. Information propagation patterns

Key to our approach is generating sparsity patterns that are consistent with how information propagates across a graph. In our setting, we assume that nodes can share the information that they have access to with their neighbors at each time step – as such, one can view the information propagation through the graph as a spreading of non-zero terms with time.

Definition 2: Let $s \in \mathbb{R}^{p\times p}$ define a sparsity pattern corresponding to the adjacency matrix of a graph, where we assume self loops such that $s_{ii} = 1$ for all i. We define the *information propagation pattern*, or path, induced by s over D time steps, $P_s(D)$, as

$$P_s(D) \quad := \quad \oplus_{i=1}^{D} \tfrac{1}{z^i} \mathrm{supp}(s^i) \tag{17}$$

Similarly, we define the set of information propagation patterns, or set of paths, induced by a set of sparsity patterns Σ over D time steps to be

$$\mathcal{P}_\Sigma(D) := \{P_s(D) \mid s \in \Sigma\}. \tag{18}$$

With these defintions, we see that indeed, the information available at node i spreading to its t^{th} neighbors by time $t \leq D$ is reflected in $P_s(D)$ by the spreading of non-zero terms as s is raised to higher and higher powers.

In a design setting, the set of sparsity patterns Σ will be chosen such that every $s \in \Sigma$ defines an adjacency matrix corresponding to a communication graph that can be implemented under the physical constraints of the system. For example, direct links between physically distant plants may be disallowed to ensure that the system can actually be built as designed.

Our approach will be to construct simple communication graphs by taking linear combinations of a small number of elements $s \in \Sigma$. As such, we need to ensure that a graph resulting from an arbitrary sum of sparsity patterns $s \in \Sigma$ (i) also respects the physical constraints of the system, and (ii) results in a controller K that can be implemented. Note that (i) holds automatically if each $s \in \Sigma$ is chosen to respect the physical constraints of the system. A sufficient condition to ensure that (ii) holds is that the so-called *maximal graph* of Σ is QI under P_{22}.

Definition 3: The *maximal graph* σ of a sparsity set Σ is given by

$$\sigma := \operatorname{supp}\left(\sum_{s \in \Sigma} s\right)$$

We say that both the sparsity set Σ, and the set of paths $\mathcal{P}_\Sigma(D)$ it induces, are *implementable under* P_{22} if $P_\sigma(D) \oplus \frac{1}{z^{D+1}}\mathcal{H}_2$ is QI under P_{22}.

Under these definitions, an arbitrary sum of sparsity patterns $P \in \mathcal{P}_\Sigma(D)$ has its support contained in $P_\sigma(D)$. If Σ is implementable under P_{22}, the QI condition on $P_\sigma(D)$ implies that any constraint set generated by such an overlay of sparsity patterns will result in a controller $K \in P_\sigma(D) \oplus \frac{1}{z^{D+1}}\mathcal{H}_2$, i.e. one that is implementable under the physical constraints of the problem.

A stronger requirement (that in practice is easily met, as will be illustrated in Section IV) is that every path be individually implementable.

Definition 4: If for every $s \in \Sigma$, we have that $P_s(D) \oplus \frac{1}{z^{D+1}}\mathcal{H}_2$ is QI under P_{22}, then $\mathcal{P}_\Sigma(D)$ is *strongly implementable*.

B. The path-based atomic norm

Our objective in this subsection is to construct an atomic norm that will induce sparsity patterns in the FIR filter V that are consistent with implementable information propagation patterns.

At a conceptual level, it follows that the atomic elements from which we wish to construct our solution, and that will induce an appropriate atomic norm (16), are precisely the information propagation patterns induced by a sparsity set Σ that is (strongly) implementable under P_{22}. Additionally, as these paths may overlap, we must take care to address this issue in designing our atomic norm so as to ensure that elements that lay in the intersection of paths are not penalized more than once.

In order to do work with FIR filters in the context of finite dimensional convex optimization, we overload notation and define the *vectorization* $\operatorname{vec}(V)$ of an FIR filter V with impulse response matrices V_i, $i = 1, ..., D$, to be

$$\operatorname{vec}(V) = \begin{bmatrix} V_1 & \dots & V_D \end{bmatrix} \tag{19}$$

With this in mind, let us fix a sparsity pattern $s \in \mathbb{R}^{p\times p}$. This sparsity pattern induces a path $P_s(D)$, which in turn induces a family of atoms $\mathcal{A}_{P_s(D)}$ in the following manner:

$$\begin{aligned}\mathcal{A}_{P_s(D)} := \{&\operatorname{vec}(V) \in \left(\mathbb{R}^{Dp^2}\right) \mid V = [V_1, \dots, V_D],\\ &||\operatorname{vec}(V)||_2 = 1,\ \operatorname{supp}(V_i) \subset \operatorname{supp}(s^i),\ \forall i = 1, ..., D\}\end{aligned} \tag{20}$$

i.e. the set of all FIR filters with unit Euclidean norm (after vectorization) with support contained within $P_s(D)$.

In particular, each atom $a_s \in \mathcal{A}_s$ is within an appropriate $\sum_{i=1}^D |\operatorname{supp}(s^i)|$-dimensional Euclidean norm unit-ball, embedded in $\mathbb{R}^{Dp^2}$. For ease of notation, we will write the condition $\operatorname{supp}(V_i) \subset \operatorname{supp}(s^i)$, $\forall i = 1, ..., D$, as $\operatorname{supp}(V) \subset \operatorname{supp}(P_s(D))$.

Consider now a(n) (strongly) implementable set Σ of sparsity patterns. We can then define the atomic set $\mathcal{A}$ as

$$\mathcal{A} := \bigcup_{s \in \Sigma} \mathcal{A}_{P_s(D)}$$

and it can be shown (by specializing the arguments in [19]) that, for an FIR filter V, $||\operatorname{vec}(V)||_{\mathcal{A}}$ is given by the value of the following convex program

$$\begin{aligned}&\min_{\{V_s\}} \textstyle\sum_{s \in \Sigma} ||\operatorname{vec}(V_s)||_2 \ s.t.\\ &\quad \operatorname{vec}(V) = \textstyle\sum_{s \in \Sigma} \operatorname{vec}(V_s)\\ &\quad \operatorname{supp}(V_s) \subset \operatorname{supp}(P_s(D))\ \forall s \in \Sigma.\end{aligned} \tag{21}$$

This program finds a minimum $\|\cdot\|_{\mathcal{A}}$ decomposition of V in terms of the information propagation patterns induced by Σ.

Geometrically, this atomic norm ball consists of the convex hull of the lower-dimensional Euclidean balls induced by each $P_s(D) \in \mathcal{P}_\Sigma(D)$ embedded in $\mathbb{R}^{Dpq}$. Much as the nuclear norm can be viewed as the ℓ_1-norm of the singular values, inducing sparsity in that respect, this norm can be seen as an ℓ_1-norm on paths, inducing sparsity at the path level. It is also worth noting that this penalty function is known as the group LASSO with overlap in the machine learning and statistics community [19].

Example 1: Consider the simple set of allowable atoms $\mathcal{A} = \{\begin{bmatrix}1 & 1 & 0\end{bmatrix}, \begin{bmatrix}0 & 1 & 1\end{bmatrix}\}$. The resulting atomic norm ball is illustrated in Figure 1.

Example 2: Consider the sparsity pattern

$$s = \begin{bmatrix} 1 & 1 & 0 & 0\\ 1 & 1 & 1 & 0\\ 0 & 1 & 1 & 1\\ 0 & 0 & 1 & 1 \end{bmatrix}$$

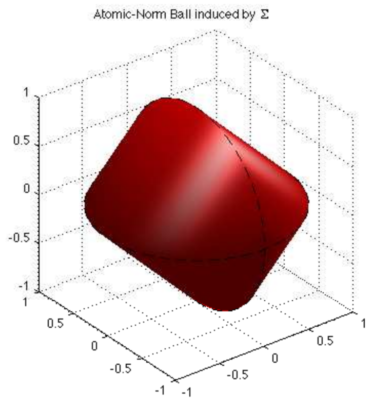


Fig. 1: The atomic norm ball induced by the atomic set $\mathcal{A}$ in Example 1. Notice that the singularities (dashed lines) of the boundary are along the unit disks contained in the support of each element of $\mathcal{A}$.

corresponding to the adjacency matrix of a 4-player chain. The information propagation path induced is then

$$P_s(2) = \frac{1}{z}\begin{bmatrix}1 & 1 & 0 & 0\\ 1 & 1 & 1 & 0\\ 0 & 1 & 1 & 1\\ 0 & 0 & 1 & 1\end{bmatrix} \oplus \frac{1}{z^2}\begin{bmatrix}1 & 1 & 1 & 0\\ 1 & 1 & 1 & 1\\ 1 & 1 & 1 & 1\\ 0 & 1 & 1 & 1\end{bmatrix}$$

and the corresopnding family of atoms is given by

$$\begin{aligned}\mathcal{A}_{P_s(2)} = \{&\mathrm{vec}(V) \in \mathbb{R}^{32} \mid V = [V_1, V_2],\\ &\|\mathrm{vec}(V)\|_2 = 1,\ \mathrm{supp}(V) \subset \mathrm{supp}(P_s(D))\}\end{aligned}$$

C. Problem formulation and solution via convex programming

We now have the definitions necessary in place to formally state our problem.

Problem 1: Given a plant (1) and a(n) (strongly) implementable sparsity set Σ, design a communication delay pattern that is (i) (strongly) implementable, (ii) simple and (iii) yields acceptable closed loop performance.

As a fair amount of definitions and notation have been introduced, before presenting our main result, we briefly summarize (i) the assumptions required for our approach to work, and (ii) the general approach taken to derive our method.

First, we recall that our results build on the solution to the decentralized $\mathcal{H}_2$ optimal control problem presented in [5]. This solution, and consequently our method, assumes both an open-loop stable plant, and a strongly connected communication graph. This latter condition is necessary for the decomposition of the optimal controller into a centralized, but delayed, component, and a local FIR component V that is solved for through a finite dimensional convex program (12).

With this convex program at our disposal, we borrow ideas from machine learning and statistics, specifically those of atomic norm minimization, and modify it for our purposes. In particular, we remove the affine constraints that impose the decentralized nature of the FIR component, and rather minimize an appropriately chosen atomic norm subject to performance constraints, so as to *induce* a simple, but implementable, communication structure between nodes.

Theorem 3: Suppose that Σ is a sparsity pattern that induces a set of paths $\mathcal{P}_\Sigma(D)$ that is (strongly) implementable under P_{22}, and $\delta > 0$ is a tuning parameter. Then the optimal FIR filter V^* that solves the finite dimensional second order cone program

$$\begin{array}{ll}\min_{\{V_s\}} \sum_{s\in\Sigma} ||\mathrm{vec}(V_s)||_2 s.t. & \\ \text{(c1)} \quad \sum_{i=1}^{D} \mathrm{Tr}(G_i(V)+T_i)(G_i(V)+T_i)^T \le \delta^2 & \\ \text{(c2)} \quad \mathrm{vec}(V) = \sum_{s\in\Sigma} \mathrm{vec}(V_s) & \\ \text{(c3)} \quad \mathrm{supp}(V_s) \subset \mathrm{supp}(P_s(D))\ \forall s \in \Sigma & \end{array} \tag{22}$$

leads to a controller that (i) is (strongly) implementable, and (ii) yields a closed loop system with norm N_{des} satisfying

$$N_{\mathrm{des}}^2 \le N_c^2 + \delta^2,$$

where N_c is the closed loop norm of the optimal centralized $\mathcal{H}_2$ controller, and $G_i(V)$ and T_i are as defined in Section II.

Before presenting the proof, we note that the objective function, along with constraints (c2) and (c3) are simply the atomic norm (21), while (c1) is precisely the objective function of (12). We therefore have that the tuning parameter δ bounds the deviation of the system with respect to the optimal centralized solution. As such, through the choice of δ, this program gives the designer explicit control over the trade off between the complexity of the communication graph and the control performance.

Proof: Convexity of (22) follows from the fact that the $\|\cdot\|_2$ is a convex function, that convexity is preserved under non-negative linear combinations, that (c1) is a second order cone constraint, and that (c2) and (c3) are affine constraints [20]. We therefore have that (22) is a finite dimensional SOCP and is solvable in polynomial time.

The (strong) implementability of the resulting controller follows immediately by the hypothesis that $\mathcal{P}_\Sigma(D)$ is (strongly) implementable. The condition on the closed loop norm follows directly from Lemma 2:

$$\begin{aligned}N_{\mathrm{des}}^2 &= N_c^2 + \textstyle\sum_{i=1}^{N} \mathrm{Tr}(G_i(V^*)+T_i)(G_i(V^*)+T_i)^T\\ &\le N_c^2 + \delta^2\end{aligned}$$

■

Remark 2: Exploiting the identity $\mathrm{vec}(AXB) = (B^T \otimes A)\mathrm{vec}(X)$ for matrices A, X and B of compatible dimension, (22) can be recast as

$$\begin{array}{c}\min_{\{V\}} ||\mathrm{vec}(V)||_{\mathcal{A}}\ s.t.\\ \|\mathrm{vec}(T) - A\mathrm{vec}(V)\|_2^2 \le \delta^2\end{array} \tag{23}$$

for an appropriately chosen matrix A such that $A \cdot \mathrm{vec}(V) = \mathrm{vec}(\sum_{i=1}^{D} G_i(V))$ (an explicit expression for A can be derived in terms of J_i and H_i, but is not informative nor necessary to our discussion). This atomic norm minimization problem is of exactly the same form as (14), and we therefore expect the solution of this program to be simple at the atomic level [15], resulting in simple communication graphs.

Remark 3: Once a sparsity pattern is identified for the FIR filter V, a refinement step (much as the one suggested in [9]) can be performed by setting $\mathcal{Y} = \sum_{s\in\Sigma^*} P_s(D)$ in (11), where $\Sigma^* \subset \Sigma$ is the set of active sparsity patterns identified by (22).

1) Extensions to larger delays and MIMO systems: We begin by considering the case where the communication delay between nodes is now given by $\tau > 1$. In this case, the information propagates at a slower rate, and this must be reflected in the resulting information propagation patterns. In particular, suppose that D is the length of the longest path between nodes in a graph. We then have that

$$P_s(\tau, D) \quad := \quad \oplus_{i=1}^{D\tau} \frac{1}{z^i} \mathrm{supp}(s^{\lceil \frac{i}{\tau} \rceil}), \tag{24}$$

i.e. we have that the sparsity pattern remains constant over τ time steps before spreading. A similar idea can be applied to the case of differing communication delays between nodes, but we do not have a concise notational way of expressing the resulting information propagation patterns.

In order to extend our results to multi-input multi-output (MIMO) systems, it suffices to replace each non-zero term in $P_s(D)$ with an appropriately sized all-ones matrix $\mathbb{1}$ such that the resulting atomic sets leads to constraints that are compatible with the MIMO nature of the plant. For example if plant i has 2 inputs, and plant j has 3 outputs, non-zero terms in the information propagation pattern corresponding to K_{ij}, that is to say the control actions taken at plant i based on the outputs of plant j, would be replaced by $\mathbb{1}_{2\times 3}$.

IV. Generating sparsity patterns

What remains to be specified is a method for generating sparsity patterns s out of which to build communication graphs. At first, this task seems to be one that may not admit a general method that is computationally tractable. Due to the inherent flexibility of our approach, it is difficult to make any general claims in this respect – what constitutes an implementable communication link is typically a very application specific property. In the following, we provide tractable algorithms that seem to yield good performance in the resulting graph/controller combination. Although these are preliminary methods, they have already shown promise in practice. The tractable design of atomic sets with *provable performance guarantees* will be the topic of future research.

We do however point out the following features of the problem which give hope that such tractable solutions may in fact exist for interesting problems: (i) The number of possible paths is significantly reduced if we assume symmetry in the communication graph (i.e $s_{ij} = s_{ji}$). Such an assumption is often easily satisfied in practice. (ii) If strong implementability is desired, QI imposes constraints on path lengths between nodes, greatly reducing the number of paths that need to be enumerated. (iii) Links that are not physically realizable also remove degrees of freedom from the problem, once again greatly reducing the size of the set of implementable paths. (iv) Since overlap is allowable between paths, algorithms do not have to keep track of what paths have already been generated, simplifying implementation. (v) In many applications, such as robotics and neurophysiology, the structure of the component paths are known a priori – it is how fast these paths need to be, and how many of them need to be present, to achieve a specific level of performance that needs to be determined. In this case the constraint set is entirely predefined by domain specific knowledge.

In the following, we present first a general algorithm that introduces what we conjecture to be a mild amount of conservatism to the design process, and then conclude with a specialized algorithm for plants with a chain structure – both of these algorithms result in path sets Σ with cardinalities that are polynomial in both the number of nodes and the number of allowable links at each node.

We assume that communication links are symmetric, fix D such that all of the paths are implementable, and suppress it when specifying paths to ease notational burden. Both algorithms center around a *base QI pattern* $s_{QI} \in \Sigma$ and a set of admissible enhancement links $\mathcal{E}$.

Definition 5: Given a sparsity set Σ, a sparsity pattern $s_{QI} \in \Sigma$ can be a base QI pattern only if it induces a strongly implementable path $P_{s_{QI}}$, and $s_{QI} \subset s$ for all $s \in \Sigma$.

Definition 6: The set $\mathcal{E}$ of admissible enhancement links encodes the additional links beyond the base QI pattern that can be physically implemented:

$$\begin{aligned} \mathcal{E} : \quad = \quad & \{E_{ij} + E_{ji} \mid \text{a direct link between nodes } i \text{ and } j \\ & \text{is physically implementable}\} \bigcup \{0\}. \end{aligned} \tag{25}$$

We can then express any $s \in \Sigma$, for any implementable sparsity pattern set Σ, as $s = s_{QI} + \sum_{e\in\mathcal{F}} e$ for some $s_{QI} \in \Sigma$ and some $\mathcal{F} \subset 2^{\mathcal{E}}$.

A. A general algorithm: the single link approach

It is a well known principle in optimal control that delay is detrimental to performance. In fact, it is a common feature that performance is predominantly determined by the first few elements of the impulse response of a controller, as higher order elements decrease in magnitude exponentially. With this in mind, we propose building a constraint set around a predefined base QI pattern s_{QI} as:

$$\Sigma = \{s \mid s = s_{QI} + e,\ e \in \mathcal{E}\}.$$

This allows for any sparsity pattern corresponding to the adjacency matrix of a physically implementable communication graph to be constructed from a sum of the elements in Σ, with $|\Sigma| = |\mathcal{E}| + 1$. However, the simplicity of this sparsity pattern set comes at a cost: there is a degree of conservatism that is introduced. For general $r, s \in \Sigma,\ t > 0$, $\mathrm{supp}(\mathrm{r^t} + \mathrm{s^t}) \subset \mathrm{supp}\left((r+s)^t\right)$ – i.e. summing the induced information propagation pattern of two sparsity patterns typically yields a subset of the information propagation pattern induced by the sum of the sparsity patterns.

However, this conservatism is only introduced in later components of V, and, by our earlier observation that performance is dominated by the first few elements of the impulse response of the controller, does not seem to introduce much additional conservatism in the graph design procedure. If the resulting performance of the algorithm is not deemed acceptable, the sparsity pattern set Σ can be augmented to additionally include all "2-link enhancements" (i.e. $\Sigma = \{s \mid s = s_{QI} + e + f,\ e, f \in \mathcal{E}\}$) – this in a sense, pushes the conservatism back even further back in time in V, but comes at a cost of $\binom{\lfloor \frac{|\mathcal{E}|}{2} \rfloor}{2}$ more elements. This process can be repeated for additional "n-link enhancements", as needed and allowable by computational constraints.

Additionally, once the set of active paths in $\Sigma^* \subset \Sigma$ is identified, the refinement step mentioned in Remark 3 can be performed, removing the conservatism in the controller design. We illustrate this approach on a 13 node system with a honeycomb physical interconnection in Section V.

B. The 2-hop N-player chain problem

Consider the N-player chain problem, illustrated for $N = 8$ players in Figure 2, where each plant is strictly proper. Then

$$P_{22}^{ij} \in \frac{1}{z^{1+|j-i|}} \mathcal{H}_2 \ \forall i, j = 1, \ldots, N.$$

We further assume that due to physical constraints, communication links can only be made between immediate and second neighbors, i.e. s_{ij} may be non-zero only if $|j-i| \leq 2$. It is trivially verified that every s of this form with $s_{ii} = 1 \ \forall i$ induces a strongly implementable path $P_s(N-2)$ (the longest path between nodes is of length $N-1$). A naive enumeration of all sparsity patterns of this type would lead to a sparsity set Σ with cardinality given by

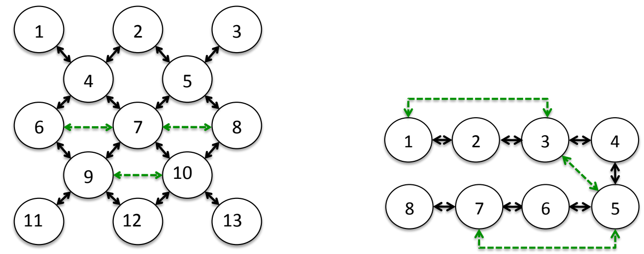
$$|\Sigma| = \sum_{i=1}^{\lfloor \frac{N}{2} \rfloor + 1} \binom{\lfloor \frac{N}{2} \rfloor + 1}{i},$$

a quantity that quickly becomes unmanageable even for moderate N. (e.g. for $N = 100$, $|\Sigma| \sim 10^{15}$). We propose the following algorithm for building a constraint set Σ with cardinality of order $O(N^2)$:

Algorithm 1 The N-player 2-hop chain problem

1) Let $\Sigma_1 = \emptyset$.
2) For each $k \in \{1, \ldots, \lceil \frac{N}{2} \rceil - 1\}$, let $l \in \{1, ..., \lfloor \frac{N-k}{2} \rfloor\}$, and $s^{kl} = T_N$, where T_N is a tridiagonal matrix consisting of all ones on the main, first super and first sub-diagonal.
3) For $i = k : 2 : k + 2(l-1)$, set $s^{kl}_{i,i+2} = s^{kl}_{i+2,i} = 1$.
4) Add s^{kl} to Σ_1.
5) Set $\Sigma = \Sigma_1 \bigcup \mathrm{mirror}(\Sigma_1)$.

The order of $|\Sigma|$ follows immediately from the ranges on k and l. The intuition behind this algorithm is that starting at node 1, we define atom s^{11} as the base chain augmented by a link between 1 and 3; s^{12} is then given by the base chain augmented with links between 1 and 3, and between 3 and 5; and so on. The process is then repeated starting at node 2, then at node 3, etc., until the $\left(\lceil \frac{N}{2} \rceil - 1\right)$'th node. The mirror operator in step 5) above simply exploits the symmetry of the chain to simplify the algorithm, and replicates the process starting from the other end of the chain. We demonstrate this



(a) 13-player honeycomb physical/comm. network

(b) 8-player chain example physical/comm. network

Fig. 2: Physical and communication networks for the examples considered in Section II. Solid black lines denote physical links and communication links included in the base path s_{QI}. Dashed green lines are additional communication links selected by our algorithm.

algorithm in the Section II on an 8 plant system, for which $|\Sigma| = 16$.

V. Examples

We now illustrate our method on two systems (1) with substantially different physical topologies. In all of the following examples, we let N denote the number of players in the problem, and set $B_2 = C_2 = I_N$, $B_1 = [10I_N, 0_{N\times N}]$, $C_1 = [10I_N, 0_{N\times N}]^T$, $D_{11} = 0_{2N\times 2N}$, $D_{12} = [0_{N\times N}, 2I_N]^T$, $D_{21} = [0_{N\times N}, 5I_N]$ and $D_{22} = 0_{2N\times N}$. In each case, we generate A randomly such that it agrees with the topology of the physical interconnection of the plant, and normalize it such that $|\lambda_{max}(A)| = .999$, ensuring stability of the open-loop system. We choose a base path s_{QI} around which to build our atoms that mimics the physical interconnection of each plant, an approach that ensures strong implementability of the resulting sparsity set. Additionally, this base path and the information propagation pattern that it induces is the conventional communication structure assumed in most decentralized control problems.

The key point to notice in both examples is that through the addition of a small number of unconventional links, the performance of the easily implemented decentralized controller is *nearly identical* to that of the physcally unrealizable centralized controller.

1) The 13-player honeycomb problem: The physical interconnection is illustrated in Figure 2(a). For the A generated for this example, the open loop norm of the system is 5915, the centralized closed loop norm N_c is 588, and the delayed centralized closed loop norm N_d is 1105. We solved program (22), with Σ generated according to the general algorithm described in Section IV, and $\delta = .0005(N_d^2 - N_c^2) = 437$, to design the communication graph shown in Figure 2(a). We then performed the refinement step to design a controller that yielded a closed loop norm of $N_{\text{des}} = 648$, which performs slightly better than a conroller designed on the base chain topology, which yields a closed loop norm of 677.

2) The 8-player 2-hop chain problem: The physical interconnection is illustrated in Figure 2(b). For the A generated for this example, the open loop norm of the system is 5850, the centralized closed loop norm N_c is 420, and the delayed centralized closed loop norm N_d is 802. We solved program (22), with Σ generated according to the specialized algorithm described in Section IV, and $\delta = .0004(N_d^2 - N_c^2) = 187$, to design the communication graph shown in Figure 2(b). We then performed the refinement step to design a controller that yielded a closed loop norm of $N_{\text{des}} = 421$, nearly identical to the centralized performance, and *much better* than the base chain closed loop norm of 542.

VI. Conclusion

This paper presented an atomic norm minimization based approach to communication graph co-design in the distributed $\mathcal{H}_2$ control setting. It was shown that this problem could be formulated as a finite dimensional convex program that provides the designer a tuning parameter that allows for an explicit trade off between closed loop performance and communication graph complexity.

There are many promising directions for future work. In order to provide any kind of rigorous guarantees about the quality of the path generating algorithms that we presented, we will need to determine a proper criteria against which to measure the success of a co-design problem. With this in mind, an immediate goal is to extend the non-asymptotic consistency results typical of the machine learning literature to our setting. Once this has been accomplished, we believe that this will provide insight for more principled path generating algorithms.

References

[1] A. Mahajan, N. Martins, M. Rotkowitz, and S. Yuksel, "Information structures in optimal decentralized control," in *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, 2012, pp. 1291–1306.

[2] M. Rotkowitz and S. Lall, "A characterization of convex problems in decentralized control," *Automatic Control, IEEE Transactions on*, vol. 50, no. 12, pp. 1984 –1996, dec. 2005.

[3] A. Rantzer, "A separation principle for distributed control," in *Decision and Control, 2006 45th IEEE Conference on*, dec. 2006, pp. 3609 – 3613.

[4] A. Gattami, "Generalized linear quadratic control theory," in *Decision and Control, 2006 45th IEEE Conference on*, dec. 2006, pp. 1510 –1514.

[5] A. Lamperski and J. C. Doyle, "Output feedback $\mathcal{H}_2$ model matching for decentralized systems with delays," in *2013 IEEE American Control Conference, To appear.*, sept. 2012.

[6] J. Hespanha, P. Naghshtabrizi, and Y. Xu, "A survey of recent results in networked control systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 138–162, 2007.

[7] N. Matni and J. C. Doyle, "Optimal distributed lqg state feedback with varying communication delays," in *Submitted to the IEEE Conference on Decision and Control, 2013*, 2013.

[8] ——, "A dual problem in h2 decentralized control subject to delays," in *The IEEE American Control Conference (ACC), 2013*, 2013.

[9] F. Lin, M. Fardad, and M. Jovanovic, "Sparse feedback synthesis via the alternating direction method of multipliers," in *American Control Conference (ACC), 2012*, 2012, pp. 4765–4770.

[10] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *Information Theory, IEEE Transactions on*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[11] E. Candès and B. Recht, "Exact matrix completion via convex optimization," *Commun. ACM*, vol. 55, no. 6, pp. 111–119, June 2012. [Online]. Available: http://doi.acm.org/10.1145/2184319.2184343

[12] D. L. Donoho, "For most large underdetermined systems of linear equations the minimal ℓ_1-norm solution is also the sparsest solution," *Comm. Pure Appl. Math*, vol. 59, pp. 797–829, 2004.

[13] ——, "Compressed sensing," *IEEE Trans. Inform. Theory*, vol. 52, pp. 1289–1306, 2006.

[14] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM Rev.*, vol. 52, no. 3, pp. 471–501, Aug. 2010. [Online]. Available: http://dx.doi.org/10.1137/070697835

[15] V. Chandrasekaran, B. Recht, P. Parrilo, and A. Willsky, "The convex geometry of linear inverse problems," *Foundations of Computational Mathematics*, vol. 12, pp. 805–849, 2012. [Online]. Available: http://dx.doi.org/10.1007/s10208-012-9135-7

[16] P. Shah, B. N. Bhaskar, G. Tang, and B. Recht, "Linear system identification via atomic norm regularization," in *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, 2012, pp. 6265–6270.

[17] S. Sabau and N. C. Martins, "Stabilizability and norm-optimal control design subject to sparsity constraints," *arXiv preprint arXiv:1209.1123*, 2012.

[18] A. Nayyar, A. Mahajan, and D. Teneketzis, "Optimal control strategies in delayed sharing information structures," *Automatic Control, IEEE Transactions on*, vol. 56, no. 7, pp. 1606 –1620, july 2011.

[19] L. Jacob, G. Obozinski, and J.-P. Vert, "Group lasso with overlap and graph lasso," in *Proceedings of the 26th Annual International Conference on Machine Learning*, ser. ICML '09. New York, NY, USA: ACM, 2009, pp. 433–440. [Online]. Available: http://doi.acm.org/10.1145/1553374.1553431

[20] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.