Minimum Cost Feedback Selection for Arbitrary Pole Placement in Structured Systems

Shana Moothedath, Prasanna Chaporkar and Madhu N. Belur

***Abstract*—This paper addresses optimal feedback selection for arbitrary pole placement of structured systems when each feedback edge is associated with a cost. Given a structured system and a feedback cost matrix, our aim is to find a feasible feedback matrix of minimum cost that guarantees arbitrary pole placement of the closed-loop structured system. The same problem but with uniform costs has been considered recently, but the claim and the NP-hardness proof there has subtle flaws: we mention this briefly in our paper and then prove the NP-hardness of the feedback selection problem using a reduction from the weighted set cover problem. We next prove the polynomial time constant factor inapproximability of the problem by showing that a constant factor approximation for the problem does not exist, unless the weighted set cover problem can also be approximated within a constant factor. We study a subclass of systems whose directed acyclic graph constructed using the strongly connected components of the state digraph is a line graph and the state bipartite graph has a perfect matching. We propose a polynomial time algorithm based on dynamic programming principles for optimal feedback selection on this class of systems. Further, over the same class of systems we relax the perfect matching assumption, and provide a polynomial time 2-optimal solution using a minimum cost perfect matching algorithm.**

***Index Terms*—Linear structured systems, Arbitrary pole placement, Linear output feedback, Minimum cost control selection.**

1. Introduction

In this paper we address optimal output feedback selection for arbitrary pole-placement in large-scale dynamical systems. Given a dynamical system, finding an output feedback that guarantees arbitrary pole-placement of the closed-loop system is an important design problem in control. In many large-scale systems, including biological systems, electronic circuits, transportation, World Wide Web, social communication, power grids and multi-agent systems, the exact link weights of the graph are not known. Hence, many papers use the topological characteristics of the network for analysing them. Moreover, the size of these networks are large, that efficient frameworks for solving various optimization problems on these networks are indispensable. We refer to [1], [2] and references therein to note the applicability of structural analysis to several real networks (see after Theorem 1 for an elaboration of specific applications).

Consider structured matrices $\bar{A},\bar{B}$ and $\bar{C}$ whose entries are $\star$'s and 0's that represent an equivalence class of control systems whose system dynamics is governed by $\dot{x}=Ax+Bu$, $y=Cx$, where $A\in\mathbb{R}^{n\times n}$, $B\in\mathbb{R}^{n\times m}$ and $C\in\mathbb{R}^{p\times n}$ has the same structure as that of $\bar{A},\bar{B}$ and $\bar{C}$ respectively. Here, $\mathbb{R}$ denotes the set of real numbers and we assume that m and p are of the order of n. More precisely,

$$\begin{aligned} A_{ij} &= 0 \text{ whenever } \bar{A}_{ij}=0, \text{ and}\\ B_{ij} &= 0 \text{ whenever } \bar{B}_{ij}=0, \text{ and}\\ C_{ij} &= 0 \text{ whenever } \bar{C}_{ij}=0. \end{aligned} \tag{1}$$

Any triple (A,B,C) that satisfies (1) is referred as a *numerical realization* of the *structured system* $(\bar{A},\bar{B},\bar{C})$. Let $P\in\mathbb{R}^{m\times p}$ denotes the feedback cost matrix, where P_{ij} is the cost for feeding the j^{th} output to the i^{th} input. It may not be feasible to connect all the outputs to all the inputs. When infeasible, we model this by considering cost of such a connection to be infinite. To denote the feedback connections made, we use matrix $\bar{K}\in\{0,\star\}^{m\times p}$, where $\bar{K}_{ij}=\star$, if j^{th} output can be fed to i^{th} input. Now the cost of $\bar{K}$ denoted by $P(\bar{K})$ is given by $P(\bar{K})=\sum_{(i,j):\bar{K}_{ij}=\star}P_{ij}$. For a given $\bar{K}$ we define, $[K]:=\{K: K_{ij}=0, \text{ if } \bar{K}_{ij}=0\}$.

The authors are in the Department of Electrical Engineering, Indian Institute of Technology Bombay, India. Email: {shana, chaporkar, belur}@ee.iitb.ac.in.

Definition 1. *The structured system $(\bar{A},\bar{B},\bar{C})$ and the feedback matrix $\bar{K}$ is said to have no structurally fixed modes if there exists a numerical realization (A,B,C) of $(\bar{A},\bar{B},\bar{C})$ such that $\cap_{K\in[K]}\sigma(A+BKC)=\phi$, where $\sigma(T)$ denotes the set of eigenvalues of a square matrix T.*

If a structured system has no structurally fixed modes (SFMs), then arbitrary pole placement is possible for *almost all* numerical realizations of it using output feedback. A numerical realization of a structured system with no SFMs that does not allow arbitrary pole-placement lies in a thin[1] set. The strength of this analysis is that it requires only the zero/non-zero pattern of the system to study an equivalence class of systems whose sparsity patterns are the same. Using the structural framework we aim to design an optimal $\bar{K}$ for a given $(\bar{A},\bar{B},\bar{C})$. Then, for almost all numerical realizations of the structured system, it is guaranteed that there exists a feedback matrix of the given structure which allows arbitrary pole-placement.

In this paper our aim is to find a minimum cost $\bar{K}$ such that the closed-loop system $(\bar{A},\bar{B},\bar{C},\bar{K})$ has no SFMs. Specifically, we wish to solve the following optimization problem: given $(\bar{A},\bar{B},\bar{C})$ and a feedback cost matrix P, let $\mathcal{K}_s:=\{\bar{K}\in\{0,\star\}^{m\times p}: (\bar{A},\bar{B},\bar{C},\bar{K}) \text{ has no SFMs}\}$. Note that $\mathcal{K}_s$ consists of all possible feedback structured matrices $\bar{K}$ such that the closed-loop system $(\bar{A},\bar{B},\bar{C},\bar{K})$ has no SFMs.

Problem 1. *Given a structured system $(\bar{A},\bar{B},\bar{C})$ and feedback cost matrix P, find $\bar{K}^\star \in \arg\min_{\bar{K}\in\mathcal{K}_s} P(\bar{K})$.*

We refer to Problem 1 as *the minimum cost feedback selection problem for arbitrary pole placement*. This problem has received attention since a few decades ago ([4], [5]) and more recently in [6], [7]. In fact, a special case of the above problem, namely when all the edges have uniform cost, and the state bipartite has a perfect matching and inputs and outputs are dedicated[2], has been considered in [6], but a key 'inference' from a lemma ([6, Lemma 4]) to the main result ([6, Theorem 5]) has a subtle flaw[3]. We elaborate on this later below in Section 2-B. This paper deals with the non-uniform cost version with non-dedicated inputs and outputs.

Let $p^\star=P(\bar{K}^\star)$ denote the optimal cost of Problem 1. For a structured system $(\bar{A},\bar{B},\bar{C})$ with feedback cost matrix P, without loss of generality, we assume that $\mathcal{K}_s$ is non-empty. Specifically, $\bar{K}^f\in\mathcal{K}_s$, where $\bar{K}^f_{ij}=\star$ for all i,j. Notice that if $\mathcal{K}_s$ is empty, then for every $\bar{K}$, the closed-loop structured system $(\bar{A},\bar{B},\bar{C},\bar{K})$ has SFMs and if $p^\star=+\infty$, then we say that arbitrary pole placement is not possible for that structured system.

[1] A non-trivial algebraic variety is 'thin' and a set of measure zero. See [3] for more details.

[2] An input (output, resp.) is said to be dedicated, if it can actuate (sense, resp.) a single state only.

[3] Note that, [6, Lemma 4] is indeed correct but its use as a special case in [6, Theorem 5] is where the flaw exists.

In this paper, we show the hardness of the above problem using the weighted set cover problem. In addition we also show the polynomial time constant factor inapproximability of the problem. Specifically, we have the following result as one of our main result (see Section 3).

Theorem 1. *Consider a structured system* $(\bar{A},\bar{B},\bar{C})$ *with n states and feedback cost matrix P. Then, there does not exist a polynomial time algorithm for solving Problem 1 that has approximation ratio* $(1-o(1))\log n$.

See Section 3 for the proof. For the rest of this paper we will use inapproximability to refer to polynomial time constant factor inapproximability.

Though Problem 1 is NP-hard and even approximating it within a multiplicative factor of $O(\log n)$ is not feasible in polynomial time on general systems, we give an $O(n^3)$ algorithm based on dynamic programming for a special class of systems: those in which the directed acyclic graph (DAG) obtained by condensing the strongly connected components (SCCs) of the state digraph to vertices (see Section 2 for more details) form a line graph[4]. The motivation for considering this structure comes from cascade or series connected systems. *Irreducible*[5] systems connected in cascade have line graph structure. This connection is also referred as chain graph, and radial graph, and path graph in the literature.

The line graph topology is useful in many applications including vehicle platooning control [8], [9], [10], multi-level voltage source inverters (VSI) [11], assembly lines and series of processing plants. While line graph topology is vitally assumed in VSI and other applications, due to paucity of space we elaborate on vehicle platooning control. Platoon formation control of vehicles has been recognized as a potential strategy for improving traffic efficiency, enhancing road safety and reducing fuel consumption [9]. The goal of longitudinal platoon control is to ensure that all the vehicles move in the same lane and at the same speed with a pre-specified inter-vehicle distance. Platoon control adjusts vehicle spatial distribution such that roadway utilization is maximized while collision is eliminated. Many control methodologies have been applied, including PID controllers, state feedback, adaptive control, state observers, among others, with safety, string stability, and team coordination as the most common objectives [8]. It is helpful to have results/algorithms for the line graph topology case, since as we know, the general topology case is NP-hard and also inapproximable to constant factor in polynomial time.

For systems with line graph topology, we consider two different cases: (i) when the state bipartite graph (see Section 2 for more details) has a perfect matching, and (ii) when the state bipartite graph does not have a perfect matching. Note that there exists a wide class of systems called as *self-damped* systems [12] that have a perfect matching in the state bipartite graph, for example consensus dynamics in multi-agent systems and epidemic dynamics. All systems with invertible state matrices have a perfect matching in their state bipartite graph. For the class of systems whose state bipartite graph does not have a perfect matching but the DAG of SCCs is a line graph, we give an $O(n^3)$ algorithm based on dynamic programming and minimum cost perfect matching that gives a 2-optimal solution to Problem 1. Note that if the DAG of SCCs has a spanning tree that is a line graph, our results hold (see Remark 2 and Figure3). We have the following theorem (see Section 4 for the proof) as another main result.

Theorem 2. *Consider a structured system* $(\bar{A},\bar{B},\bar{C})$ *and a feedback cost matrix P given as input to Algorithms 2 and 3. Let the DAG of SCCs of the state digraph be a line graph and* $p^\star$ *denote the optimal cost of Problem 1. Then,*

(i) *if state bipartite graph has a perfect matching, then output* $\bar{K}^a$ *of Algorithm 2 is an optimal solution to Problem 1, i.e.,* $P(\bar{K}^a) = p^\star$.
(ii) *if state bipartite graph does not have a perfect matching, then output* $\bar{K}^{ab}$ *of Algorithm 3 is a 2-optimal solution to Problem 1, i.e.,* $P(\bar{K}^{ab}) \leqslant 2\,p^\star$.

The organization of the paper is as follows: in Section 2 we discuss graph theoretic preliminaries used in the sequel, few existing results, related work in this area and our key contributions. In Section 3 we show the NP-hardness of the problem using a reduction of the weighted set cover problem. We also give the negative approximation result of the problem in this section. In Section 4 we discuss two special classes of linear dynamical systems. For the first class of systems considered we give a polynomial time optimal algorithm based on dynamic programming for solving Problem 1. For the second class of systems, we give a polynomial time 2-optimal approximation algorithm for solving Problem 1. In Section 5 we explain our dynamic programming based algorithms given in Section 4 through illustrative examples. Finally, Section 6 contains some concluding remarks.

2. Preliminaries and Related Work

In this section we first discuss few graph theoretic preliminaries and existing results. Subsequently, we discuss the related work in the area of feedback selection problem and then describe the contribution of our paper.

A. Preliminaries and Existing Results

We define the state digraph $\mathcal{D}(\bar{A}) := (V_X, E_X)$ where $V_X = \{x_1,\ldots,x_n\}$ and an edge $(x_j,x_i) \in E_X$ if $\bar{A}_{ij} \neq 0$. Thus a directed edge (x_j,x_i) exists if state x_j can *influence* state x_i. We next define the system digraph $\mathcal{D}(\bar{A},\bar{B},\bar{C}) := (V_X \cup V_U \cup V_Y, E_X \cup E_U \cup E_Y)$, where $V_U = \{u_1,\ldots,u_m\}$ and $V_Y = \{y_1,\ldots,y_p\}$. An edge $(u_j,x_i) \in E_U$ if $\bar{B}_{ij} \neq 0$ and an edge $(x_j,y_i) \in E_Y$ if $\bar{C}_{ij} \neq 0$. Thus a directed edge (u_j,x_i) exists if input u_j can *actuate* state x_i and a directed edge (x_j,y_i) exists if output y_i can *sense* state x_j. Then the closed-loop system digraph $\mathcal{D}(\bar{A},\bar{B},\bar{C},\bar{K}) := (V_X \cup V_U \cup V_Y, E_X \cup E_U \cup E_Y \cup E_K)$, where $(y_j,u_i) \in E_K$ if $\bar{K}_{ij} \neq 0$. Here a directed edge (y_j,u_i) exists if output y_j can be *fed to* input u_i.

A digraph is said to be strongly connected if for each ordered pair of vertices (v_i,v_k) there exists a path from v_i to v_k. A strongly connected component (SCC) is a subgraph that consists of a maximal set of strongly connected vertices. Using the SCCs of $\mathcal{D}(\bar{A})$ we construct a directed acyclic graph (DAG), where each node in the DAG is an SCC of $\mathcal{D}(\bar{A})$. Also, the edges in the DAG are such that, there exists an edge between two nodes in the DAG if and only if there exists an edge in $\mathcal{D}(\bar{A})$ that connects two states in those SCCs. For the state digraph $\mathcal{D}(\bar{A})$, SCCs are characterized as follows.

Definition 2. *An SCC is said to be linked if it has atleast one incoming or outgoing edge from another SCC. Further, an SCC is said to be non-top linked (non-bottom linked, resp.) if it has no incoming (outgoing, resp.) edges to (from, resp.) its vertices from (to, resp.) the vertices of another SCC. Non-top/non-bottom linked SCCs are also referred as source/sink SCCs in the literature.*

Now using the closed-loop system digraph $\mathcal{D}(\bar{A},\bar{B},\bar{C},\bar{K})$ the following result has been shown [13].

Proposition 1 ([13], Theorem 4)**.** *A structured system* $(\bar{A},\bar{B},\bar{C})$ *have no structurally fixed modes with respect to an information pattern* $\bar{K}$ *if and only if the following conditions hold: a) in the digraph* $\mathcal{D}(\bar{A},\bar{B},\bar{C},\bar{K})$, *each state node* x_i *is contained in an SCC which*

[4]A directed line graph is a graph which is a directed path starting at the root vertex and ending at the tip vertex.

[5]A system is said to be irreducible if its digraph is strongly connected.

includes an edge from E_K, and b) there exists a finite node disjoint union of cycles[6] $\mathcal{C}_g = (V_g, E_g)$ *in* $\mathcal{D}(\bar{A},\bar{B},\bar{C},\bar{K})$*, where g belongs to the set of natural numbers such that* $V_X \subseteq \cup_g V_g$.

Condition a) can be checked in $O(n^2)$ operations [14] by finding all SCCs in $\mathcal{D}(\bar{A},\bar{B},\bar{C},\bar{K})$ and verifying if each of them has atleast one feedback edge in it. Condition b) can be checked in $O(n^{2.5})$ [14] operations using the information path condition given in [13]. If $\mathcal{D}(\bar{A})$ is a single SCC, then the graph is said to be *irreducible*. In such a case satisfying condition a) in Proposition 1 is trivial as any single (y_i,u_j) edge is enough to satisfy the required. Then, solving Problem 1 simplifies to satisfying condition b) optimally which is polynomial [15]. Define $\mathcal{B}(\bar{A},\bar{B},\bar{C},\bar{K}) := (V_{X'} \cup V_{U'} \cup V_{Y'}, V_X \cup V_U \cup V_Y, \mathcal{E}_X \cup \mathcal{E}_U \cup \mathcal{E}_Y \cup \mathcal{E}_K \cup \mathcal{E}_{\mathbb{U}} \cup \mathcal{E}_{\mathbb{Y}})$, where $V_{X'} = \{x'_1,\ldots,x'_n\}$, $V_{U'} = \{u'_1,\ldots,u'_m\}$, $V_{Y'} = \{y'_1,\ldots,y'_p\}$ and $V_X = \{x_1,\ldots,x_n\}$, $V_U = \{u_1,\ldots,u_m\}$ and $V_Y = \{y_1,\ldots,y_p\}$. Also, $(x'_i,x_j) \in \mathcal{E}_X \Leftrightarrow (x_j,x_i) \in E_X$, $(x'_i,u_j) \in \mathcal{E}_U \Leftrightarrow (u_j,x_i) \in E_U$, $(y'_j,x_i) \in \mathcal{E}_Y \Leftrightarrow (x_i,y_j) \in E_Y$ and $(u'_i,y_j) \in \mathcal{E}_K \Leftrightarrow (y_j,u_i) \in E_K$. Moreover, $\mathcal{E}_{\mathbb{U}}$ include edges (u'_i,u_i) for $i = 1,\ldots,m$ and $\mathcal{E}_{\mathbb{Y}}$ include edges (y'_j,y_j) for $j = 1,\ldots,p$. Given a bipartite graph $G_B := (V,\widetilde{V},\mathcal{E})$, where $V \cap \widetilde{V} = \phi$ and $\mathcal{E} \subseteq V \times \widetilde{V}$, a matching M is a collection of edges $M \subseteq \mathcal{E}$ such that no two edges in M share a common end point. For $|V| = |\widetilde{V}|$, if $|M| = |V|$, then M is said to be a perfect matching, where $|D|$ denotes the cardinality of a set D. Using the bipartite graph construction given above [13], a matching condition for checking condition b) is given in [16].

Proposition 2 ([16], Theorem 3)**.** *Consider a closed-loop structured system* $(\bar{A},\bar{B},\bar{C},\bar{K})$*. Then, the bipartite graph* $\mathcal{B}(\bar{A},\bar{B},\bar{C},\bar{K})$ *has a perfect matching if and only if all state nodes are spanned by disjoint union of cycles in* $\mathcal{D}(\bar{A},\bar{B},\bar{C},\bar{K})$.

In a special case, if the state bipartite graph $\mathcal{B}(\bar{A}) := (V_{X'},V_X,\mathcal{E}_X)$ has a perfect matching, then all state nodes lie in node disjoint cycles that consists of only x_i's. Thus condition b) is satisfied even without using any feedback edge. Summarizing, in general given a closed-loop structured system $(\bar{A},\bar{B},\bar{C},\bar{K})$ we can check if the system has SFMs or not in $O(n^{2.5})$ computations. Unlike checking for SFMs, designing minimum cost feedback matrix such that the closed-loop system does not have SFMs is computationally hard. Now we discuss some related work in this area.

B. Related Work and Contributions

In this section, we discuss only the relevant literature. Structural analysis for various other problems can be found in [17] and references therein, while [18] addresses optimal sensor selection in structured systems with a perfect matching in $\mathcal{B}(\bar{A})$.

The concept of fixed modes under feedback structural constraints is introduced in [19] and the concept of structurally fixed modes is introduced in [20]. There are necessary and sufficient graph theoretic conditions for checking the existence of SFMs in structured systems [13]. In this paper, optimal selection of a feedback matrix for arbitrary pole placement is our main focus. Next we describe existing work in this area.

Feedback selection for arbitrary pole placement is considered in [4], [5], [15], [21], [6] and [16]. Reference [4] considers sparsest feedback selection for a given structured system $(\bar{A},\bar{B},\bar{C})$. The authors proposed a method for finding the minimum set of feedback edges by determining the minimum number of inputs and outputs, which itself is an NP-hard problem to solve [22]. Reference [5] considers optimal feedback selection when each feedback edge is associated with a cost and proposes an algorithm that gives a sub-optimal solution. However, the approach given there is by solving an NP-hard problem, the multi-commodity network flow problem. In short, the scheme given in [5] is sub-optimal and not polynomial time. Due to this, the algorithm proposed in [4] or [5] is not a polynomial time solution to the feedback selection problem.

Given a structured state matrix $\bar{A}$, finding jointly sparsest $\bar{B}$, $\bar{C}$ and $\bar{K}$ such that the closed-loop system has no SFMs is considered in [21]. Finding a minimum cost input-output set and feedback matrix for a given structured system $(\bar{A},\bar{B},\bar{C})$ such that the resulting closed-loop system has no SFMs is considered in [15], when every input, output and feedback edge are associated with costs. However, because of the NP-hardness of the problem, a special class of systems where the state digraph is irreducible is studied in [15]. Given $(\bar{A},\bar{B},\bar{C})$ and costs corresponding to each input and output, finding a minimum cost input-output set such that the resulting closed-loop system has no SFMs is considered in [16]. Since the problem is NP-hard, the authors of [16] proposed an order optimal polynomial time approximation algorithm.

The papers [6], [7] considered a special case of Problem 1, namely when $\mathcal{B}(\bar{A})$ has a perfect matching and all feedback links are of uniform cost, and dedicated inputs and outputs. Papers [6], [7] claimed the NP-hardness of this case and using that result claimed the NP-hardness for the so-called non-dedicated input-output case. However, for the uniform-cost, dedicated i/o case considered in [6], [7], Problem 1 is not NP-hard: a key 'inference' from a lemma ([6, Lemma 4]) to the main result ([6, Theorem 5]) has a subtle flaw. While our manuscript [23] elaborates on the subtle flaw[3] of [6], [7], where we further show there that the case considered in [6], [7] is not just not NP-hard, but in fact, of linear complexity: we provide only a summary here. Papers [6], [7] consider only a special case of the optimal solution in the reduction from the graph decomposition problem. In a reduction one must ideally show that *any* optimal solution to the given problem gives an optimal solution to the NP-hard or NP-complete problem used for reduction [24, Section 1.5]. This paper differs from our work in [23] as follows.

The case considered in [6], [7] and [23] assumes n dedicated inputs and n dedicated outputs with uniform cost feedback links. Hence the case considered there assumes that all feedback links are feasible. On the other hand, in this paper, we consider feedback links with non-uniform cost and thus some feedback links are allowed to be infeasible[7]. Further, there is no restriction on inputs and outputs to be dedicated. We prove the NP-hardness of this problem using a reduction of the weighted set cover problem. We also prove the inapproximability of the problem.

We summarize our key contributions.

- We prove that Problem 1 is NP-hard (hardness result holds even when $\mathcal{B}(\bar{A})$ has a perfect matching and $\mathcal{D}(\bar{A})$ has only one non-top linked SCC).
- We prove that Problem 1 cannot be approximated in polynomial time to a multiplicative factor $(1-o(1))\log n$, where n denotes the number of states in the system.
- We give a polynomial time algorithm of complexity $O(n^3)$ for solving Problem 1 for a special class of systems whose DAG of SCCs is a line graph and $\mathcal{B}(\bar{A})$ has a perfect matching.
- We give a polynomial time 2-optimal approximation algorithm of complexity $O(n^3)$ for solving Problem 1 for systems whose DAG of SCCs is a line graph and $\mathcal{B}(\bar{A})$ does not have a perfect matching.

Using the details given in this section, now we prove the NP-hardness of Problem 1 in the next section.

[6] In a digraph, a cycle is a directed path whose starting and ending vertices are the same such that there are no node repetitions.

[7] Recall that after Problem 1 statement, we allowed edges to have costs from $\mathbb{R}_+ \cup \{\infty\}$.

Algorithm 1 Pseudo-code for reducing the weighted set cover problem to an instance of Problem 1

Input: Weighted set cover problem with universe $\mathcal{U} = \{1,\ldots,N\}$, sets $\mathcal{P} = \{\mathcal{S}_1,\ldots,\mathcal{S}_r\}$ and weight function w

Output: Structured system $(\bar{A},\bar{B},\bar{C})$ and feedback cost matrix P

1: Define $x_1,\ldots,x_{N+1}$ and $y_1,\ldots,y_r$ and u_1 to be interconnected by the following definition of $\bar{A},\bar{B},\bar{C}$

2: Define a structured system $(\bar{A},\bar{B},\bar{C})$ as follows:

3: $\bar{A}_{ij} \leftarrow \begin{cases} \star, & \text{for } i = j, \text{ for } i \in \{1,\ldots,N\}, \\ \star, & \text{for } i \in \{1,\ldots,N\} \text{ and } j = N+1, \\ 0, & \text{otherwise.} \end{cases}$

4: $\bar{B}_{i1} \leftarrow \begin{cases} \star, & \text{for } i = N+1, \\ 0, & \text{otherwise.} \end{cases}$

5: $\bar{C}_{ij} \leftarrow \begin{cases} \star, & \text{for } j \in \mathcal{S}_i, \\ 0, & \text{otherwise.} \end{cases}$

6: Define feedback cost matrix P as:

7: $P_{1j} \leftarrow w(j), \text{ for } j \in \{1,\ldots,r\}$.

8: Given a solution $\bar{K}$ to Problem 1 on $(\bar{A},\bar{B},\bar{C})$, define: sets selected under $\bar{K}$, $\mathcal{S}(\bar{K}) \leftarrow \{\mathcal{S}_j : \bar{K}_{1j} \neq 0\}$,

9: Weight of the set, $w(\mathcal{S}(\bar{K})) \leftarrow \sum_{\mathcal{S}_i \in \mathcal{S}(\bar{K})} w(i)$.

3. Hardness Results

In this section we prove the hardness of Problem 1 using a well known NP-hard problem, the weighted set cover problem. For the uniform cost case of Problem 1, one can reduce from the set cover problem in the similar lines. Thus Problem 1 is NP-hard even when costs are uniform. In addition to the NP-hardness of the problem, in this paper we also prove the inapproximability of the problem, by showing that the problem cannot be approximated to a multiplicative factor $(1-o(1))\log n$, where n denotes the number of states in the system. The NP-hardness result is obtained by reducing the weighted set cover problem to an instance of Problem 1. We prove that the problem is NP-hard even for the case where the state bipartite graph $\mathcal{B}(\bar{A})$ has a perfect matching and there is only one non-top linked SCC in $\mathcal{D}(\bar{A})$. We first detail the weighted set cover problem, denoted as $(\mathcal{U},\mathcal{P},w)$. Given universe $\mathcal{U} = \{1,2,\cdots,N\}$ of N items, r sets $\mathcal{P} = \{\mathcal{S}_1,\mathcal{S}_2,\cdots,\mathcal{S}_r\}$ with $\mathcal{S}_i \subset \mathcal{U}$ and $\bigcup_{i=1}^{r} \mathcal{S}_i = \mathcal{U}$ and a weight function $w : \mathcal{P} \to \mathbb{R}$, the weighted set cover problem consists of finding a set $\mathcal{S}^\star \subseteq \mathcal{P}$ such that $\cup_{\mathcal{S}_i \in \mathcal{S}^\star} \mathcal{S}_i = \mathcal{U}$ and $\sum_{\mathcal{S}_i \in \mathcal{S}^\star} w(i) \leqslant \sum_{\mathcal{S}_i \in \widetilde{\mathcal{S}}} w(i)$ for any $\widetilde{\mathcal{S}}$ that satisfies $\cup_{\mathcal{S}_i \in \widetilde{\mathcal{S}}} = \mathcal{U}$. In order to prove the hardness of the problem we give a reduction of a general instance of the weighted set cover problem to an instance of Problem 1.

Algorithm 1 pseudo-code description: The pseudo-code showing a polynomial time reduction of the weighted set cover problem to an instance of Problem 1 is presented in Algorithm 1. Consider a general instance of the weighted set cover problem consisting of universe $\mathcal{U}$ with $|\mathcal{U}| = N$, sets $\mathcal{P} = \{\mathcal{S}_1,\ldots,\mathcal{S}_r\}$ and weight w. We construct a structured system $(\bar{A},\bar{B},\bar{C})$ that has states $x_1,\ldots,x_{N+1}$, input u_1 and outputs $y_1,\ldots,y_r$ (Step 1). Notice that in $\mathcal{D}(\bar{A})$ every state has an edge to itself (Step 3). In addition, state x_{N+1} has an edge to all other states. Thus $\mathcal{B}(\bar{A})$ has a perfect matching, $M = \{(x_i', x_i)$ for $i \in \{1,\ldots,N+1\}\}$. Hence, condition b) in Proposition 1 is satisfied. We consider a single input u_1 that connects to state x_{N+1} only (Step 4). Notice that construction of $\bar{C}$ relies on $\mathcal{P}$. Specifically, $\bar{C}_{ij} = \star$ if $x_j \in \mathcal{S}_i$ (Step 5). Finally, the cost matrix P, which gives the costs for feeding outputs y_j's to input u_1, is defined as shown in Step 7. Note that $\bar{K} = \{\bar{K}_{ij} = \star, \text{ for all } i,j\} \in \mathcal{K}_s$. Thus $\mathcal{K}_s$ is non-empty. Now we formulate and prove the following.

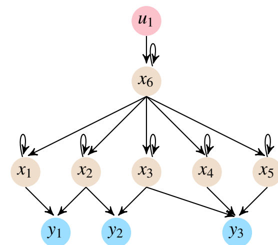

Figure 1: Illustrative example demonstrating construction of $\mathcal{D}(\bar{A},\bar{B},\bar{C})$ given in Algorithm 1 for a weighted set cover problem with $\mathcal{U} = \{1,\ldots,5\}$, $\mathcal{P} = \{\mathcal{S}_1,\mathcal{S}_2,\mathcal{S}_3\}$, where $\mathcal{S}_1 = \{1,2\}$, $\mathcal{S}_2 = \{2,3\}$ and $\mathcal{S}_3 = \{3,4,5\}$.

Lemma 1. *Consider the weighted set cover problem $(\mathcal{U},\mathcal{P},w)$ and a structured system $(\bar{A},\bar{B},\bar{C})$ and feedback cost matrix $P \in \mathbb{R}^{m\times p}$ constructed using Algorithm 1. Let $\mathcal{K}_s$ be the set of feasible solutions to Problem 1. For this structured system, (i) $\bar{K} \in \mathcal{K}_s$ if and only if the sets selected under $\bar{K}$, $\mathcal{S}(\bar{K})$ covers $\mathcal{U}$ and (ii) $w(\mathcal{S}(\bar{K})) = P(\bar{K})$.*

Proof. (i) **Only-if part:** We assume $\bar{K} \in \mathcal{K}_s$ and then show that $\mathcal{S}(\bar{K})$ is a cover. Given $\bar{K}$ is a solution to Problem 1. Thus, $\bar{K}$ satisfies condition a). We need to prove $\cup_{\mathcal{S}_i \in \mathcal{S}(\bar{K})} = \mathcal{U} = \{1,\ldots,N\}$. Instead, suppose there exists an element $j \in \mathcal{U}$ that is not covered by $\mathcal{S}(\bar{K})$. Let $\mathcal{S}(\bar{K})$ consist of sets $\mathcal{S}_{i_1},\ldots,\mathcal{S}_{i_k}$ and the corresponding outputs are $y_{i_1},\ldots,y_{i_k}$. Thus $\bar{K}_{1g} = \star$, for $g \in \{i_1,\ldots,i_k\}$. Since element j is not covered by $\mathcal{S}(\bar{K})$, there does not exist $y \in \{y_{i_1},\ldots,y_{i_k}\}$ that has edge (x_j, y). Thus x_j does not satisfy condition a) in Proposition 1. This contradicts the assumption that $\bar{K}$ is a solution to Problem 1.

(i) **If part:** We assume that $\mathcal{S}(\bar{K})$ is a cover and then show that $\bar{K} \in \mathcal{K}_s$. Suppose not. Since $\mathcal{B}(\bar{A})$ has a perfect matching all state nodes lie in disjoint cycles which consist of only state nodes. Thus condition b) in Proposition 1 is satisfied without using any feedback edge. Thus $\bar{K} \notin \mathcal{K}_s$ implies that there exists a state x_j that does not satisfy condition a). Let $\bar{K} \in \{0,\star\}^{1\times(N+1)}$ have $\star$'s at indices $i_1,\ldots,i_k$. That means outputs $y_{i_1},\ldots,y_{i_k}$ are fed back to input u_1. The corresponding sets are $\mathcal{S}_{i_1},\ldots,\mathcal{S}_{i_k}$. Assume $j \leqslant N$. Since x_j does not satisfy condition a), there does not exist $g \in \{i_1,\ldots,i_k\}$ such that edge (x_j, y_g) is present. Then, there exists no set $\mathcal{S}_g \in \{\mathcal{S}_{i_1},\ldots,\mathcal{S}_{i_k}\}$ such that element j is covered. Now if $j = N+1$, then $\{i_1,\ldots,i_k\} = \phi$. Thus no output is fed back to u_1 and thus $\mathcal{S}(\bar{K}) = \phi$. This contradicts the assumption that $\mathcal{S}(\bar{K})$ is a cover. Thus $\bar{K} \in \mathcal{K}_s$. This completes the proof of statement (i).

Finally, to prove (ii), we note that Step 7 in Algorithm 1 gives $w(\mathcal{S}(\bar{K})) = P(\bar{K})$ and this completes the proof. □

Next we give the NP-hardness result using a reduction of the weighted set cover problem. We show that any instance of the weighted set cover problem can be reduced to an instance of Problem 1 such that an optimal solution to Problem 1 gives an optimal solution to the weighted set cover problem.

Theorem 3. *Consider a weighted set cover problem and let $(\bar{A},\bar{B},\bar{C})$ and P be the structured system and the feedback cost matrix constructed using Algorithm 1 respectively. Let $\bar{K}^\star$ be an optimal solution to Problem 1 and $\mathcal{S}(\bar{K}^\star)$ be the cover corresponding to $\bar{K}^\star$. Then, (i) $\mathcal{S}(\bar{K}^\star)$ is feasible and an optimal solution to the weighted set cover problem, and (ii) Problem 1 is NP-hard.*

Proof. Given a general instance of the weighted set cover problem, we first construct a structured system $(\bar{A},\bar{B},\bar{C})$ and feedback cost matrix P using Algorithm 1. Here we prove that an optimal solution to Problem 1 gives an optimal solution to the weighted set cover problem. Let $\bar{K}$ be a feasible solution to Problem 1. Using Lemma 1 the sets selected under $\bar{K}$, $\mathcal{S}(\bar{K})$ covers $\mathcal{U}$. Hence, $\mathcal{S}(\bar{K})$ is a feasible solution to the weighted set cover problem. For proving optimality, we use a contradiction argument. Let $\bar{K}^\star$ be an optimal solution to Problem 1. From Lemma 1, $\mathcal{S}(\bar{K}^\star)$ covers $\mathcal{U}$ and $P(\bar{K}^\star) = w(\mathcal{S}(\bar{K}^\star))$. Thus $\mathcal{S}(\bar{K}^\star)$ is a feasible solution to the weighted set cover problem.

To prove optimality, we show that $w(\mathcal{S}(\bar{K}^\star)) \leqslant w(\mathcal{S})$ for any $\mathcal{S}$ that satisfies $\cup_{\mathcal{S}_i \in \mathcal{S}} \mathcal{S}_i = \mathcal{U}$. In other words, an optimal solution to Problem 1 gives an optimal solution to the weighted set cover problem. Suppose not. Then there exists a cover $\widetilde{\mathcal{S}}$ such that $w(\widetilde{\mathcal{S}}) < w(\mathcal{S}(\bar{K}^\star))$. Let $\widetilde{\mathcal{S}}$ consist of sets $\{\mathcal{S}_{i_1}, \mathcal{S}_{i_2}, \ldots, \mathcal{S}_{i_k}\}$ and the corresponding outputs are $\{y_{i_1}, y_{i_2}, \ldots, y_{i_k}\}$. Note that there is only one input u_1. Connecting $\{y_{i_1}, y_{i_2}, \ldots, y_{i_k}\}$ to u_1 satisfies condition a) in Proposition 1. This is because for any non-bottom linked SCC, say $\mathscr{B}_k = x_k$, there is some $y \in \{y_{i_1}, y_{i_2}, \ldots, y_{i_k}\}$ connecting x_k. So, $u_1 \to x_{N+1} \to x_k \to y \to u_1$ is a cycle and hence x_{N+1} and x_k belong to the same SCC that has $y \to u_1$ edge. Since $\mathscr{B}_k$ is arbitrary, condition a) holds. So, $\{y_{i_1}, y_{i_2}, \ldots, y_{i_k}\}$ is a feasible solution, and has a cost given by the cost of the set cover. Thus for $\widetilde{\bar{K}} = \{\widetilde{\bar{K}}_{1j} = \star : j \in \{i_1, \ldots, i_k\}\}$, $P(\widetilde{\bar{K}}) < P(\bar{K}^\star)$. This contradicts the assumption that $\bar{K}^\star$ is an optimal solution to Problem 1. This proves that an optimal solution to Problem 1 gives an optimal solution to the weighted set cover problem.

To prove (ii), use Lemma 1 and since any optimal solution to Problem 1 gives an optimal solution to the weighted set cover problem, Problem 1 is NP-hard. □

Remark 1. *Problem 1 is NP-hard even when the feedback links are of uniform costs: use Algorithm 1 with all weights=1, to show the NP-hardness of the problem. This reduction corresponds to minimum set cover problem.*

Note that for the structured system $(\bar{A}, \bar{B}, \bar{C})$ constructed in Algorithm 1, the bipartite graph $\mathcal{B}(\bar{A})$ has a perfect matching. Thus all state nodes lie in a cycle that consists of only x_i's and thus condition b) in Proposition 1 is satisfied. Thus Theorem 3 implies that satisfying condition a) optimally itself is NP-hard even for systems that has a single non-top linked SCC.

The following lemma shows that an approximate solution to Problem 1 on the structured system constructed using Algorithm 1 gives an approximate solution to the weighted set cover problem.

Lemma 2. *Consider the weighted set cover problem and the structured system $(\bar{A}, \bar{B}, \bar{C})$ and cost matrix P constructed in Algorithm 1. For every $\varepsilon \geqslant 1$, if there exists an ε-optimal solution to Problem 1, then there exists an ε-optimal solution to the weighted set cover problem.*

Proof. The proof of this lemma has two parts: (i) we show that an optimal solution $\bar{K}^\star$ to Problem 1 gives an optimal solution $\mathcal{S}(\bar{K}^\star)$ to the weighted set cover problem, and (ii) we show that, if $P(\bar{K}) \leqslant \varepsilon\, p^\star$, then $w(\mathcal{S}(\bar{K})) \leqslant \varepsilon\, w(\mathcal{S}^\star)$.

Note that (i) is proved in Theorem 3. For proving (ii) we use Lemma 1. Given

$$\begin{aligned} p(\bar{K}) &\leqslant \varepsilon p^\star, \\ w(\mathcal{S}(\bar{K})) &\leqslant \varepsilon p^\star = \varepsilon\, w(\mathcal{S}(\bar{K}^\star)) = \varepsilon\, w(\mathcal{S}^\star). \end{aligned}$$

This completes the proof. □

Next we prove that Problem 1 <u>cannot</u> be approximated in polynomial time to a constant factor. The inapproximability result holds even for systems whose state bipartite graph $\mathcal{B}(\bar{A})$ has a perfect matching and $\mathcal{D}(\bar{A})$ has a single non-top linked SCC.

Proof of Theorem 1: The set cover problem cannot be approximated to factor $(1-o(1))\log N$, where N denotes the cardinality of the universe [25]. Thus the weighted set cover problem also cannot be approximated to factor $(1-o(1))\log N$ since set cover is a special case where all weights are non-zero and uniform. However, by Lemma 2, if there exists an approximation algorithm that gives an ε-optimal solution to Problem 1 for a structured system constructed using Algorithm 1, then it gives an ε-optimal solution to the weighted set cover problem. Thus, since weighted set cover cannot be approximated to factor $(1-o(1))\log N$, Problem 1 also cannot be approximated even for this special case. Hence, the inapproximability holds for the general structured systems also. This completes the proof of Theorem 1. □

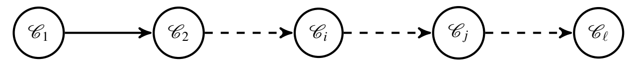

Figure 2: The DAG of SCCs in $\mathcal{D}(\bar{A})$ of a structured system

4. Line Graph Systems

In this section, we consider Problem 1 on two special classes of systems. We first present a polynomial complexity algorithm based on dynamic programming to solve Problem 1 for one class of systems. Then we give a polynomial time 2-optimal approximation algorithm for the second class of systems for solving Problem 1 using the dynamic programming algorithm proposed and a minimum cost perfect matching algorithm. We consider structured systems whose directed acyclic graph (DAG) obtained by condensing SCCs in $\mathcal{D}(\bar{A})$ into nodes is a line graph. In other words, the DAG constructed after condensing SCCs in $\mathcal{D}(\bar{A})$ to super nodes and connecting these super nodes whenever there exists an edge connecting two states in those SCCs is a directed path as shown in Figure 2.

Let $\{\mathscr{C}_1, \ldots, \mathscr{C}_\ell\}$ denote the *ordered* set of SCCs in $\mathcal{D}(\bar{A})$. Note that in this graph there is exactly one non-top linked SCC, $\mathscr{C}_1$, and exactly one non-bottom linked SCC, $\mathscr{C}_\ell$. We further assume that $\mathcal{B}(\bar{A})$ has a perfect matching. Thus condition b) in Proposition 1 is satisfied and hence solving Problem 1 optimally is equivalent to satisfying condition a) optimally. Note that connecting an output y that is connected to $\mathscr{C}_\ell$ to an input u that is connected to $\mathscr{C}_1$ may not be optimal to satisfy condition a) as this connection can be very expensive when compared to the rest of the connections. Further, an optimal solution may consist of connections that cover some of the SCCs multiple times. This can happen if satisfying condition a) is cheaper that way when compared to others.

If the feedback costs are uniform, then Problem 1 is trivial. In that case since $\mathscr{C}_1$ is the only non-top linked SCC and $\mathscr{C}_\ell$ is the only non-bottom linked SCC, connecting an output y that connects to $\mathscr{C}_\ell$ to an input u that connects to $\mathscr{C}_1$ will satisfy condition a). Similarly, if the digraph $\mathcal{D}(\bar{A})$ is irreducible, that is $\mathcal{D}(\bar{A})$ is a single SCC, then too the solution is trivial. In that case connecting (y_j, u_i) where P_{ij} is the smallest entry in the matrix P is optimal. Thus optimal solution $\bar{K}^\star$ for this case has $\star$ only at one location, i.e., $\bar{K}^\star_{ij}$. Figure 2 shows a schematic diagram of the line graph whose vertices are SCCs in $\mathcal{D}(\bar{A})$. We prove that Problem 1 can be solved in polynomial time for this class of systems.

For this section the following assumption holds.

Assumption 1. *The DAG of SCCs in $\mathcal{D}(\bar{A})$ is a line graph.*

We propose a polynomial time algorithm for solving Problem 1 for structured systems when the bipartite graph $\mathcal{B}(\bar{A})$ has a perfect matching and Assumption 1 holds. The proposed algorithm is a dynamic programming algorithm. Since $\mathcal{B}(\bar{A})$ has a perfect matching, the algorithm aims at achieving condition a) in Proposition 1 optimally. The pseudo-code of the proposed scheme is presented in Algorithm 2.

Consider a structured system $(\bar{A}, \bar{B}, \bar{C})$ and cost matrix P. We denote the SCCs in $\mathcal{D}(\bar{A})$ as $\{\mathscr{C}_1, \ldots, \mathscr{C}_\ell\}$. We define U_k as the set of inputs that connect to some states in $\mathscr{C}_k$ (see Step 2). Similarly, we define Y_k as the set of outputs that are connected from some states in $\mathscr{C}_k$ (see Step 3). Now we have the following definition.

Definition 3. *An SCC $\mathscr{C}_k$ is said to be covered if condition a) is satisfied for all states in $\mathscr{C}_k$. In other words, an edge (y_j, u_i) covers $\mathscr{C}_k$ if all the state nodes in $\mathscr{C}_k$ lie in an SCC with edge (y_j, u_i).*

Algorithm 2 Dynamic programming based pseudo-code for solving Problem 1 on structured systems when $\mathcal{B}(\bar{A})$ has a perfect matching and Assumption 1 holds

Input: Structured system $(\bar{A}, \bar{B}, \bar{C})$, feedback cost matrix P
Output: Feedback matrix $\bar{K}^a$

1: $\{\mathscr{C}_1, \ldots, \mathscr{C}_\ell\}$ are the SCCs in $\mathcal{D}(\bar{A})$
2: $U_k \leftarrow \{u_i : \bar{B}_{ri} = \star$ and $x_r \in \mathscr{C}_k\}$
3: $Y_k \leftarrow \{y_j : \bar{C}_{jr} = \star$ and $x_r \in \mathscr{C}_k\}$
4: $\mathscr{U}_k \leftarrow \cup_{i=1}^{k} U_i$
5: $\mathscr{Y}_k \leftarrow \cup_{i=k}^{\ell} Y_i$
6: $W([0]) \leftarrow 0$, $\mathscr{S}_0 \leftarrow \phi$
7: **for** $k = 1, \ldots, \ell$ **do**
8: $\quad W([k]) \leftarrow$ min cost to keep $\{\mathscr{C}_1, \ldots, \mathscr{C}_k\}$ in cycles
9: $\quad A_k \leftarrow \{(y_j, u_i) : y_j \in \mathscr{Y}_k$ and $u_i \in \mathscr{U}_k\}$
10: $\quad t_k(i) \leftarrow \min_q\{u_i \in \mathscr{U}_q : (y_j, u_i) \in A_k\}$
11: $\quad W([k]) \leftarrow \min_{(y_j,u_i) \in A_k}\{P_{ij} + W([t_k(i) - 1])\}$
12: $\quad$ If $W([k]) = P_{vw} + W([z])$, then $\mathscr{S}_k \leftarrow (y_w, u_v) \cup \mathscr{S}_z$, where $v \in \{1, \ldots, m\}$, $w \in \{1, \ldots, p\}$ and $z \in \{1, \ldots, k\}$
13: **end for**
14: $\bar{K}^a \leftarrow \{\bar{K}^a_{ij} = \star : (y_j, u_i) \in \mathscr{S}_\ell\}$

Note that connecting a $u_i \in U_k$ to some $y_j \in Y_k$ covers $\mathscr{C}_k$. However, in addition to these there are other feedback edges that can cover $\mathscr{C}_k$. To characterize all the feedback edges that cover SCC $\mathscr{C}_k$, we define sets $\mathscr{U}_k$ and $\mathscr{Y}_k$. Here $\mathscr{U}_k$ consists of all inputs that are connected to some states in $\mathscr{C}_j$'s for $j \leqslant k$. Similarly, $\mathscr{Y}_k$ consists of all outputs that are connected from some states in $\mathscr{C}_j$'s for $j \geqslant k$. Thus A_k, as defined in equation (2), consists of all edges that cover SCC $\mathscr{C}_k$ (see Step 9 of Algorithm 2). The use of dynamic programming in Algorithm 2 is based on the following lemma.

Lemma 3. *Consider a structured system $(\bar{A}, \bar{B}, \bar{C})$ and cost matrix P given as input to Algorithm 2. Let $\bar{K}^\star$ be an optimal solution to Problem 1 and $\mathscr{S}^\star = \{(y_j, u_i) : \bar{K}^\star_{ij} = \star\}$. Define*

$$A_k := \{(y_j, u_i) : y_j \in \mathscr{Y}_k \text{ and } u_i \in \mathscr{U}_k\}. \quad (2)$$

Then, for every $k \in \{1, \ldots, \ell\}$, $\mathscr{S}^\star \cap A_k \neq \phi$.

Proof. Given $\bar{K}^\star$ is an optimal solution to Problem 1 and $\mathscr{S}^\star$ is the corresponding set of minimum cost feedback edges. Thus edges in $\mathscr{S}^\star$ cover SCCs $\mathscr{C}_1, \ldots, \mathscr{C}_\ell$. The sets $\mathscr{U}_k$ and $\mathscr{Y}_k$ in Algorithm 2 are constructed in such a way that the set A_k consists of all possible feedback edges that can cover SCC $\mathscr{C}_k$. Suppose $\mathscr{S}^\star \cap A_k = \phi$. Then, the edges in $\mathscr{S}^\star$ do not cover $\mathscr{C}_k$. Thus $\bar{K}^\star$ does not satisfy condition a) in Proposition 1. Hence, $\bar{K}^\star \notin \mathcal{K}_s$. This contradicts the assumption that $\bar{K}^\star$ is an optimal solution to Problem 1. Thus for all k, $\mathscr{S}^\star \cap A_k \neq \phi$. ☐

Now for $(y_j, u_i) \in A_k$, $t_k(i)$ is defined as the lowest index q such that $u_i \in \mathscr{U}_q$ (see Step 10). Thus $t_k(i) \leqslant k$. The significance of $t_k(i)$ is that edge $(y_j, u_i) \in A_k$ not only covers SCC $\mathscr{C}_k$, but also covers all the SCCs $\mathscr{C}_{t_k(i)}, \ldots, \mathscr{C}_k$. Hence if (y_j, u_i) is present in the set of edges that cover $\mathscr{C}_1, \ldots, \mathscr{C}_k$, then the rest of the edges need to cover only $\mathscr{C}_1, \ldots, \mathscr{C}_{t_k(i)-1}$. Now $W([k])$ given in Step 11 of the algorithm denotes the minimum cost for covering $\mathscr{C}_1, \ldots, \mathscr{C}_k$ and $\mathscr{S}_k$ denotes the corresponding feedback edges (see Step 12). The dynamic programming step of the algorithm proceeds as follows.

For $k = 1$, we start at SCC $\mathscr{C}_1$. To cover $\mathscr{C}_1$, we will pick an edge in A_1 that is of the least cost. Thus $\mathscr{S}_1$ consists of a single edge which is from A_1. Now we cover $\mathscr{C}_1$, $\mathscr{C}_2$ together. Thus an edge in A_2 will be present. This edge will connect an output $y_j \in \mathscr{Y}_2$ to an input u_i in $\mathscr{U}_2$. Suppose $u_i \in U_2$ and $u_i \notin U_1$. Then edge (y_j, u_i) covers only $\mathscr{C}_2$ and not $\mathscr{C}_1$. Thus the optimal cost to cover $\mathscr{C}_1, \mathscr{C}_2$ is $P_{ij} + W([1])$. Else if $u_i \in U_2$, then SCCs $\mathscr{C}_1, \mathscr{C}_2$ are covered. Then the optimal cost to cover $\mathscr{C}_1, \mathscr{C}_2$ is P_{ij}. Finally, the minimum cost to cover $\mathscr{C}_1, \mathscr{C}_2$ is obtained by finding minimum over all edges in A_2. A generic dynamic programming equation is given in Step 11 of Algorithm 2. $\mathscr{S}_k$ keeps track of the edges required to cover $\mathscr{C}_1, \ldots, \mathscr{C}_k$ with the minimum cost. Every stage of the dynamic programming algorithm is updated using Steps 10 and 11. Now the optimal solution to Problem 1 is obtained using the edges present in $\mathscr{S}_\ell$ as shown in Step 14. For showing the optimality of Algorithm 2, now we prove Theorem 2 (i).

Proof of Theorem 2 (i): We prove (i) using an induction argument. The induction hypothesis is that $W([k])$ is the minimum cost to cover SCCs $\mathscr{C}_1, \ldots, \mathscr{C}_k$ and $\mathscr{S}_k$ is the corresponding optimal set of feedback edges.

Base step: we consider $k = 1$ as the base step. For $k = 1$, $\mathscr{U}_1 = U_1$. Thus $t_k(i) = 1$. Hence, $W([1]) = \min_{(y_j,u_i) \in A_1}\{P_{ij}\}$. Note that here A_1 consists of all possible edges that can result in making all state nodes in $\mathscr{C}_1$ lie in an SCC with a feedback edge. In other words all possible feedback edges that can cover $\mathscr{C}_1$. Thus the algorithm selects an optimal edge in A_1 such that all state nodes in $\mathscr{C}_1$ lie in an SCC with that feedback edge. Suppose (y_j, u_i) is chosen. Then clearly $u_i \in U_1$ and $y_j \in Y_q$ for some $q \geqslant 1$. Thus condition a) is satisfied for all states in $\mathscr{C}_1$ optimally. This proves the base step.

Induction step: for the induction step we assume that the optimal cost to cover SCCs $\mathscr{C}_1, \ldots, \mathscr{C}_{k-1}$ are $W([1]), \ldots, W([k-1])$ respectively. Also, the corresponding edge sets are $\mathscr{S}_1, \ldots, \mathscr{S}_{k-1}$ respectively.

Now we will prove that $W([k])$ is the minimum cost to cover $\mathscr{C}_1, \ldots, \mathscr{C}_k$ and $\mathscr{S}_k$ is the corresponding feedback edge set. Note that A_k consists of all feedback edges that can cover $\mathscr{C}_k$. Thus an edge in A_k has to be used for covering $\mathscr{C}_k$. Let $(y_j, u_i) \in A_k$. Note that (y_j, u_i) not only covers $\mathscr{C}_k$ but also covers $\mathscr{C}_{t_k(i)}, \ldots, \mathscr{C}_k$. Thus the optimal cost to cover $\mathscr{C}_1, \ldots, \mathscr{C}_k$ using (y_j, u_i) is $P_{ij} + W([t_k(i) - 1])$. Notice that $W([k])$ is found after performing a minimization over all edges in A_k. Since $t_k(i) \leqslant k$ and we assumed that the induction hypothesis is true for $\mathscr{C}_1, \ldots, \mathscr{C}_{k-1}$, $W([k])$ is the minimum cost for covering $\mathscr{C}_1, \ldots, \mathscr{C}_k$. Further, $\mathscr{S}_k$ is the union of that edge $(y_j, u_i) \in A_k$ that is selected in the minimization step and $\mathscr{S}_{t_k(i)-1}$. Thus $\mathscr{S}_k$ is the corresponding set of edges of $W([k])$. This completes the proof of Theorem 2 (i). ☐

Next we consider a class of structured systems where only Assumption 1 holds, i.e., the bipartite graph $\mathcal{B}(\bar{A})$ does not have a perfect matching. Since $\mathcal{B}(\bar{A})$ does not have a perfect matching, condition b) in Proposition 1 also has to be satisfied using the feedback connections. In this case, we propose a two stage algorithm. The proposed algorithm uses the dynamic programming algorithm explained above and a minimum cost perfect matching algorithm [14]. The dynamic programming algorithm gives solution $\bar{K}^a$ that satisfies condition a). The minimum cost perfect matching algorithm gives a solution $\bar{K}^b$ that satisfies condition b). We prove that combining these together we get a 2-optimal solution to Problem 1.

The pseudo-code for solving Problem 1 on a structured system where only Assumption 1 holds is presented in Algorithm 3. Firstly, an optimal set of feedback edges that satisfy condition a) in Proposition 1 is obtained using the dynamic programming algorithm given in Algorithm 2. Let $\bar{K}^a$ denote the feedback matrix obtained as solution to the dynamic programming algorithm (see Step 1). Note that this feedback matrix may not guarantee condition b). To satisfy condition b) we run a minimum cost perfect matching algorithm on the bipartite graph $\mathcal{B}(\bar{A}, \bar{B}, \bar{C}, \bar{K}^p)$ with cost function defined as shown in Step 5. Let $M^\star$ be an optimal matching obtained and $\bar{K}^b$ is the feedback matrix selected under $M^\star$ (see Step 8). From Proposition 2, $\bar{K}^b$ satisfies condition b) in Proposition 1. Note that feedback matrix

Algorithm 3 Pseudo-code for solving Problem 1 on structured systems where Assumption 1 holds

Input: Structured system $(\bar{A},\bar{B},\bar{C})$ and feedback cost matrix P
Output: Feedback matrix $\bar{K}^{ab}$

1: Find feedback matrix satisfying condition a) using Algorithm 2, say $\bar{K}^a$
2: Define $\bar{K}^p \leftarrow \{\bar{K}^p_{ij} = \star : P_{ij} \neq \infty\}$
3: Construct the bipartite graph $\mathcal{B}(\bar{A},\bar{B},\bar{C},\bar{K}^p) = ((V_{X'}, V_X \cup V_U \cup V_Y), \mathcal{E}_X \cup \mathcal{E}_U \cup \mathcal{E}_Y \cup \mathcal{E}_{K^p} \cup \mathcal{E}_{\mathbb{U}} \cup \mathcal{E}_{\mathbb{Y}})$ as described in Section 2
4: For $e \in \mathcal{E}_X \cup \mathcal{E}_U \cup \mathcal{E}_Y \cup \mathcal{E}_{K^p} \cup \mathcal{E}_{\mathbb{U}} \cup \mathcal{E}_{\mathbb{Y}}$ define:
5: Cost, $c(e) \leftarrow \begin{cases} P_{ij}, & \text{for } e = (u'_i, y_j) \in \mathcal{E}_{K^p}, \\ 0, & \text{otherwise.} \end{cases}$
6: Find minimum cost perfect matching of $\mathcal{B}(\bar{A},\bar{B},\bar{C},\bar{K}^p)$ under cost c, say $M^\star$
7: Find feedback matrix satisfying condition b) optimally using $M^\star$, say $\bar{K}^b$
8: $\bar{K}^b \leftarrow \{\bar{K}^b_{ij} = \star : (u'_i, y_j) \in M^\star\}$
9: $\bar{K}^{ab} \leftarrow \{\bar{K}^{ab}_{ij} = \star$ if either $\bar{K}^a_{ij} = \star$ or $\bar{K}^b_{ij} = \star\}$

$\bar{K}^{ab}$ obtained by taking element wise union of $\bar{K}^a$ and $\bar{K}^b$ (see Step 9) satisfies both the conditions in Proposition 1 and hence is a feasible solution to Problem 1. We have the following corollary using Steps 5 and 8 of Algorithm 3.

Corollary 1. *Let M be a perfect matching in $\mathcal{B}(\bar{A},\bar{B},\bar{C},\bar{K})$ and $\bar{K}(M) := \{\bar{K}(M)_{ij} = \star : (u'_i, y_j) \in M\}$ be the feedback matrix selected under M. Then, $c(M) = P(\bar{K}(M))$.*

Now we give the proof of Theorem 2 (ii).

Proof of Theorem 2 (ii): For proving (ii), let $\bar{K}^\star$ be an optimal solution to Problem 1 with cost $p^\star$. Thus $\bar{K}^\star$ satisfies both the conditions in Proposition 1. Thus the optimal cost for satisfying each condition individually is atmost $p^\star$. Thus $p^\star \geqslant P(\bar{K}^a)$ and $p^\star \geqslant P(\bar{K}^b)$. Hence, $2p^\star \geqslant P(\bar{K}^a) + P(\bar{K}^b)$. Thus $P(\bar{K}^{ab}) \leqslant 2p^\star$. □

The following theorem gives complexities of the two algorithms proposed in this paper for solving Problem 1.

Theorem 4. *Consider a structured system $(\bar{A},\bar{B},\bar{C})$ with n number of states and cost matrix P. Then, both Algorithm 2 and Algorithm 3 each have complexity $O(n^3)$.*

Proof. Finding the SCCs in $\mathcal{D}(\bar{A})$ has $O(n^2)$ complexity. Let m, p denote the number of inputs and outputs in the structured system. Then each stage of Algorithm 2 has to compute atmost mp number of values and has to find the least value amongst them. Note that $m = O(n)$ and $p = O(n)$. Thus each stage of the algorithm is of complexity $O(n^2)$. The maximum number of iterations required is the number of SCCs in $\mathcal{D}(\bar{A})$ which is atmost n. Thus complexity of Algorithm 2 is $O(n^3)$.

Algorithm 2 has complexity $O(n^3)$ and the minimum cost perfect matching algorithm has complexity $O(n^{2.5})$. Combining both, Algorithm 3 has complexity $O(n^3)$ and this completes the proof. □

Remark 2. *In the DAG of SCCs of $\mathcal{D}(\bar{A})$, if there exists a spanning tree that is a line graph, then all the analysis and results discussed in this paper still hold. Figure 3 shows a schematic diagram of DAG of SCCs of $\mathcal{D}(\bar{A})$ of such a system. In such a case, one needs to look at only that particular spanning tree for solving Problem 1. This gives a generalization of the structured systems that are studied in this paper.*

In the next section we explain the dynamic programming algorithm proposed in the paper using an illustrative example.

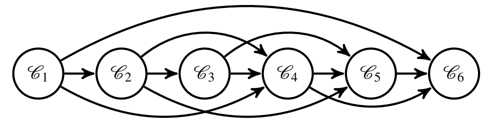

Figure 3: The line graph DAG corresponding to $\mathcal{D}(\bar{A})$

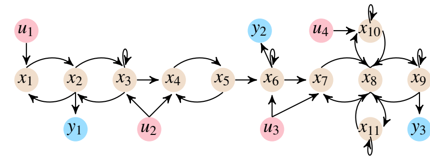

Figure 4: Digraph $\mathcal{D}(\bar{A},\bar{B},\bar{C})$ of the structured system considered for demonstrating Algorithm 2

5. Illustrative Examples

In this section we describe the proposed dynamic algorithm using an example. Figure 4 shows the digraph of a structured system whose cost matrix is given by $P_1 = \begin{bmatrix} 2 & 10 & 100 \\ 7 & 8 & 5 \\ 9 & 5 & 50 \\ 10 & 11 & 13 \end{bmatrix}$.

There are four SCCs each one represents state digraph of systems connected in cascade, for example vehicles in vehicle platoon control: $\mathscr{C}_1 = \{x_1, x_2, x_3\}$, $\mathscr{C}_2 = \{x_4, x_5\}$, $\mathscr{C}_3 = \{x_6\}$ and $\mathscr{C}_4 = \{x_7, x_8, x_9, x_{10}\}$. Also, $\mathcal{B}(\bar{A})$ has a perfect matching. Here $U_1 = \{u_1, u_2\}$, $U_2 = \{u_2\}$, $U_3 = \{u_3\}$, $U_4 = \{u_3, u_4\}$. Similarly, $Y_1 = \{y_1\}$, $Y_2 = \phi$, $Y_3 = \{y_2\}$ and $Y_4 = \{y_3\}$. Subsequently, $\mathscr{U}_1 = \{u_1, u_2\}$, $\mathscr{U}_2 = \{u_1, u_2\}$, $\mathscr{U}_3 = \{u_1, u_2, u_3\}$ and $\mathscr{U}_4 = \{u_1, u_2, u_3, u_4\}$, $\mathscr{Y}_1 = \{y_1, y_2, y_3\}$, $\mathscr{Y}_2 = \{y_2, y_3\}$, $\mathscr{Y}_3 = \{y_2, y_3\}$ and $\mathscr{Y}_4 = \{y_3\}$. Also, $W([0]) = 0$ and $\mathscr{S}_0 = \phi$.

For $k = 1$ our aim is to cover SCC $\mathscr{C}_1$. The inputs and outputs that can achieve this are $\mathscr{U}_1$ and $\mathscr{Y}_1$ respectively. Thus, $W([1]) = \min\{P_{11} + W([0]),\ P_{12} + W([0]),\ P_{13} + W([0]),\ P_{21} + W([0]),\ P_{22} + W([0]), P_{23} + W([0])\} = \min\{2+0, 10+0, 100+0, 7+0, 8+0, 5+0\} = 2$ and $\mathscr{S}_1 = (y_1, u_1)$.

For $k = 2$ our aim is to cover SCC $\mathscr{C}_2$. The inputs and outputs that can achieve this are $\mathscr{U}_2$ and $\mathscr{Y}_2$ respectively. Thus, $W([2]) = \min\{P_{12} + W([0]), P_{13} + W([0]), P_{22} + W([0]), P_{23} + W([0])\} = \min\{10+0, 100+0, 8+0, 5+0\} = 5$ and $\mathscr{S}_2 = (y_3, u_2)$.

For $k = 3$ our aim is to cover SCC $\mathscr{C}_3$. The inputs and outputs that can achieve this are $\mathscr{U}_3$ and $\mathscr{Y}_3$ respectively. Thus, $W([3]) = \min\{P_{12} + W([0]),\ P_{13} + W([0]),\ P_{22} + W([0]),\ P_{23} + W([0]),\ P_{32} + W([2]),\ P_{33} + W([2])\ \} = \min\{10+0, 100+0, 8+0, 5+0, 5+5, 50+5\} = 5$ and $\mathscr{S}_3 = (y_3, u_2)$.

For $k = 4$ our aim is to cover SCC $\mathscr{C}_4$. The inputs and outputs that can achieve this are $\mathscr{U}_4$ and $\mathscr{Y}_4$ respectively. Thus, $W([4]) = \min\{P_{13} + W([0]), P_{23} + W([0]), P_{33} + W([2]), P_{43} + W([3])\} = \min\{100+0, 5+0, 50+5, 13+5\} = 5$ and $\mathscr{S}_4 = (y_3, u_2)$. Thus connecting (y_3, u_2) is an optimal connection in this example. Thus $\bar{K}^a = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \star \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ is an optimal solution to Problem 1.

Figure 5 shows the digraph of a structured system whose cost matrix is given by $P_2 = \begin{bmatrix} 10 & 2 & 500 \\ 199 & 1000 & 25 \\ 200 & 99 & 37 \end{bmatrix}$. We demonstrate Algorithm 3 here using this example. There are four SCCs each one represents state digraphs of systems connected in cascade, for example vehicles in vehicle platoon control: $\mathscr{C}_1 = \{x_1, x_2, x_3\}$, $\mathscr{C}_2 = \{x_4\}$, $\mathscr{C}_3 = \{x_5, x_6, x_7, x_8\}$ and $\mathscr{C}_4 = \{x_9, x_8, x_{10}, x_{11}, x_{12}, x_{13}\}$. Also, $\mathcal{B}(\bar{A})$ does not have a perfect matching. Here $U_1 = \{u_1\}$, $U_2 = \phi$, $U_3 = \{u_2\}$, $U_4 = \{u_3\}$ and $Y_1 = \{y_1\}$, $Y_2 = \{y_2\}$, $Y_3 = \{y_1, y_2, y_3\}$ and $Y_4 = \{y_3\}$. Subsequently, $\mathscr{U}_1 = \{u_1\}$, $\mathscr{U}_2 = \{u_1\}$, $\mathscr{U}_3 = \{u_1, u_2\}$ and $\mathscr{U}_4 = \{u_1, u_2, u_3\}$, $\mathscr{Y}_1 = \{y_1, y_2, y_3\}$, $\mathscr{Y}_2 = \{y_1, y_2, y_3\}$, $\mathscr{Y}_3 = \{y_1, y_2, y_3\}$ and $\mathscr{Y}_4 = \{y_3\}$. Also, $W([0]) = 0$ and $\mathscr{S}_0 = \phi$.

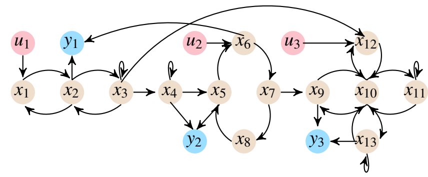

Figure 5: Digraph $\mathcal{D}(\bar{A},\bar{B},\bar{C})$ of the structured system considered for demonstrating Algorithm 3

For $k=1$ our aim is to cover SCC $\mathscr{C}_1$. The inputs and outputs that can achieve this are $\mathscr{U}_1$ and $\mathscr{Y}_1$ respectively. Thus, $W([1]) = \min\{P_{11}+W([0]),\ P_{12}+W([0]),\ P_{13}+W([0])\ \} = \min\{10+0, 2+0, 500+0\} = 2$ and $\mathscr{S}_1 = (y_2,u_1)$.

For $k=2$ our aim is to cover SCC $\mathscr{C}_2$. The inputs and outputs that can achieve this are $\mathscr{U}_2$ and $\mathscr{Y}_2$ respectively. Thus, $W([2]) = \min\{P_{11}+W([0]),\ P_{12}+W([0]),\ P_{13}+W([0])\} = \min\{10+0, 2+0, 8+0, 500+0\} = 2$ and $\mathscr{S}_2 = (y_2,u_1)$.

For $k=3$ our aim is to cover SCC $\mathscr{C}_3$. The inputs and outputs that can achieve this are $\mathscr{U}_3$ and $\mathscr{Y}_3$ respectively. Thus, $W([3]) = \min\{P_{11}+W([0]),\ P_{12}+W([0]),\ P_{13}+W([0]),\ P_{21}+W([2]),\ P_{22}+W([2]),\ P_{23}+W([2])\ \} = \min\{10+0, 2+0, 500+0, 199+2, 1000+2, 25+2\} = 2$ and $\mathscr{S}_3 = (y_2,u_1)$.

For $k=4$ our aim is to cover SCC $\mathscr{C}_4$. The inputs and outputs that can achieve this are $\mathscr{U}_4$ and $\mathscr{Y}_4$ respectively. Thus, $W([4]) = \min\{P_{13}+W([0]), P_{23}+W([2]), P_{33}+W([3]),\} = \min\{500+0, 25+2, 37+2\} = 27$ and $\mathscr{S}_4 = \{(u_2,y_3),(y_2,u_1)\}$. Thus connecting (y_3,u_2) is an optimal connection in this example. Thus $\bar{K}^a = \begin{bmatrix} 0 & \star & 0 \\ 0 & 0 & \star \\ 0 & 0 & 0 \end{bmatrix}$ is an optimal solution to Problem 1.

Now we find a minimum cost perfect matching in $\mathcal{B}(\bar{A},\bar{B},\bar{C},\bar{K})$ under cost matrix P_2. $M^\star = \{(x_1',x_2),\ (x_2',x_1),\ (x_3',x_3),\ (x_4',x_4),\ (x_5',x_8),\ (x_6',x_5),\ (x_7',x_6),\ (x_8',x_7),\ (x_9',x_{10}),\ (x_{10}',x_{12}),\ (x_{11}',x_{11}),\ (x_{12}',u_3),\ (x_{13}',x_{13}),\ (u_1',u_1),\ (u_2',u_2),\ (u_3',y_3),\ (y_1',y_1),\ (y_2',y_2),\ (y_3',x_9)\}$. The corresponding $\bar{K}^b = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \star \end{bmatrix}$. Thus $\bar{K}^{ab} = \begin{bmatrix} 0 & \star & 0 \\ 0 & 0 & \star \\ 0 & 0 & \star \end{bmatrix}$. This completes the discussion of Algorithm 3 using the example given in Figure 5.

6. Conclusion and Future Work

This paper deals with feedback selection of structured systems for arbitrary pole placement when each feedback edge is associated with a cost. Our aim is to optimally select minimum cost feedback matrix such that arbitrary pole placement is possible. We proved that this problem cannot be solved in polynomial time unless P = NP. In this paper we give a reduction of a well studied NP-hard problem, the weighted set cover problem, to an instance of Problem 1. We also show that Problem 1 cannot be approximated in polynomial time to factor $(1-o(1))\log n$, where n denotes the number of states in the system. Due to the NP-hardness and the polynomial time inapproximability of the problem, we considered a special class of systems, where the directed acyclic graph of SCCs of $\mathcal{D}(\bar{A})$ is a line graph and $\mathcal{B}(\bar{A})$ has a perfect matching. We gave a polynomial time optimal algorithm based on dynamic programming for solving Problem 1 on this class of systems. Further, we studied another special class of systems after relaxing the perfect matching assumption, and gave a 2-optimal polynomial time algorithm for solving Problem 1 on this class of systems. Systems consisting of cascade connected irreducible subsystems satisfy the assumed topology. Finding a good approximation algorithm for a general system is a topic of future research.

References

[1] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, “Controllability of complex networks,” *Nature*, vol. 473, no. 7346, pp. 167–173, 2011.

[2] S. Pequito, V. M. Preciado, A.-L. Barabási, and G. J. Pappas, “Trade-offs between driving nodes and time-to-control in complex networks,” *Scientific Reports*, vol. 7, pp. 39 978:1–14, 2017.

[3] K. J. Reinschke, *Multivariable control: a graph theoretic approach*. Springer-Verlag, 1988.

[4] M. E. Sezer, “Minimal essential feedback patterns for pole assignment using dynamic compensation,” in *Proceedings of the IEEE Conference on Decision and Control (CDC)*, vol. 22, San Antonio, USA, 1983, pp. 28–32.

[5] K. Ünyelioğlu and M. E. Sezer, “Optimum feedback patterns in multivariable control systems,” *International Journal of Control*, vol. 49, no. 3, pp. 791–808, 1989.

[6] J. F. Carvalho, S. Pequito, A. P. Aguiar, S. Kar, and G. J. Pappas, “Static output feedback: on essential feasible information patterns,” in *Proceedings of the IEEE Conference on Decision and Control (CDC)*, Osaka, Japan, 2015, pp. 3989–3994.

[7] J. F. Carvalho, S. Pequito, A. P. Aguiar, S. Kar, and G. J. Pappas, “Static output feedback: on essential feasible information patterns,” *ArXiv e-prints*, Sept. 2015. [Online]. Available: http://adsabs.harvard.edu/abs/2015arXiv150902383F

[8] L. Y. Wang, A. Syed, G. G. Yin, A. Pandya, and H. Zhang, “Control of vehicle platoons for highway safety and efficient utility: Consensus with communications and vehicle dynamics,” *Journal of Systems Science and Complexity*, vol. 27, no. 4, pp. 605–631, 2014.

[9] Y. Zheng, S. E. Li, K. Li, F. Borrelli, and J. K. Hedrick, “Distributed model predictive control for heterogeneous vehicle platoons under unidirectional topologies,” *IEEE Transactions on Control Systems Technology*, vol. 25, no. 3, pp. 899–910, 2017.

[10] G. Klančar, D. Matko, and S. Blažič, “A control strategy for platoons of differential drive wheeled mobile robot,” *Robotics and Autonomous Systems*, vol. 59, no. 2, pp. 57–64, 2011.

[11] J. Rodriguez, J.-S. Lai, and F. Z. Peng, “Multilevel inverters: a survey of topologies, controls, and applications,” *IEEE Transactions on Industrial Electronics*, vol. 49, no. 4, pp. 724–738, 2002.

[12] A. Chapman and M. Mesbahi, “On strong structural controllability of networked systems: A constrained matching approach,” in *Proceedings of the IEEE American Control Conference (ACC)*, Washington DC, USA, 2013, pp. 6126–6131.

[13] V. Pichai, M. Sezer, and D. Šiljak, “A graph-theoretic characterization of structurally fixed modes,” *Automatica*, vol. 20, no. 2, pp. 247–250, 1984.

[14] R. Diestel, *Graph Theory*. Springer: New York, 2000.

[15] S. Pequito, S. Kar, and G. J. Pappas, “Minimum cost constrained input-output and control configuration co-design problem: a structural systems approach,” in *Proceedings of the IEEE American Control Conference (ACC)*, Chicago, USA, 2015, pp. 4099–4105.

[16] S. Moothedath, P. Chaporkar, and M. N. Belur, “Approximating Constrained Minimum Cost Input-Output Selection for Generic Arbitrary Pole Placement in Structured Systems,” *ArXiv e-prints*, May 2017. [Online]. Available: http://adsabs.harvard.edu/abs/2017arXiv170509600M

[17] Y.-Y. Liu and A.-L. Barabási, “Control principles of complex systems,” *Reviews of Modern Physics*, vol. 88, no. 3, pp. 035 006:1–58, 2016.

[18] M. Doostmohammadian, H. Zarrabi, and H. R. Rabiee, “Sensor selection cost optimisation for tracking structurally cyclic systems: a p-order solution,” *International Journal of Systems Science*, pp. 1–11, 2017.

[19] S.-H. Wang and E. Davison, “On the stabilization of decentralized control systems,” *IEEE Transactions on Automatic Control*, vol. 18, no. 5, pp. 473–478, 1973.

[20] M. Sezer and D. Šiljak, “Structurally fixed modes,” *Systems & Control Letters*, vol. 1, no. 1, pp. 60–64, 1981.

[21] S. Pequito, S. Kar, and A. P. Aguiar, “A framework for structural input/output and control configuration selection in large-scale systems,” *IEEE Transactions on Automatic Control*, vol. 61, no. 2, pp. 303–318, 2016.

[22] S. Pequito, S. Kar, and A. P. Aguiar, “On the complexity of the constrained input selection problem for structural linear systems,” *Automatica*, vol. 62, pp. 193–199, 2015.

[23] S. Moothedath, P. Chaporkar, and M. N. Belur, “Optimal Feedback Selection for Structurally Cyclic Systems with Dedicated Actuators and Sensors,” *ArXiv e-prints*, June 2017. [Online]. Available: http://adsabs.harvard.edu/abs/2017arXiv170607928M

[24] M. R. Garey and D. S. Johnson, *Computers and Intractability*. W. H. Freeman: New York, 2002, vol. 29.

[25] U. Feige, “A threshold of ln n for approximating set cover,” *Journal of the ACM (JACM)*, vol. 45, no. 4, pp. 634–652, 1998.