On Structured Realizability and Stabilizability of Linear Systems

Laurent Lessard[1,2]   Maxim Kristalny[1,2]   Anders Rantzer[1]

Submitted, American Control Conference 2013

Abstract

We study the notion of *structured realizability* for linear systems defined over graphs. A stabilizable and detectable realization is *structured* if the state-space matrices inherit the sparsity pattern of the adjacency matrix of the associated graph. In this paper, we demonstrate that not every structured transfer matrix has a structured realization and we reveal the practical meaning of this fact. We also uncover a close connection between the structured realizability of a plant and whether the plant can be stabilized by a structured controller. In particular, we show that a structured stabilizing controller can only exist when the plant admits a structured realization. Finally, we give a parameterization of all structured stabilizing controllers and show that they always have structured realizations.

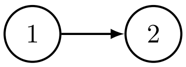
1 Introduction

Linear time-invariant systems are typically represented using transfer functions or state-space realizations. The relationship between these representations is well understood; every transfer function has a minimal state-space realization and one can easily move between representations depending on the need.

In this paper, we address the question of whether this relationship between representations still holds for systems defined over directed graphs. Consider the simple two-node graph of Figure 1. For $i = 1, 2$, the node i represents a subsystem with inputs u_i and outputs y_i, and the edge means that subsystem 1 can influence subsystem 2 but not vice versa. The transfer matrix for any such a system has the sparsity pattern $\mathcal{S}_1$.

For a state-space realization to make sense in this context, every state should be associated with a subsystem, which means that the state should be computable using the information available to that subsystem. In the case of $\mathcal{S}_1$, the states that determine y_1 must only depend on u_1, while the states that determine y_2 may depend on both u_1 and u_2. Consider the following example, which belongs to the graph constraint of Figure 1.

$$G_1 = \begin{bmatrix} \frac{1}{s+1} & 0 \\ \frac{1}{s+1} & \frac{1}{s+2} \end{bmatrix} = \left[\begin{array}{cc|cc} -1 & 0 & 1 & 0 \\ 0 & -2 & 0 & 1 \\ \hline 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{array}\right] \tag{1}$$



$$\mathcal{S}_1 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

Figure 1: A two-node graph and its adjacency matrix.

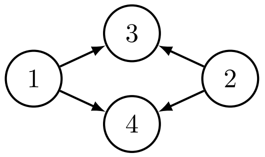
More generally, we say a system is $\mathcal{S}$*-realizable* if it has a stabilizable and detectable realization for which A, B, C, D each have the sparsity of $\mathcal{S}$. A more formal definition is given in Section 3.3.

While this definition seems natural and one might expect every structured transfer function to have a corresponding $\mathcal{S}$-realization, it is not the case in general. For example, G_2 defined in (2) belongs to the graph constraint of Figure 2, but no $\mathcal{S}_2$-realization exists. A proof is given in Appendix A. This example shows that there is no immediate relation between the sparsity of a transfer matrix and that of its state-space matrices.

Our main result is that $\mathcal{S}$-realizability is necessary for $\mathcal{S}$-stabilizability; finding a stabilizing controller that has the same $\mathcal{S}$-structure as the plant. Non-realizable systems exist, but such systems cannot stabilize or be stabilized by other structured systems.

The paper is organized as follows. In Section 2, we cover some related work in the literature and in Section 3 we cover basic definitions and concepts touching on systems over graphs, realizability, and stabilizability. Our main results are in Sections 4 and 5, which are followed by concluding remarks in Section 6.

$$G_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{1}{s-1} & \frac{1}{s-1} & 0 & 0 \\ \frac{1}{s-1} & \frac{1}{s-1} & 0 & 0 \end{bmatrix} \tag{2}$$



$$\mathcal{S}_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}$$

Figure 2: A four-node graph and its adjacency matrix.

[1] L. Lessard, M. Kristalny, and A. Rantzer are with the Department of Automatic Control at Lund University, Lund, Sweden. {lessard,maxim,rantzer}@control.lth.se

[2] This work was supported by the Swedish Research Council through the LCCC Linnaeus Center

2 Literature Review

There is a large body of work exploring control systems defined over graphs. For a broad class of systems, synthesizing an optimal controller can be reduced to solving a convex optimization problem [5, 7, 9].

Several solution approaches have been reported. LMI methods [1, 8, 12, 20] work directly with state-space realizations for the plant and controller. Vectorization [10] avoids the sparsity constraint by reshaping the transfer matrix. Alternatively, one can solve a sequence of finite-dimensional convex problems whose solutions converge to the optimal structured controllers [7].

Surprisingly, the issue of structured realizability is not addressed in any of the aforementioned works. Indeed, the LMI methods assume that structured realizations for the plant and controller always exist, while transfer function methods make no mention of state-space — so it's conceivable that a transfer function method might generate a controller which has no structured realization!

In this paper, we consider directed graphs with delay-free links. Our framework is similar to that of Swigart and Lall [15]. For many such problems, explicit state-space solutions have been directly computed [3, 4, 6, 13, 14]. In these works, a structured realization is assumed for the plant, and the optimal controller turns out to have a structured realization as well. The topic of whether such realizations should exist in general, or whether sub-optimal structured stabilizing controllers should always have structured realizations is not discussed.

To the best of our knowledge, the only existing work dealing with structured realizability is the work of Vamsi and Elia [16, 17]. In these papers, the authors provide sufficient conditions for structured realizability, as well as an LMI approach for controller synthesis that guarantees realizability. However, they give no example of a provably non-realizable system, and the sufficient conditions provided are potentially very restrictive.

In Section 1, we showed that structured realizability is a meaningful concept by giving an example of a non-realizable transfer function. In the sections that follow, we provide some very general results; a parameterization of all structured stabilizing controllers, and a proof that $\mathcal{S}$-realizability is necessary for $\mathcal{S}$-stabilizability.

3 Preliminaries

3.1 Directed Graphs

A directed graph is a set of nodes $\mathcal{V} = \{1, \dots, N\}$ and edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. If $(i, j) \in \mathcal{E}$, we say that there is an edge from i to j, and we write $i \to j$. We make several assumptions regarding the graph.

$A1$) Self-loops: for all $i \in \mathcal{V}$, $i \to i$.

$A2$) Transitive closure: if $i \to j$ and $j \to k$, then $i \to k$.

$A3$) There are no directed cycles of length 2 or greater.

These assumptions are natural, and will be further justified in Section 3.2. Given a graph satisfying the assumptions above, define the adjacency matrix $\mathcal{S} \in \{0,1\}^{N\times N}$

$$\mathcal{S}_{ij} = \begin{cases} 1 & j \to i \\ 0 & \text{otherwise} \end{cases}$$

Under assumptions $A1$–$A3$, one can always relabel the nodes such that $\mathcal{S}$ is lower-triangular with a full diagonal. Therefore, we will assume a lower-triangular $\mathcal{S}$ from now on. See Figure 2 for an example of a graph and its associated adjacency matrix.

3.2 Systems over Graphs

The systems considered in this paper are linear, time-invariant, continuous-time, and rational. We denote the set of proper rational transfer functions as $\mathcal{R}_p$. If all the poles of $G \in \mathcal{R}_p$ are contained in the open left-half plane, G is *stable*, and we write $G \in \mathcal{RH}_\infty$.

For systems defined over graphs, additional notation is needed to keep track of the input and output partitions.

Definition 1 (Index sets)**.** *Suppose we have a graph $\mathcal{S} \in \{0,1\}^{N\times N}$. An* index set *$k$ is a tuple $(k_1, \dots, k_N)$ of nonnegative integers. We also define the set of nonempty indices as $\Omega_k = \{i \in \mathcal{V} \mid k_i \neq 0\}$.*

Definition 2. *Suppose we have a graph $\mathcal{S} \in \{0,1\}^{N\times N}$ with associated index sets k and m, and F is a commutative ring. We write $A \in \mathcal{S}(F, k, m)$ to mean that*

$$A = \begin{bmatrix} A_{11} & \cdots & A_{1N} \\ \vdots & \ddots & \vdots \\ A_{N1} & \cdots & A_{NN} \end{bmatrix}$$

If $j \to i$, then $A_{ij} \in F^{k_i \times m_j}$. Otherwise, $A_{ij} = 0$. If k and m are clear by context, we simply write $A \in \mathcal{S}(F)$.

A plant G defined over the graph $(\mathcal{V}, \mathcal{E})$ is written in its most general form as $G \in \mathcal{S}(\mathcal{R}_p, k, m)$. Intuitively, this means that if $i \to j$, then input u_i affects output y_j, and the associated transfer function is G_{ij}. If there is no edge from i to j, then $G_{ij} = 0$.

We seek controllers that obey the same structure. Namely, if $i \to j$ then the control signal u_j may depend on the measurement y_i, and the associated transfer function is K_{ij}. If there is no edge from i to j then $K_{ij} = 0$.

We can now see why Assumptions $A2$–$A3$ make sense. The graph $(\mathcal{V}, \mathcal{E})$ represents information flow; if a subcontroller i shares what it knows with subcontroller j along the link $i \to j$ and similarly for $j \to k$, one would expect subcontroller k to have access to the information from subcontroller i as well. Directed cycles can also be removed by treating all nodes involved as a single node.

Remark 3. *Assumption A2 leads to a useful algebraic property. If $G_1 \in \mathcal{S}(\mathcal{R}_p, k, m)$ and $G_2 \in \mathcal{S}(\mathcal{R}_p, m, p)$ then $G_1 G_2 \in \mathcal{S}(\mathcal{R}_p, k, p)$. In other words, structured transfer functions as defined above form an* algebra.

3.3 Structured Realizability

Structured realizability is a core concept in this paper. Roughly speaking, we are interested in finding a state-space realization for a structured plant comprised of matrices that are also structured.

Definition 4 (Structured realizability)**.** *A transfer function $G \in \mathcal{S}(\mathcal{R}_p, k, m)$ is said to be $\mathcal{S}(k,m)$–*realizable *if there exists an index set n and matrices $A \in \mathcal{S}(\mathbb{R}, n, n)$, $B \in \mathcal{S}(\mathbb{R}, n, m)$, $C \in \mathcal{S}(\mathbb{R}, k, n)$, and $D \in \mathcal{S}(\mathbb{R}, k, m)$ such that (A, B, C, D) is a stabilizable and detectable realization for G. When m, and k are clear by context, we will simply write that G is $\mathcal{S}$-realizable.*

Note that we allow the index set n to have zero-entries. If a component of n is zero, it means that no state is associated with that subsystem, and the corresponding rows and columns of A collapse.

A key aspect of this definition is the requirement of stabilizability and detectability. Indeed, we will see in Section 4.1 that one can always construct a trivial realization with the correct structure by introducing duplicate (non-minimal) states.

3.4 Stabilizability

Throughout this paper, we use the conventional notion of internal stability for feedback interconnections [2, 21]. If a plant G and controller K are connected in feedback as in Figure 3, then K is stabilizing if and only if the map $(u_1, u_1) \mapsto (y_1, y_2)$ is proper and stable. When dealing with decentralized systems, there are other natural ways to define stability. For a further discussion on this topic, see Section 6. We now state the formal input-output definition of stabilization.

Definition 5 (Stabilization)**.** *Suppose $G \in \mathcal{R}_p^{k\times m}$ and $K \in \mathcal{R}_p^{m\times k}$. We say that K* stabilizes *G if*

(i) $I - G(\infty)K(\infty)$ is invertible, and

(ii) $\begin{bmatrix} I & -G \\ -K & I \end{bmatrix}^{-1} \in \mathcal{RH}_\infty$

There is also a useful state-space characterization of stabilization, which we state as a proposition.

Proposition 6. *Suppose $G \in \mathcal{R}_p^{k\times m}$, $K \in \mathcal{R}_p^{m\times k}$ have realizations given by (A, B, C, D) and (A_K, B_K, C_K, D_K) respectively. Then the following are equivalent.*

(i) (C, A, B) and (C_K, A_K, B_K) are stabilizable and detectable, and K stabilizes G.

(ii) $(I - DD_K)$ is invertible, and $\bar{A}$ is Hurwitz, where

$$\bar{A} = \begin{bmatrix} A & 0 \\ 0 & A_K \end{bmatrix} + \begin{bmatrix} B & 0 \\ 0 & B_K \end{bmatrix} \begin{bmatrix} I & -D_K \\ -D & I \end{bmatrix}^{-1} \begin{bmatrix} 0 & C_K \\ C & 0 \end{bmatrix}.$$

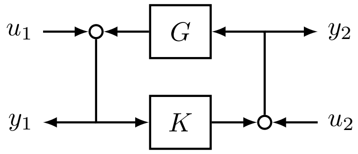


Figure 3: Feedback loop with inputs and outputs added to the feedback path.

Proof. Proposition 6 is a standard result [2, 21], though it is typically stated with the assumption that (C, A, B) is stabilizable and detectable. To prove the converse, note that if $\bar{A}$ is Hurwitz, then

$$\left(\begin{bmatrix} A & 0 \\ 0 & A_K \end{bmatrix}, \begin{bmatrix} B & 0 \\ 0 & B_K \end{bmatrix} \right) \qquad \text{is stabilizable.}$$

It follows from the PBH test that (A, B) is stabilizable as well. A similar argument holds for detectability. ■

In this paper, we seek controllers that are both stabilizing and have a particular structure. Therefore, we introduce new terminology to indicate this more restricted notion of stabilization.

Definition 7. *Suppose $G \in \mathcal{S}(\mathcal{R}_p, k, m)$. We say that G is $\mathcal{S}$-stabilizable if there exists $K \in \mathcal{S}(\mathcal{R}_p, m, k)$ such that K stabilizes G.*

Note that these definitions are symmetric; K stabilizes G if and only if G stabilizes K. The same symmetry relationship holds for $\mathcal{S}$-stabilization.

In the absence of structural constraints, it is well-known that every $G \in \mathcal{R}_p$ can be stabilized. The main thrust of this paper is to explain what happens when G is structured, and we seek a controller that is $\mathcal{S}$-stabilizing. We will see in Sections 4–5 that not every $G \in \mathcal{S}$ is $\mathcal{S}$-stabilizable, and $\mathcal{S}$-realizability plays an important role.

4 Simple Cases

For certain systems, finding a structured realization is straightforward. In this section, we explore two such cases that will be useful later: stable systems, and linear chain structures.

4.1 Stable Systems

If the plant is stable, it can always be $\mathcal{S}$-realized, regardless of the underlying graph. By duplicating states in a way that guarantees the correct structure, the resulting realization is always stabilizable and detectable since the starting plant was stable. We give the construction in the following lemma. A similar result appeared in [17].

Lemma 8. *Suppose $G \in \mathcal{S}(\mathcal{RH}_\infty, k, m)$. Then G is $\mathcal{S}$-realizable.*

Proof. We may construct a realization for G as follows. Separate G into its block-columns G_i for $i = 1, \ldots, N$.

Find minimal realizations

$$G_i = \left[\begin{array}{c|c} A_i & B_i \\ \hline C_i & D_i \end{array}\right].$$

In general, A_i and B_i will be full, but C_i and D_i will have zero-rows corresponding to the zero-rows in G_i. Now stack the columns side-by-side and obtain a joint realization.

$$\left[\left[\begin{array}{c|c} A_1 & B_1 \\ \hline C_1 & D_1 \end{array}\right] \quad \cdots \quad \left[\begin{array}{c|c} A_N & B_N \\ \hline C_N & D_N \end{array}\right]\right] =$$

$$\left[\begin{array}{ccc|ccc} A_1 & & & B_1 & & \\ & \ddots & & & \ddots & \\ & & A_N & & & B_N \\ \hline C_1 & \cdots & C_N & D_1 & \cdots & D_N \end{array}\right] \tag{3}$$

The realization (3) has the desired structure because A and B are block-diagonal, and C and D are in $\mathcal{S}(\mathbb{R})$. Finally, the realization is stabilizable and detectable since we started with a stable system and each A_i is stable. ■

In the case where G is not stable, the construction method used in the proof of Lemma 8 still produces a realization with the correct structure, but the realization may fail to be stabilizable and detectable. This is a consequence of the fact that realizing the block-columns separately and then re-combining them may cause some unstable poles to get duplicated. Indeed, this is precisely what happens if we attempt to realize (2).

4.2 Chain Structures

Linear chain structures correspond to adjacency matrices whose lower-triangular part is full. It turns out such systems are always $\mathcal{S}$-realizable.

Lemma 9. *Suppose $G \in \mathcal{S}(\mathcal{R}_p, k, m)$ and $\mathcal{S}$ has the full lower-triangular sparsity pattern*

$$\mathcal{S} = \begin{bmatrix} 1 & & 0 \\ \vdots & \ddots & \\ 1 & \cdots & 1 \end{bmatrix}$$

Then G is $\mathcal{S}$-realizable.

Proof. Let (A, B, C, D) be a minimal realization of G. We will sequentially construct a state transformation matrix T such that $(T^{-1}AT, T^{-1}B, CT, D)$ has the desired sparsity pattern. Consider the simplest case, $N = 2$. Partition the realization according to the block-structure

$$\begin{bmatrix} G_{11} & 0 \\ G_{21} & G_{22} \end{bmatrix} = \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & 0 \\ C_2 & D_{21} & D_{22} \end{array}\right]. \tag{4}$$

Note that D must already have the correct sparsity pattern. A realization for the zero-block G_{12} is

$$0 = \left[\begin{array}{c|c} A & B_2 \\ \hline C_1 & 0 \end{array}\right]. \tag{5}$$

Let T_1 be the transformation matrix that puts (5) into Kalman canonical form. There are typically four blocks in such a decomposition, but there will only be three in this case since the system we are realizing is identically zero and thus there can be no modes that are both controllable and observable. Applying T_1 to (4), we obtain

$$G = \left[\begin{array}{ccc|cc} A_{\bar{c}o} & 0 & 0 & B_{11} & 0 \\ A_{21} & A_{\bar{c}\bar{o}} & 0 & B_{21} & 0 \\ A_{31} & A_{32} & A_{c\bar{o}} & B_{31} & B_{c\bar{o}} \\ \hline C_{\bar{c}o} & 0 & 0 & D_{11} & 0 \\ C_{21} & C_{22} & C_{23} & D_{21} & D_{22} \end{array}\right]$$

This realization is block-lower-triangular, and we notice that the index sets n may not be unique. For example, the modes $A_{\bar{c}\bar{o}}$ can be part of either the A_{11} block or the A_{22} block.

For the case where $N > 2$, put the first block of the diagonal into G_{11} and lump the rest of the block-lower-triangular structure into G_{22}. Upon applying the T_1 found from the $N = 2$ case, we are left with

$$G_{22} = \left[\begin{array}{c|c} A_{c\bar{o}} & B_{c\bar{o}} \\ \hline C_{23} & D_{22} \end{array}\right].$$

Apply this approach recursively by finding a T_2 that puts the first zero-block-row of G_{22} into Kalman canonical form. Continuing in this manner eventually yields a realization for G that has the desired sparsity pattern. Furthermore, this realization is stabilizable and detectable since it is minimal. ■

5 Main Results

Our main results draw the connection between $\mathcal{S}$-stabilizability and $\mathcal{S}$-realizability. We will show that $\mathcal{S}$-stabilizable plants are always $\mathcal{S}$-realizable, and $\mathcal{S}$-stabilizing controllers are always $\mathcal{S}$-realizable. In other words, if structured systems connected in feedback results in a an internally stable interconnection, then both systems must be $\mathcal{S}$-realizable.

The main result has two essential ingredients. In Section 5.1, we assume the plant is $\mathcal{S}$-realizable. We then find a Youla-like parameterization of all $\mathcal{S}$-stabilizing controllers and show that all $\mathcal{S}$-stabilizing controllers are $\mathcal{S}$-realizable. In Section 5.2, we assume instead that our plant is $\mathcal{S}$-stabilizable, but make no assumptions regarding $\mathcal{S}$-realizability. We then show that there must exist a $\mathcal{S}$-stabilizing controller that is also $\mathcal{S}$-realizable. These results are combined into one concise statement, which we give in Corollary 14.

5.1 Parameterization of Stabilizing Controllers

In this subsection, we assume the plant is $\mathcal{S}$-realizable, and we give a characterization of all structured stabilizing controllers. This parameterization is similar to the classical work of Youla [19], and the unstructured version is one of the pillars of classical $\mathcal{H}_2/\mathcal{H}_\infty$ theory [21]. For the framework adopted herein, the sparsity constraint imposed on the controller amounts to the same sparsity constraint being imposed on the Youla parameter Q.

Similar parameterizations have appeared in the literature. Most notably, the work of Qi et.al. [7] treat the lower-triangular case and other cases of interest. A more general parameterization is provided in [11], where the authors consider general quadratically invariant systems in which the plant and controller may have different sparsity patterns.

Theorem 10. *Suppose $G \in \mathcal{S}(\mathcal{R}_p, k, m)$ is $\mathcal{S}$-realizable, with a structured realization given by (A, B, C, D).*

(i) G is $\mathcal{S}$-stabilizable if and only if (C_{ii}, A_{ii}, B_{ii}) is stabilizable and detectable for all $i \in \Omega_n$.

(ii) In this case, for $i \in \Omega_n$, choose F_i and L_i such that $A_{ii} + B_{ii}F_i$ and $A_{ii} + L_i C_{ii}$ are Hurwitz. Define $F_d = \mathbf{diag}\{F_i\}_{i\in\Omega_n}$ and $L_d = \mathbf{diag}\{L_i\}_{i\in\Omega_n}$. A particular $\mathcal{S}$-stabilizing controller is given by

$$K_0 = \left[\begin{array}{c|c} A + BF_d + L_dC + L_dDF_d & -L_d \\ \hline F_d & 0 \end{array}\right]$$

Proof. Suppose (C_{ii}, A_{ii}, B_{ii}) is stabilizable and detectable for all $i \in \Omega$. By definition, there must exist F_i and L_i such that $A_{ii} + B_{ii}F_i$ and $A_{ii} + L_iC_{ii}$ are Hurwitz. Define $F_d = \mathbf{diag}\{F_i\}_{i\in\Omega_n}$ and $L_d = \mathbf{diag}\{L_i\}_{i\in\Omega_n}$. Since A is block-lower-triangular, $A + BF_d$ and $A + L_dC$ are also Hurwitz. It is straightforward to check that K_0 defined in the theorem statement above is a stabilizing controller with the correct structure.

Conversely, suppose G is stabilizable by some $K \in \mathcal{S}(\mathcal{R}_p, m, k)$. Let $\bar{\mathcal{S}}$ be the adjacency matrix with a full lower-triangular sparsity pattern. Then $K \in \bar{\mathcal{S}}(\mathcal{R}_p, m, k)$. By Lemma 9, K is $\bar{\mathcal{S}}$-realizable. Let (A_K, B_K, C_K, D_K) be a stabilizable and detectable realization for K structured according to $\bar{\mathcal{S}}$. By Proposition 6, $(I - DD_K)$ must be invertible, and $\bar{A}$ must be Hurwitz. Rearrange the columns and rows of $\bar{A}$ by taking the first sub-blocks of each large block, then the second sub-blocks, and so on. The resulting matrix is block-lower-triangular, and has the same eigenvalues as $\bar{A}$. The blocks along the main diagonal are

$$\begin{bmatrix} A_{ii} & 0 \\ 0 & A_{Kii} \end{bmatrix} + \begin{bmatrix} B_{ii} & 0 \\ 0 & B_{Kii} \end{bmatrix} \begin{bmatrix} I & -D_{Kii} \\ -D_{ii} & I \end{bmatrix}^{-1} \times \begin{bmatrix} 0 & C_{Kii} \\ C_{ii} & 0 \end{bmatrix} \quad \text{for } i \in \Omega_n \tag{6}$$

Applying Proposition 6 once more, we conclude that (C_{ii}, A_{ii}, B_{ii}) is stabilizable and detectable. ∎

The second result is a parameterization of all $\mathcal{S}$-stabilizing controllers for $\mathcal{S}$-realizable plants.

Theorem 11. *Suppose $G \in \mathcal{S}(\mathcal{R}_p, k, m)$ is $\mathcal{S}$-realizable, with a structured realization given by (A, B, C, D). Let*

$$J = \left[\begin{array}{c|cc} A + BF_d + L_dC + L_dDF_d & -L_d & B + L_dD \\ \hline F_d & 0 & I \\ -(C + DF_d) & I & -D \end{array}\right]$$

where F_d and L_d are defined as in Theorem 10. Also, define $F_l(J, Q) = J_{11} + J_{12}Q(I - J_{22}Q)^{-1}J_{21}$.

(i) Every $K \in \mathcal{S}(\mathcal{R}_p, m, k)$ that stabilizes G is parameterized by $K = F_l(J, Q)$, where $Q \in \mathcal{S}(\mathcal{RH}_\infty, m, k)$ such that $I + DQ(\infty)$ is nonsingular.

(ii) Every stabilizing controller is $\mathcal{S}$-realizable.

Proof. By the proof of Theorem 10, $A + BF_d$ and $A + L_dC$ are Hurwitz. The classical parameterization of all stabilizing controllers [21] is given by $F_l(J, Q)$ where J is defined in the theorem statement above, and $Q \in \mathcal{RH}_\infty$ such that $I + DQ(\infty)$ is nonsingular. It remains to show that $K \in \mathcal{S}$ if and only if $Q \in \mathcal{S}$. This is a straightforward consequence of Remark 3, and the fact that all subblocks of J are in $\mathcal{S}$.

We will now show that every stabilizing controller is $\mathcal{S}$-realizable. Note that $Q \in \mathcal{S}(\mathcal{RH}_\infty, m, k)$, so we may apply Lemma 8. Let (A_Q, B_Q, C_Q, D_Q) be a structured realization for Q that is stabilizable and detectable. The associated controller has a realization

$$F_l(J, Q) = \left[\begin{array}{cc|c} \hat{A}_{11} & \hat{A}_{12} & \hat{B}_1 \\ \hat{A}_{21} & \hat{A}_{22} & \hat{B}_2 \\ \hline \hat{C}_1 & \hat{C}_2 & \hat{D} \end{array}\right]$$

which can be computed using the Redheffer Star-Product [21, §10]. For example, the $\hat{A}$ and $\hat{B}$ terms are:

$$\begin{aligned} \hat{A}_{11} &= A + BF_d + L_dC + L_dDF_d \\ &\qquad - (B + L_dD)(I + D_QD)^{-1}(C + DF_d) \\ \hat{A}_{12} &= (B + L_dD)(I + D_QD)^{-1}C_Q \\ \hat{A}_{21} &= -B_Q(I + DD_Q)^{-1}(C + DF_d) \\ \hat{A}_{22} &= A_Q - B_Q(I + DD_Q)^{-1}DC_Q \\ \hat{B}_1 &= -L_d + (B + L_dD)(I + D_QD)^{-1}D_Q \\ \hat{B}_2 &= B_Q(I + DD_Q)^{-1} \end{aligned}$$

Using Remark 3 and the fact that the realization for Q is structured, it is straightforward to show that the $\hat{A}_{ij}$ and $\hat{B}_i$ are each structured according to $\mathcal{S}$. The same is true of the $\hat{C}_i$ and $\hat{D}$. Stabilizability can be verified explicitly. For example, one can check that

$$\begin{bmatrix} \hat{A}_{11} & \hat{A}_{12} \\ \hat{A}_{21} & \hat{A}_{22} \end{bmatrix} + \begin{bmatrix} \hat{B}_1 \\ \hat{B}_2 \end{bmatrix} \begin{bmatrix} C + DF_d & DC_Q \end{bmatrix} = \begin{bmatrix} A + BF_d & BC_Q \\ 0 & A_Q \end{bmatrix}$$

which is clearly Hurwitz. A similar argument applies for detectability. Finally, we can permute the states as we did in the proof of Theorem 11 to obtain a realization for $F_l(J, Q)$ for which all matrices are in $\mathcal{S}$. This shows that the controller is $\mathcal{S}$-realizable, as required. ∎

5.2 Stabilizable Systems are Realizable

In this subsection, we assume the plant is $\mathcal{S}$-stabilizable, but make no assumptions regarding $\mathcal{S}$-realizability. In this case, we show that one can always construct an $\mathcal{S}$-realizable controller.

Theorem 12. *Suppose $G \in \mathcal{S}(\mathcal{R}_p, k, m)$ is stabilized by some $K \in \mathcal{S}(\mathcal{R}_p, m, k)$. Then,*

(i) G is stabilized by $K_d = \mathbf{diag}(K_{11}, \dots, K_{NN})$.

(ii) K_d is $\mathcal{S}$-realizable.

Proof. The proof is similar to the proof of Theorem 10. G and K may not be $\mathcal{S}$-realizable, so we embed their sparsity patterns in $\bar{\mathcal{S}} \supset \mathcal{S}$, the set of block-lower-triangular transfer matrices. Apply Lemma 9 to find triangular realizations, and the result follows from Proposition 6. ∎

Theorem 12 provides a simple test for $\mathcal{S}$-stabilizability; stabilizing each subsystem individually must also globally stabilize the system. Otherwise, no $\mathcal{S}$-stabilizing controller exists.

Corollary 13. *Suppose $G \in \mathcal{S}(\mathcal{R}_p, k, m)$. For $i = 1, \dots, N$, let $K_i \in \mathcal{R}_p^{m_i \times k_i}$ be any controller that stabilizes G_{ii}. Define $K_d = \mathbf{diag}(K_1, \dots, K_N)$. Then G is $\mathcal{S}$-stabilizable if and only if K_d stabilizes G.*

Proof. If K_d stabilizes G, then clearly G is $\mathcal{S}$-stabilizable, since $K_d \in \mathcal{S}$. The converse follows from Theorem 12. ∎

The main result of this paper is that stabilizable plants are always $\mathcal{S}$-realizable. We state it below as a Corollary.

Corollary 14 (Main Result)**.** *Suppose $G \in \mathcal{S}(\mathcal{R}_p, k, m)$ is $\mathcal{S}$-stabilizable. Then G is $\mathcal{S}$-realizable, and every $K \in \mathcal{S}(\mathcal{R}_p, m, k)$ that stabilizes G is $\mathcal{S}$-realizable.*

Proof. By Theorem 12, every $\mathcal{S}$-stabilizable plant G can be stabilized by an $\mathcal{S}$-realizable controller K_d. Interchanging the roles of the plant and controller, K_d is an $\mathcal{S}$-realizable plant stabilized by G. By Theorem 11, G must be $\mathcal{S}$-realizable. Applying Theorem 11 once more to G, we conclude that all stabilizing controllers must be $\mathcal{S}$-realizable. ∎

Note that the converse of Corollary 14 is generally false. One can construct an $\mathcal{S}$-realizable plant that is not $\mathcal{S}$-stabilizable. Consider for example:

$$G = \begin{bmatrix} \frac{1}{s+1} & 0 \\ \frac{1}{s-1} & \frac{1}{s+1} \end{bmatrix} = \left[\begin{array}{ccc|cc} -1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 1 \\ \hline 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{array}\right]$$

If we realize G by clustering the unstable mode with A_{11}, i.e. using the index set $n = (2, 1)$, it leads to an undetectable (C_{11}, A_{11}). If we instead associate the unstable mode with A_{22} by using $n = (1, 2)$, it will lead to an unstabilizable (A_{22}, B_{22}). By Theorem 10, this plant is not stabilizable by a lower-triangular controller. Of course, non-structured stabilizing controllers exist since the realization for G given above is minimal.

6 Concluding Remarks

The main contribution of this paper is to establish the connection between the structured versions of stabilizability and realizability. In the context of structured controller synthesis, it is a basic requirement that the plant is $\mathcal{S}$-stabilizable and that the controller is $\mathcal{S}$-stabilizing. By Corollary 14, it is necessary that the plant and controller are both $\mathcal{S}$-realizable. The counterexample (2) causes no problems because as a plant, G_2 would not be $\mathcal{S}$-stabilizable, and as a controller, G_2 could not stabilize any structured plant.

As mentioned in Section 2, the work of Vamsi and Elia [17, 16] also addresses the issue of structured realizability. The underlying assumptions are different however. The authors assume discrete-time subsystems with one-timestep delays along directed edges. In the work herein, we assumed a delay-free network of continuous-time systems. Some results, such as the counterexample (2), are valid in either framework. However, the other results in this paper do not immediately translate to discrete-time.

In this paper, we adopted the classical definition of internal stability. This notion of stabilization is the weakest one possible, since it does not guarantee that the signals that travel between subsystems or between subcontrollers remains bounded. It follows that Corollary 14 will remain true even for stricter notions of stabilization. The work of Yadav et.al. [18] explores other types of stability in great detail.

While $\mathcal{S}$-stabilizability guarantees $\mathcal{S}$-realizability, the question of how to explicitly construct a structured realization is still open. Lemmas 8 and 9 provide constructions for the special cases of stable and triangular systems respectively, but no universal method for constructing structured realizations is known.

References

[1] G. E. Dullerud and R. D'Andrea. Distributed control of heterogeneous systems. *IEEE Transactions on Automatic Control*, 49(12):2113–2128, 2004.

[2] G. E. Dullerud and F. Paganini. *A Course in Robust Control Theory: A Convex Approach.* Springer Verlag, 2000.

[3] J.-H. Kim and S. Lall. Separable optimal cooperative control problems. In *American Control Conference*, 2012. (To Appear).

[4] L. Lessard. Decentralized lqg control of systems with a broadcast architecture. In *IEEE Conference on Decision and Control*, 2012. (To Appear).

[5] L. Lessard and S. Lall. Reduction of decentralized control problems to tractable representations. In *IEEE Conference on Decision and Control*, volume 1, pages 1621–1626, 2009.

[6] L. Lessard and S. Lall. A state-space solution to the two-player decentralized optimal control problem. In *Allerton Conference on Communication, Control, and Computing*, pages 1559–1564, 2011.

[7] X. Qi, M. Salapaka, P. Voulgaris, and M. Khammash. Structured optimal and robust control with multiple criteria: A convex solution. *IEEE Transactions on Automatic Control*, 49(10):1623–1640, 2004.

[8] A. Rantzer. Linear quadratic team theory revisited. In *American Control Conference*, pages 1637–1641, 2006.

[9] M. Rotkowitz, R. Cogill, and S. Lall. Convexity of optimal control over networks with delays and arbitrary topology. *International Journal of Systems, Control and Communications*, 2(1):30–54, 2010.

[10] M. Rotkowitz and S. Lall. Convexification of optimal decentralized control without a stabilizing controller. In *Proceedings of the International Symposium on Mathematical Theory of Networks and Systems (MTNS)*, pages 1496–1499, 2006.

[11] S. Sabău and N. Martins. On the stabilization of LTI decentralized configurations under quadratically invariant sparsity constraints. In *Allerton Conference on Communication, Control, and Computing*, pages 1004–1010, 2010.

[12] C. W. Scherer. Structured finite-dimensional controller design by convex optimization. *Linear Algebra and its Applications*, 351-352:639–669, 2002.

[13] P. Shah and P. A. Parrilo. $\mathcal{H}_2$-optimal decentralized control over posets: A state space solution for state-feedback. In *IEEE Conference on Decision and Control*, pages 6722–6727, 2010.

[14] J. Swigart. *Optimal Controller Synthesis for Decentralized Systems*. PhD thesis, Stanford University, 2010.

[15] J. Swigart and S. Lall. A graph-theoretic approach to distributed control over networks. In *IEEE Conference on Decision and Control*, pages 5409–5414, 2009.

[16] A. S. M. Vamsi and N. Elia. Design of distributed controllers realizable over arbitrary directed networks. In *IEEE Conference on Decision and Control*, pages 4795–4800, 2010.

[17] A. S. M. Vamsi and N. Elia. Optimal realizable networked controllers for networked systems. In *American Control Conference*, pages 336–341, 2011.

[18] V. Yadav, M. Salapaka, and P. Voulgaris. Architectures for distributed controller with sub-controller communication uncertainty. *IEEE Transactions on Automatic Control*, 55(8):1765–1780, aug. 2010.

[19] D. Youla, H. Jabr, and J. Bongiorno, J. Modern wiener-hopf design of optimal controllers–Part II: The multivariable case. *IEEE Transactions on Automatic Control*, 21(3):319–338, 1976.

[20] D. Zelazo and M. Mesbahi. $\mathcal{H}_2$ analysis and synthesis of networked dynamic systems. In *American Control Conference*, pages 2966–2971, 2009.

[21] K. Zhou, J. Doyle, and K. Glover. *Robust and optimal control*. Prentice-Hall, 1995.

A Proof of Counterexample

In this section, we provide a proof that the example (2) of Figure 2 is not $\mathcal{S}_2$-realizable.

Proof. Suppose G_2 has a stabilizable and detectable realization in $\mathcal{S}_2$. Label the blocks of this realization as

$$A = \begin{bmatrix} A_{11} & 0 & 0 & 0 \\ 0 & A_{22} & 0 & 0 \\ A_{31} & A_{32} & A_{33} & 0 \\ A_{41} & A_{42} & 0 & A_{44} \end{bmatrix}.$$

and similarly for B and C. Expand $C(sI-A)^{-1}B$ and equate with G_2. The four equations corresponding to each of the nonzero entries G_{ij} are

$$\begin{bmatrix} C_{ij} & C_{ii} \end{bmatrix} \left(sI - \begin{bmatrix} A_{jj} & 0 \\ A_{ij} & A_{ii} \end{bmatrix} \right)^{-1} \begin{bmatrix} B_{jj} \\ B_{ij} \end{bmatrix} = \frac{1}{s-1} \qquad (7)$$

for $(i,j) \in \big\{(3,1),(3,2),(4,1),(4,2)\big\}$. A minimal realization for G_2 is given by

$$G_2 = \left[\begin{array}{c|cccc} 1 & 1 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{array}\right]$$

Since we have assumed our realization to be stabilizable and detectable, the eigenvalues of A must consist of the unstable eigenvalue 1, together with some number of stable eigenvalues. Since A is block-lower-triangular, the eigenvalue 1 must appear as an eigenvalue of exactly one of the A_{ii} for $i = 1, \ldots, 4$. For $(i,j) = (3,1)$, we deduce from (7) that the eigenvalue 1 is contained in either A_{11} or A_{33}, but for $(i,j) = (4,2)$, we deduce that it is contained in either A_{22} or A_{44}. This contradiction implies that no structured stabilizable and detectable realization of G_2 can exist. ■