Pole assignment problem: a structural investigation

AYLA $\$EFIK$ and M. EROL SEZER$†$

Based on the structure of a closed-loop system under a specified feedback pattern, a qualitative analysis of the problem of pole assignability is considered. The problem is first formulated algebraically, in terms of the relation $p = g(f)$ between the vector $p$ of the closed-loop characteristic polynomial coefficients and the vector $f$ of the non-zero elements of the feedback matrix. Then, translation to the structural framework is achieved by means of two theorems which give graph-theoretical sufficient conditions for solvability. These structural conditions also guarantee genericity of pole assignability.

1. Introduction

In the analysis of dynamical systems, such features as high dimensionality, uncertainty in system parameters, and constraints on information structure often lead to complications which cannot be solved by traditional methods. On the other hand, it may be possible to establish a way out through such problems after gaining sufficient insight into the structure of the system. This need for dealing with system structures is met by a qualitative analysis based on the structure of the system (structural analysis). This type of analysis is concerned with general properties of systems such as controllability, observability, existence of fixed modes, etc., which may also be regarded as the potential system properties. This is consistent with physical reality since system parameter values are never known precisely. The fact that digital computers work with true zeros and fuzzy numbers is another justification for this approach. Investigation of these properties from the genericity point of view is also of interest. A system is said to possess a property generically if that property holds for almost all values of the non-zero system parameters. In other words, if a property of a system is generic, then it fails to hold only in pathological cases when there is an exact matching of system parameters.

It was Lin (1974) who first introduced the concept in his characterization of structural controllability for single-input systems. His result was extended to the multi-input case by Shields and Pearson (1976). Sezer and Šiljak (1981) developed their characterization for structurally fixed modes in the same context.

This paper is concerned with a structural analysis of the problem of pole assignability. Non-zero system parameters are assumed to be algebraically independent and structural modelling based on structured matrices and directed graphs (digraphs) is used for system description. Graph-theoretic formulations due to Reinschke (1984) serve as tools in constructing our main results, which give graphical sufficient conditions for generic pole assignability by constant output feedback.

Section 2 is devoted to the formulation of the problem both algebraically and generically together with the establishment of the framework necessary for our
structural approach. In §3, we state and prove the two main theorems of the paper. Section 4 includes examples of classes of generically pole assignable systems that satisfy these results, thus demonstrating their non-triviality. A search algorithm to check the existence of a constant output feedback matrix which satisfies the conditions of the theorems is given in the Appendix.

2. Problem formulation and preliminaries

2.1. Algebraic formulation of the pole assignment problem

Consider a linear, time-invariant system described as

\[ \mathcal{S}: \begin{cases} \dot{x} = Ax + Bu \\ y = Cx \end{cases} \]  

where \( x(t) \in \mathbb{R}^n \), \( u(t) \in \mathbb{R}^m \) and \( y(t) \in \mathbb{W} \) are the state, input and output of \( \mathcal{S} \), and \( A, B \) and \( C \) are real constant matrices of appropriate dimensions.

Let \( F \) be an \( m \times r \) matrix with \( v \leq mr \) non-zero elements which can be chosen arbitrarily. Applying a constant output feedback

\[ u = Fy \]  

(2)

to \( \mathcal{S} \) in (1) results in a closed-loop system

\[ \mathcal{S}(F): \dot{x} = (A + BFC)x \]  

(3)

having a characteristic polynomial

\[ p(s) = \det (sI - A - BFC) \]

\[ = s^n + p_1 s^{n-1} + \cdots + p_{n-1} s + p_n \]  

(4)

Let the non-zero, arbitrary elements of \( F \) be represented as a point \( f = (f_1, f_2, \ldots, f_v) \) in \( \mathbb{R}^v \), and the coefficients of the characteristic polynomial \( p(s) \) in (4) as a point \( p = (p_1, p_2, \ldots, p_n) \) in \( \mathbb{W} \). Then, the relation between \( p \) and \( f \) can be represented by a smooth mapping \( g: \mathbb{R}^v \rightarrow \mathbb{W} \) as

\[ p = g(f) \]  

(5)

where \( \mathbb{W} \) is a smooth manifold in \( \mathbb{R}^v \). The pole-assignment problem is concerned with the existence of a solution \( f \in \mathbb{R}^v \) of (5) for every given \( p \in \mathbb{W} \).

We observe that \( v \geq n \) is a necessary condition for solvability of (5) for all \( p \), which we assume to hold in the rest of the paper. Let us, then, partition the feedback variables \( f_1, f_2, \ldots, f_v \) into two disjoint subsets of \( f_c \) and \( f_e \), respectively. Fixing the variables in \( f_c \) at particular real values, (5) is reduced to

\[ p = \tilde{g}(f_c) \]  

(6)

where \( \tilde{g}: \mathbb{R}^n \rightarrow \mathbb{R}^v \) is a restriction of \( g \) to \( \mathbb{R}^n \). The following result (Reinschke 1987) gives a sufficient condition for pole assignability.

**Lemma 1**

Assume \( n \leq v \leq mr \). If there exists a partitioning of the feedback variables \( f_1, f_2, \ldots, f_v \) into two disjoint sets \( f_c \) and \( f_e \) containing \( n \) and \( v - n \) elements such that
Pole assignment problem

after appropriately fixing those in \( f_c \) the derivative \( \tilde{g}_{f_c} \) is unimodular, then the system \( \mathcal{S} \) is arbitrarily pole assignable by the feedback \( \mathcal{F} \).

Note that when \( \tilde{g}_{f_c} \) is unimodular, then \( \det \tilde{g}_{f_c} \) is a constant so that \( \tilde{g} \) is a homeomorphism; that is, for every \( p \in \mathbb{R}^n \), there exists a unique \( f_c \in \mathbb{R}^n \) satisfying \( \tilde{g}(f_c) = g(f_c, f_c) = p \).

Our main concern is this paper lies in a qualitative analysis of the pole assignment problem, based on the structure of the pair \( (\mathcal{S}, \mathcal{F}) \). In particular, we aim at deriving a structural counterpart of Lemma 1, and providing graph-theoretic conditions for generic pole assignability. We devote the rest of this section to establishing the framework needed for this approach.

2.2. Structural representation of systems

In this section, we review some basic concepts and results related to structured matrices, genericity (Shields and Pearson 1976), graph theory (Harary et al. 1965), and structural representation of systems (Sage 1977).

Structured matrices and generic properties

Two matrices \( M_1, M_2 \in \mathbb{R}^{n \times q} \) are said to be 'structurally equivalent' if there is one-to-one correspondence between the locations of their non-zero entries. The equivalence class of structurally equivalent matrices in \( \mathbb{R}^{n \times q} \) can be represented by a \( p \times q \) 'structured matrix' \( M \), whose entries are either fixed zeros or algebraically independent parameters in \( \mathbb{R} \). If \( M \) has \( m \) non-zero parameters, then associated with \( M \) we define a parameter space \( \mathbb{R}^m \) such that for every \( d \in \mathbb{R}^m \), \( M(d) \) defines a fixed matrix in the equivalence class that \( M \) represents. A fixed matrix \( M \) is said to be admissible with respect to \( \mathcal{M} \), denoted as \( M \in \mathcal{M} \), if \( M = M(d) \) for some \( d \in \mathbb{R}^m \).

Let \( \Pi \) be a property asserted about the structured matrix \( M \). Then it is a mapping \( \Pi : \mathbb{R}^m \rightarrow \{0, 1\} \) defined as

\[
\Pi(d) = \begin{cases} 1, & \text{if } \Pi \text{ holds for } M(d) \\ 0, & \text{otherwise} \end{cases}
\]

Let \( \Phi(d) \) be a polynomial in \( d = (d_1, ..., d_m) \) with real coefficients. The set

\[
\Gamma = \{ d \in \mathbb{R}^m \mid \Phi(d) = 0 \}
\]

is a 'variety' in \( \mathbb{R}^m \). \( \Gamma \) is said to be proper if \( \Gamma \neq \mathbb{R}^m \) and non-trivial if \( \Gamma \neq \emptyset \). The property \( \Pi \) is said to be 'generic' if there exists a proper variety \( \Gamma \) such that \( \ker \Pi \subset \Gamma \). A generic property holds almost everywhere in \( \mathbb{R}^m \).

The 'generic rank' of a structured matrix \( \mathcal{M} \), denoted by \( \hat{\rho}(\mathcal{M}) \), is the maximal rank \( \mathcal{M}(d) \) can achieve for \( d \in \mathbb{R}^m \). The set \( \{ d \in \mathbb{R}^m \mid \text{rank } \mathcal{M}(d) < \hat{\rho}(\mathcal{M}) \} \) can easily be shown to be a proper variety in \( \mathbb{R}^m \), so that almost all fixed matrices \( \mathcal{M}(d) \) have rank \( \hat{\rho}(\mathcal{M}) \).

Diagraphs

A 'diagraph' is an ordered pair \( \mathcal{D} = (\mathcal{V}, \mathcal{E}) \), where \( \mathcal{V} \) is a finite set of 'vertices' and \( \mathcal{E} \) a set of oriented 'edges'. An edge oriented from \( v_j \in \mathcal{V} \) to \( v_i \in \mathcal{V} \) is denoted by the ordered pair \( (v_j, v_i) \), where \( v_j \) is called the 'tail' and \( v_i \) the 'head' of the edge. If \( (v_j, v_i) \in \mathcal{E} \), then \( v_j \) is said to be 'adjacent' to \( v_i \), and \( v_i \) adjacent from \( v_j \). The
adjacency relation can be described by a square binary matrix, \( R = (r_{ij}) \) such that \( r_{ij} = 1 \) if and only if \((v_j, v_i) \in \mathcal{E}\). A sequence of edges \(\{(v_1, v_2), (v_2, v_3), \ldots, (v_{k-1}, v_k)\}\) where all vertices are distinct is called a 'path' from \(v_1\) to \(v_k\), denoted by \((v_1, v_k)\). In this case, \(v_k\) is said to be 'reachable' from \(v_1\). If \(v_k\) coincides with \(v_1\), then the path is called a 'cycle'. The path that remains after the removal of an edge of a cycle is called the 'complementary path' of that edge relative to the cycle. Any two cycles are said to be 'disjoint' if they have no common vertices. A collection of disjoint cycles is called a 'cycle family'.

A diagraph \( \mathcal{D}_s = (\mathcal{V}_s, \mathcal{E}_s) \), with a vertex set \( \mathcal{V}_s = \{v_0, v_1, \ldots, v_r\} \) and an edge set \( \mathcal{E}_s = \{(v_0, v_1), (v_1, v_2), \ldots, (v_{r-1}, v_r)\} \), is called a 'stem', \(v_0\) and \(v_r\) are the 'origin' and the 'tip' of the stem, respectively. A diagraph \( \mathcal{D}_b = (\mathcal{V}_b, \mathcal{E}_b) \), with \( \mathcal{V} \) as above and \( \mathcal{E}_b = \mathcal{E} \cup \{(v_i, v_j)\} \) is called a 'bud'. \(v_0\) is the origin and \((v_0, v_j)\) is called the 'distinguished edge' of \( \mathcal{D}_b \). A cactus is a diagraph \( \mathcal{D}_c = \mathcal{D}_s \cup \mathcal{D}_{b_1} \cup \mathcal{D}_{b_2} \ldots \cup \mathcal{D}_{b_k} \), where \( \mathcal{D}_s \) is a stem with origin \(v_0\) and tip \(v_r\) and \( \mathcal{D}_{b_i} \) are buds with origins \(v_i \neq v_j \) such that \(v_i\) is the only vertex common to \( \mathcal{D}_s \cup \mathcal{D}_{b_1} \cup \mathcal{D}_{b_2} \ldots \cup \mathcal{D}_{b_{i-1}} \) and \( \mathcal{D}_{b_i} \), \(i = 1, \ldots, k\). Origin \(v_0\) and tip \(v_r\) of \( \mathcal{D}_c \) are also the origin and the tip of \( \mathcal{D}_c \), respectively. If \( \mathcal{D}_c \) above is replaced by a bud, then the diagraph is called a 'precactus', denoted by \( \mathcal{D}_p \). Clearly, by deleting an appropriate edge of a precactus, it can be reduced to a cactus.

In a cactus, every vertex is reachable from the origin through a unique path. If in a cactus \( \mathcal{D}_c = (\mathcal{V}_c, \mathcal{E}_c) \), vertices that are adjacent from the origin \(v_0\) are \(v_1, v_2, \ldots, v_n\), then the sets \( \mathcal{V}_r = \{\mathcal{V} \mid \mathcal{V} \text{ is reachable from } v_1\} \) are disjoint and \( \mathcal{V}_c = \{v_0\} \cup \mathcal{V}_1 \cup \mathcal{V}_2 \cup \ldots \cup \mathcal{V}_q \). Each of the subgraphs of \( \mathcal{D}_c \) defined by one of the vertex sets \( \{v_0\} \cup \mathcal{V}_i \) is called a 'bunch' of the cactus. The bunch that contains the tip of the cactus is called the 'terminal bunch', and the others (if any) 'non-terminal bunches'. Thus a terminal bunch is a cactus itself and a non-terminal bunch is a precactus.

**System structure matrix and system digraph**

Associated with the system \( \mathcal{S} \) of (1), we define a square structured matrix as

\[
S = \begin{bmatrix}
A & B & O \\
O & O & O \\
C & O & O
\end{bmatrix}
\]

which is called the 'system structure matrix'. Viewing the matrix \( S \) as a binary matrix with zero and non-zero elements, we define the digraph \( \mathcal{D} = (\mathcal{V}, \mathcal{E}) \) which assumes \( S \) as its adjacency matrix to be the digraph of the system \( \mathcal{S} \). For convenience, the vertex set of \( \mathcal{D} \) can be partitioned as \( \mathcal{V} = \mathcal{U} \cup \mathcal{I} \cup \mathcal{O} \), where \( \mathcal{U} \), \( \mathcal{I} \) and \( \mathcal{O} \) are the sets of input, state and output variables, respectively. The system digraph \( \mathcal{D} \) completely characterizes the structure of \( \mathcal{S} \). We say that two dynamic systems are structurally equivalent if their digraphs are the same up to an enumeration of \( \mathcal{U} \), \( \mathcal{I} \) and \( \mathcal{O} \).

The subgraph \( \mathcal{D}_{ux} = (\mathcal{I} \cup \mathcal{U}, \mathcal{E}_{ux}) \) obtained by removing from \( \mathcal{D} \) the output vertices and the edges connected to them is called the input-truncated system digraph and corresponds to the system structure matrix

\[
S_{ux} = \begin{bmatrix}
A & B \\
O & O
\end{bmatrix}
\]
When a feedback of the form (2) is applied to $\mathcal{S}$ of (1), the resulting closed loop-system of (3) has the system structure matrix

$$
\mathbf{S}(\mathcal{F}) = \begin{bmatrix}
A & B & 0 \\
O & O & F \\
C & O & 0
\end{bmatrix}
$$

Accordingly, the system digraph becomes $\mathcal{G}(\mathcal{F}) = (\mathcal{V}, \mathcal{E} \cup \mathcal{F}_e)$, where $\mathcal{F}_e = \{(y_j, u_i) \mid f_{ij} \neq 0\}$ is the set of feedback edges.

For convenience, the edges in $\mathcal{E}$ are called the 'd-edges' and those in $\mathcal{F}_e$ the 'f-edges'. Accordingly, a cycle is called an f-cycle if it contains at least one f-edge and a d-cycle otherwise. Similarly, a cycle family is called an f-cycle family if it contains at least one f-edge, a simple f-cycle family if it contains one and only one f-edge, and a d-cycle family otherwise. Note that if a feedback variable $f_{ij}$ is given a fixed non-zero value, then the corresponding f-edge $(y_j, u_i)$ becomes a d-edge as $f_{ij}$ is no more different from a non-zero parameter of $A$, $B$ or $C$.

A system $\mathcal{S}$ is said to be 'structurally controllable' if it is either controllable or structurally equivalent to a controllable system. $\mathcal{S}$ is structurally controllable if and only if the output-truncated system digraph $D_{\mathcal{G}}$ is spanned by a family of disjoint cacti, that is, there exist a family of cacti $\mathcal{D}_{cl} = (\mathcal{V}_{cl}, \mathcal{E}_{cl})$ with $\mathcal{V}_{cl} = \{u_{k_i}\} \cup \mathcal{X}_i$ and $\mathcal{E}_{cl} \subset \mathcal{G}_{ax}$ such that $\cup \mathcal{X}_i = \mathcal{F}$.

### 2.3. Generic pole assignment problem

Imitating the definitions of the well-known structural properties such as structural controllability and existence of structurally fixed modes, we state the following definition.

**Definition 1**

A system $\mathcal{S}$ of (1) is said to be structurally pole assignable by a feedback $\mathcal{F}$ of (2) if there exists a system structurally equivalent to $\mathcal{S}$ which is pole assignable by $\mathcal{F}$. \hfill $\square$

Let us assume, as in an analysis of structural controllability that the non-zero parameters of the system structure matrix $\mathbf{S}$ in (7) are algebraically independent, and correspond to the data point $d \in \mathbb{R}^n$. Then, the relation in (5) can be expressed as

$$
p = g(d, f)
$$

(10)

to indicate the dependence of $g$ on the system parameters. Clearly, structural pole assignable is concerned with the existence of a particular data point $d^* \in \mathbb{R}^n$ for which the equation

$$
p = g(d^*, f) = g^*(f)
$$

(11)

has a solution for every given $p \in \mathbb{R}^n$. We note that solvability of (11) does not readily imply solvability of (10) for almost all $d \in \mathbb{R}^n$. In other words, structural pole assignable is not a generic property, or at least, cannot easily be proved to be a generic property. The reason is that the solvability of the non-linear equation (10) cannot easily be reduced to a condition involving only the parameter vector $d$. 

Pole assignment problem
We do, however, aim at obtaining structural conditions in terms of the system digraph, which guarantee genericity of structural pole assignability. For this purpose we refer to the formulation of Reinschke (1984), which is summarized below.

Consider the closed loop system digraph \( D(F) = (V, E \cup \Xi) \) associated with the system structure matrix \( S(F) \) of (9). By assigning a weight to every edge, \( D(F) \) becomes a weighted digraph. The weight of a \( d \)-edge is the corresponding non-zero parameter value of \( A, B \) or \( C \), and the weight of an \( f \)-edge is the corresponding variable feedback gain. Accordingly, the weight of a path, a cycle or a cycle family is the product of weights of all edges involved. Denoting the number of cycles in a cycle family \( \mathcal{C}F \) by \( s(\mathcal{C}F) \), the weight of \( \mathcal{C}F \) by \( \omega(\mathcal{C}F) \), and defining the width \( \gamma(\mathcal{C}F) \) of \( \mathcal{C}F \) to be the total number of state vertices covered by \( \mathcal{C}F \), Reinschke proved the following.

**Lemma 2**

The coefficients \( p_k = g_k(f) \), \( k = 1, 2, \ldots, n \), of the closed-loop characteristic polynomial are given as

\[
g_k(f) = \sum_{\gamma(\mathcal{C}F) = k} (-1)^{s(\mathcal{C}F)}\omega(\mathcal{C}F)
\]

where the summation is carried over all cycle families of width \( k \).

An immediate application of this lemma is that a feedback variable appears in a coefficient \( p_k \) of the closed-loop characteristic polynomial only if it takes part in a cycle family of width \( k \).

3. **Graphical conditions for generic pole assignability**

We start by considering a special case of Lemma 1.

**Corollary 1**

Let \( f_v \) and \( f_c \) be as defined in Lemma 1, with the feedback variables in \( f_v \) renumbered as \( f_1, f_2, \ldots, f_n \). For a partitioning \( N = 1 \cup (N - 1) \) with \( 1 \neq \emptyset \), of the index set \( N = \{1, 2, \ldots, n\} \), define auxiliary variables \( f_k \) as

\[
f_k = \begin{cases} f_k, & k \in 1 \\ \theta_k f_k + \psi_k, & k \in N - 1 \end{cases}
\]

(13)

where \( \theta_k = \theta_k(d) \) is a non-zero polynomial in \( d \), and \( \psi_k = \psi_k(d, f_\ell) \) is a polynomial in \( f_\ell, \ell \in \mathbb{I} \), with coefficients being polynomials in \( d \). Suppose that the restriction \( \tilde{g} \) of \( g \) in (6) to \( \mathbb{R}^n \) is given by

\[
\tilde{g}_k(d; f_c) = \tilde{g}_k(d; f) = \alpha_k + \sum_{\ell=1}^n e_{k\ell} f_\ell, \quad k = 1, 2, \ldots, n
\]

(14)

where \( \alpha_k = \alpha_k(d) \) and \( e_{k\ell} = e_{k\ell}(d) \). Then, \( \mathcal{S} \) is generically pole assignable by \( \mathcal{F} \) if the coefficient matrix \( E = E(d) = (e_{k\ell}) \) has full generic rank. \( \square \)
Proof

The derivative of $\tilde{g}$ is computed as

$$\tilde{g}_{f_i} = E(d)\Xi(d, f_i)$$

where $\Xi = (\xi_m)$ has elements

$$\xi_m = \begin{cases} 1, & k \in \mathbb{I}, l = k \\ 0, & k \in \mathbb{I}, l \neq k \\ \theta_k, & k \in \mathbb{N} - \mathbb{I}, l = k \\ 0, & k, l \in \mathbb{N} - \mathbb{I}, l \neq k \\ \partial \psi_k / \partial f_i, & k \in \mathbb{N} - \mathbb{I}, l \in \mathbb{I} \end{cases}$$

It follows that $\Xi$ can be permuted into

$$\begin{bmatrix} I & 0 \\ \partial \Psi / \partial f_i & \Theta_{N-1} \end{bmatrix}$$

where $\Theta_{N-1} = \text{diag} \{ \theta_k, k \in \mathbb{N} - \mathbb{I} \}$, and $\partial \Psi / \partial f_i = (\partial \psi_k / \partial f_i), k \in \mathbb{N} - \mathbb{I}, l \in \mathbb{I}$. Thus $\Xi(d, f)$ is generically unimodular, and the proof follows from Lemma 1.

We note that under the conditions of Corollary 1, the mapping $\tilde{g}$ can be decomposed as $\tilde{g} = g \circ h$, where $g : \mathbb{R}^n \to \mathbb{R}^n$ is the affine mapping defined in (14), and $h : \mathbb{R}^n \to \mathbb{R}^n$ is defined in (13), both mappings being homeomorphisms. The significance of Corollary 1 lies in the fact that its assumptions and the full generic rank condition on the matrix $E$ can be characterized, with the help of Lemma 2, in terms of the weighted closed-loop digraph $\mathcal{D} (\mathcal{F})$. This leads us to two main results which we state and prove below.

Theorem 1

Suppose that in $\mathcal{D} (\mathcal{F})$ there exists a choice of $n$ distinct $f$-edges, renumbered conveniently as $f_1, f_2, \ldots, f_n$, which after converting the remaining $f$-edges into $d$-edges by fixing their weights at arbitrary values, satisfy the following conditions:

(i) no two $f$-edges occur in the same cycle;

(ii) all $f$-cycles have a vertex in common;

(iii) for $k = 1, 2, \ldots, n$, there exist particular simple $f$-cycle families of width $k$, denoted by $\mathcal{C} \mathcal{F}_k^*$, such that

(a) $f_k \in \mathcal{C} \mathcal{F}_k^*$; and

(b) any other simple $f$-cycle family of width $k$ which contains an $f$-edge $f_l, l \leq k$, also contains a $d$-edge which appears in no $\mathcal{C} \mathcal{F}_j^*, j \leq k$.

Then $\mathcal{F}$ is generically pole assignable with $\mathcal{F}$. 

Proof

Conditions (i) and (ii) guarantee that every $f$-cycle family is a simple $f$-cycle family so that each product term $\omega (\mathcal{C} \mathcal{F})$ in (12) contains at most one variable weight. In other words, each $g_k$ in (12) is an affine function of $f_1, f_2, \ldots, f_n$ as in (14), so that $\tilde{g}$ has the structure in Corollary 1 with $f_k = f_k, k \in \mathbb{N}$, that is with $\mathbb{I} = \mathbb{N}$. Therefore, it suffices to show that the coefficient matrix $E = (e_{kl})$ in Corollary 1 is generically non-singular. For this, we first note that condition (iii)(a) implies that

\[ e_{kl} = \begin{cases} 1, & k \in \mathbb{I}, l = k \\ 0, & k \in \mathbb{I}, l \neq k \\ \theta_k, & k \in \mathbb{N} - \mathbb{I}, l = k \\ 0, & k, l \in \mathbb{N} - \mathbb{I}, l \neq k \\ \partial \psi_k / \partial f_i, & k \in \mathbb{N} - \mathbb{I}, l \in \mathbb{I} \end{cases} \]
A. Šefik and M. E. Sezer

each $e_{kk}$, $k \in \mathbb{N}$, contains at least one non-zero product corresponding to $\mathcal{F}_k^*$, which we denote by $e_{kk}^*$. We now define $d_n = d$, $E_n(d_n) = E(d)$, and partition $E_n$ as

$$E_n(d_n) = E_{n-1}(d_{n-1}) + \hat{e}_m(d_n) + \hat{e}_m'(d_n)$$

where $\hat{e}_m$ denotes what is left from $e_{mn}$ after separating $e_{mn}^*$ (if there remains any).

For a fixed $I \leq n$, either $f_i$ appears in no cycle family of width $n$, in which case

$$e_{mm}(d_n) = 0$$

or if it does, then by condition (iii)(b), the corresponding product term contains the weight of a $d$-edge, which occurs in no $e_{kk}^*$, $k \leq n$. Let $d_{n-1}$ denote the parameter vector after all parameters corresponding to such $d$-edges are set to zero. Then $E_n(d_{n-1})$ is of the form

$$E_n(d_{n-1}) = \begin{bmatrix} E_{n-1}(d_{n-1}) & e_m(d_{n-1})'s \\ e_m(d_{n-1})'s & e_{mm}(d_{n-1}) + \hat{e}_m(d_n) \\ \end{bmatrix}$$

where $e_{mm}(d_{n-1})$ consists of a single non-zero product term, and each diagonal element $e_{kk}(d_{n-1})$ of $E_{n-1}(d_{n-1})$ still contains the product term $e_{kk}^*(d_{n-1}) = e_{kk}^*(d_n)$, $k = 1, 2, ..., n-1$. Obviously, $E_n(d_n)$ is generically non-singular if $E_n(d_{n-1})$ is. On the other hand, $E_n(d_{n-1})$ is generically non-singular if and only if $E_{n-1}(d_{n-1})$ is. Now, replacing $d_n$ and $E_n(d_n)$ by $d_{n-1}$ and $E_{n-1}(d_{n-1})$ and repeating the argument above, we come to the conclusion that $E_n(d_n)$ is generically non-singular if $E_1(d_1) = e_{m}^*(d)$ is non-zero, which is guaranteed by condition (iii)(a). This completes the proof.

A more general result, which makes full use of Corollary 1 is given by the following.

**Theorem 2**

The result of theorem 1 remains valid if condition (ii) is replaced by the following.

(ii)' To any two $f$-edges $f_p$ and $f_q$ that appear in disjoint cycles there corresponds a unique pair of edges $f_o$ and $d_o$ such that

(a) $d_o$ appears in every cycle of $f_o$, but in no cycle of $f_p$ or $f_q$, and

(b) to any two disjoint cycles $\mathcal{C}_p$ and $\mathcal{C}_q$ of $f_p$ and $f_q$ there corresponds a cycle $\mathcal{C}_o$ of $f_o$ which covers exactly the same state vertices as $\mathcal{C}_p$ and $\mathcal{C}_q$ cover, and vice versa.

**Proof**

The proof is based on the following facts.

**Fact 1**

$\mathcal{D}(\mathcal{F})$ does not contain more than two pairwise disjoint $f$-cycles.
Proof of Fact 1

Suppose that $\mathcal{D}(\mathcal{F})$ contains three pairwise disjoint $f$-cycles formed by the $f$-edges $f_p$, $f_q$ and $f_s$. Let us denote, for convenience, the pair of edges $f_i$ and $d_i$ associated with each pair $(f_i, f_j)$, $i, j = p, q, s$, $i \neq j$, by $f_{ij}$ and $d_{ij}$. Then, condition (ii) implies that $\mathcal{D}(\mathcal{F})$ contains a subgraph which is isomorphic to one of the basic structures shown in Fig. 1. (There are eight possible combinations of different orientations of the edges $f_{ij}$, $i, j = p, q, s$, $i \neq j$, but six of these are essentially the same as one of the other two except for a relabelling of $p, q$ and $s$.) However, each of these subgraphs contradicts condition (i), the one in Fig. 1(a) containing a cycle which includes three $f$-edges, $f_{pq}$, $f_{qp}$ and $f_{qs}$, and the one in Fig. 1(b) containing a cycle which includes two $f$-edges $f_{pq}$ and $f_{qs}$. Therefore, $\mathcal{D}(\mathcal{F})$ cannot contain three disjoint $f$-cycles. It cannot contain four or more pairwise disjoint $f$-cycles either, because this necessarily includes the existence of three pairwise disjoint $f$-cycles. This completes the proof of Fact 1. $\square$

Fact 2

The correspondence between the $(f_i, d_i)$'s and the pairs of $(f_p, f_q)$'s in the statement of condition (ii)' is one-to-one. $\square$

Proof of Fact 2

If $(f_i, d_i)$ corresponds to two distinct pairs $(f_{i'}, f_{q'})$ and $(f_{i''}, f_{q''})$ then either all cycles formed by $f_{i'}$ and $f_{i''}$ or all cycles formed by $f_{q'}$ and $f_{q''}$ should cover the same state vertices. Suppose, without loss of generality, that the former is true and that $p < p'$. Since $f_{i'}$ appears in $\mathcal{F}_p^*$, which is of width $p'$, then so does $f_p$ in some $\mathcal{F}_p$ of width $p$. However, every $d$-edge in $\mathcal{F}_p$ appears either in $\mathcal{F}_p^*$ or in $\mathcal{F}_p^*$, which violates condition (iii)(b). The situation is illustrated in Fig. 2, where $p = 1$, $p' = 2$, $\mathcal{F}_p^* = \{d_1, d_2, f_p\}$, $\mathcal{F}_p^* = \{d_3, d_4, d_5, f_p\}$ and $\mathcal{F}_p = \{d_1, d_4, d_5, f_p\}$. $\square$

Fact 3

Suppose the pair $(f_i, d_i)$ corresponds to the (unique) pair $(f_p, f_q)$. If $f_i$ appears in a product term in some $g_\kappa(f)$ of (12), then so does the product $f_i f_q$, and vice versa. Moreover, all the product terms that contain $f_i$ in any $g_\kappa(f)$ are of the form $e_{ik}(e_{if} + e_{pq} f_q f_q)$, where $e_{ik}$, $e_{if}$, and $e_{pq}$ are polynomials in $d$ with $e_r$ and $e_{pq}$ being the same in all such expressions. $\square$

Proof of Fact 3

Let $\mathcal{C}_1, \mathcal{C}_2, \ldots$, denote all simple $f$-cycles formed by $f_i$; and for each $i$, let $\mathcal{C}_{d_1}, \mathcal{C}_{d_2}, \ldots$, denote all $d$-cycle families which have no vertex in common with $\mathcal{C}_i$. Then, any simple $f$-cycle family containing $f_i$ is of the form $\mathcal{C}_i = \mathcal{C}_{t_i} \cup \mathcal{C}_{d_{ij}}$ for some $i$ and $j$, so that $\omega(\mathcal{C}_i) = \omega(\mathcal{C}_{t_i}) \times \omega(\mathcal{C}_{d_{ij}}) = e_{t_i} f_i e_{d_{ij}}$. By condition (ii)', to every $\mathcal{C}_i$ there correspond disjoint simple $f$-cycles $\mathcal{C}_p$ and $\mathcal{C}_q$ formed by $f_p$ and $f_q$, which are also disjoint from all $\mathcal{C}_{d_{ij}}$. Therefore, they form an $f$-cycle family $\mathcal{C}_{pq} = \mathcal{C}_p \cup \mathcal{C}_q \cup \mathcal{C}_{d_{ij}}$ of the same width as that of $\mathcal{C}_i$, and having the weight $\omega(\mathcal{C}_{pq}) = e_{p_{ij}} \times e_{d_{pq}} f_q f_q \times e_{d_{ij}}$. This shows the existence of the product $f_p f_q$ wherever $f_i$ appears. The converse is also true, and the proof of the first part is
complete. Now, let \( e \) be the product of the weights of the \( d \)-edges which are common to all \( C_{ri} \), but do not occur in some \( V_{pi} \cup V_{qi} \). Obviously, \( d_i \) appears in \( e \), so that \( e_i = e_i' \times e_p \). Also define \( e_p \) and \( e_q \) to be the products of the weights of the \( d \)-edges which are common to all \( V_{pi} \) and \( V_{qi} \), respectively, and which do not appear in some \( C_{ri} \), and therefore write \( e_p = e_p' \times e_p \) and \( e_q = e_q' \times e_q \). Since for fixed \( i \), \( V_{pi} \cup V_{qi} \) and \( V_{ri} \) cover exactly the same state vertices, then \( e_p \) and \( e_q \) may

Figure 1. The two basic structures mentioned in the proof of Fact 1.
only contain weights of $d$-edges that are adjacent either from the input or to the output associated with $f_p$ and $f_q$, respectively. Furthermore, $e'_i = e'_p \times e'_q$. Then, $\omega(\mathcal{C}_P) + \omega(\mathcal{C}_{Pq}) = e'_i e_{d0} (e_r f_r + e_p e_q f_p f_q)$ independent of the widths of the cycle families $\mathcal{C}_P$ and $\mathcal{C}_{Pq}$, and the proof follows. \hfill \Box

Now, returning to the proof of Theorem 2, Fact 1 together with condition (i) imply that each product term $\omega(\mathcal{C}_P)$ in (12) contains at most two variable weights. Also, defining

$$0 = \{k \mid f_k \text{ forms a cycle which is disjoint from some other } f\text{-cycle}\}$$

and $f_r$ as in (13) with $\theta_r = e$, and $\psi_r = e_p f_p f_q$, Fact 3 guarantees the structure in (12). The rest of the proof is the same as that of Theorem 1. \hfill \Box

The usefulness of Theorems 1 and 2 depends largely on the choice of the $n$ feedback gains to the included in $f_r$, as well as on the choice of the zero or non-zero fixed values to be assigned to the remaining feedback gains in $f_c$. An algorithm, which determines whether such a choice of $n$ feedback edges that satisfy the conditions of Theorem 2 exists, is given in the Appendix.

4. Examples of generically pole assignable systems

In this section we show that certain classes of systems which are known to be generically pole assignable by state or dynamic output feedback satisfy the conditions of Theorem 2 and thus demonstrate that Theorem 2 characterizes a non-trivial class of pole assignable structures.

4.1. Structurally controllable systems with state feedback

Consider a system described by

$$\mathcal{S} : \dot{x} = Ax + Bu$$

(16)
and the full state feedback law

\[ F: \quad u = Fx \]  

(17)

where \( x \in \mathbb{R}^n \) and \( u \in \mathbb{R}^m \). Since \( F \) is a special case of static output feedback with states considered as outputs, the resulting closed-loop system \( \mathcal{F}(F) \) can be represented by the reduced system structure matrix

\[ S(F) = \begin{bmatrix} A & B \\ F & 0 \end{bmatrix} \]  

(18)

Let the corresponding open- and closed-loop system digraphs be \( \mathcal{D}_{ux} = (\mathcal{I} \cup \mathcal{U}, \mathcal{E}_{ux}) \) and \( \mathcal{D}_{ux}(F) = (\mathcal{I} \cup \mathcal{U}, \mathcal{E}_{ux} \cup \mathcal{E}_F) \). We now state our main result concerning \( \mathcal{F}(F) \).

**Theorem 3**

The following are equivalent:

(a) \( \mathcal{F} \) is structurally controllable;

(b) \( \mathcal{F}(F) \) is generically pole-assignable;

(c) there exists a choice of \( n \) feedback edges such that when the remaining feedback edges are assigned suitable fixed weights, \( \mathcal{D}_{ux}(F) \) satisfies the conditions of Theorem 1.

The proof of Theorem 3 is based on the following two lemmas.

**Lemma 3**

Let \( \mathcal{D}_c = (\mathcal{I} \cup \{u\}, \mathcal{E}) \) be a cactus. Then there exists an enumeration of the state vertices such that

(a) if \( x_i \) is on a non-terminal bunch and \( x_j \) is on the terminal bunch, then \( i < j \);

(b) if \( (x_i, x_j) \in \mathcal{E} \) and \( x_j \) is not the tail of the distinguished edge of some bud, then \( j = i + k + 1 \), where \( k \) is the total number of state vertices on the precactus with origin \( x_i \).\[ \square \]

**Proof**

Using a modified depth-first search algorithm (Tarjan 1972), scan first the non-terminal bunches (if there are any) in any order, and last the terminal bunch of \( \mathcal{D}_c \), and assign the integers 1, 2, ..., \( n \) to the state vertices during the scanning process according to the following simple recursive scheme. Let the current vertex being visited be \( x_i \). If there is a cactus or precactus with origin at \( x_i \), then replace \( \mathcal{D}_c \) by this cactus or precactus (with \( x_i \) taking the role of \( u \)) and repeat. Otherwise, let the unique vertex adjacent from \( x_i \) be \( x^* \). If \( x^* \) is not yet assigned an integer, let \( i = i + 1 \), \( x_i = x^* \), and repeat. Otherwise, \( x^* \) should be adjacent from the root of the cactus currently being scanned. Continue with another bunch. \[ \square \]

It is obvious that this scanning of \( \mathcal{D}_c \) results in an enumeration of the state vertices which satisfies the requirements. To illustrate the scheme, enumeration of the vertices of a simple cactus is shown in Fig. 3.
Lemma 4

Let $S = (A, B)$ be structurally controllable. Then there exists a fixed feedback matrix $F_1$ and a column $b_i$ of $B$ such that

(a) the non-zero elements of $A + BF_1$ and $b_i$ are algebraically independent, and

(b) the system $S_1 = (A + BF_1, b_i)$ is structurally controllable.

Proof

If $(A, b_i)$ is structurally controllable for some $i$, let $F_1 = 0$. Otherwise, let $D_{ax}$ be spanned by a union of cacti $D_{e_1}, D_{e_2}, ..., D_{e_k}$ with roots $u_{i_1}, u_{i_2}, ..., u_{i_k}$ and tips $x_{n_1}, x_{n_1 + n_2}, ..., x_{n_1 + ... + n_k}$, where $1 \leq k \leq m$, $1 \leq i_1 < ... < i_k \leq m$, and $n_1 + n_2 + ... + n_k = n$. Let $F_1 = (f_{pq})$ with

$$f_{pq} = \begin{cases} 1, & \text{if } p = i, q = n_1 + ... + n_{l-1}, \text{ for some } 2 \leq l \leq k \\ 0, & \text{otherwise} \end{cases}$$

and let $i = i_l$. Then, since the elements of $(A, B)$ are algebraically independent and non-zero elements of $F_1$ are fixed as unity, the elements of $(A + BF_1, b_i)$ are also algebraically independent. Moreover, $S_1$ is spanned by a cactus obtained by making the roots of $D_{e_l}$ coincide with $x_{n_1 + ... + n_l}$, $l = 1, ..., k - 1$. \hfill \square

Note that Lemma 4 is a structural counterpart of the well-known algebraic result (Davison and Wang 1973) that if $(A, B)$ is controllable, then for almost all $F_1$, $(A + BF_1, b_i)$ are controllable.
We now prove Theorem 3.

Proof of Theorem 3

Owing to Lemma 4, it suffices to give the proof for the single-input case.

(a) \iff (b): Obvious.

(c) \implies (b): Theorem 1.

(a) \implies (c): Let the system digraph \( D_{as} \) be spanned by a cactus \( D_c \), whose state vertices are enumerated as in Lemma 3. Let the feedback edges be enumerated in the same way so that \( f_i = (x_i, u) \), \( i = 1, 2, ..., n \). Since all \( f \)-cycles in \( D_{as}(F) \) pass through vertex \( u \), conditions (i) and (ii) of Theorem 1 are readily satisfied. The enumeration of the state vertices guarantees that for \( i = 1, 2, ..., n \), any state vertex \( x_j \) with \( j \leq i \) either lies on the complementary path of \( f_i \) in \( D_c(F) \), and hence belongs to the \( f \)-cycle defined by \( f_i \), or belongs to a \( d \)-cycle in \( D_c(\mathcal{F}) \) which has no vertex in common with the complementary path of \( f_i \). Let \( \mathcal{F}_i \) denote the union of these cycles in \( D(\mathcal{F}) \). Obviously, \( \mathcal{F}_i \) is a simple \( f \)-cycle family of width \( i \) which contains \( f_i \). (For example, referring to Fig. 3, \( \mathcal{F}_2 \) consists of the \( f \)-cycle \( \{(x_4, x_5), (x_5, x_4)\} \) and the \( d \)-cycles \( \{(x_2, x_2)\} \) and \( \{(x_4, x_5), (x_5, x_4)\} \).

This proves condition (iii)(a) of Theorem 1. Now, let \( \mathcal{F}_i \) be any simple \( f \)-cycle family of width \( i \) which includes an \( f \)-edge \( f_j \) for some \( j < i \). If \( \mathcal{F}_i \) contains a \( d \)-edge which does not belong to the edge set of \( D_c \), then this edge does not appear in any \( \mathcal{F}_i \), and condition (iii)(b) of Theorem 1 is readily satisfied for \( \mathcal{F}_i \).

Suppose all the \( d \)-edges of \( \mathcal{F}_i \) belong to \( D_c \). Since \( \mathcal{F}_i \) covers exactly \( i \) vertices, it covers a vertex \( x_k \) with \( k \geq i \). Then, the edge originating from \( x_k \) in \( \mathcal{F}_i \) is a \( d \)-edge (the only \( f \)-edge in \( \mathcal{F}_i \) is \( f_j \) which originates from \( x_j \)) which does not appear in any \( \mathcal{F}_l \), \( l \leq k \). Again, (iii)(b) is satisfied. This completes the proof. \( \square \)

4.2. A class of structurally controllable and observable systems with dynamic output feedback

Consider a single-input/single-output system

\[
\mathcal{P}: \begin{align*}
\dot{x} &= Ax + bu \\
y &= c^Tx
\end{align*}
\]

(19)

to be controlled by a dynamic output feedback of the form

\[
\mathcal{P}': \begin{align*}
\dot{x} &= \hat{A}\dot{x} + \hat{b}y \\
u &= \hat{c}^T\dot{x} + \hat{f}y
\end{align*}
\]

(20)

where \( \hat{x} \in \mathbb{R}^d \) is the state of the controller \( \mathcal{P}' \). It is well known (Davison and Chatterjee 1971) that the closed-loop system consisting of \( \mathcal{P} \) and \( \mathcal{P}' \) is the same as the one obtained by applying a constant output feedback of the form

\[
\mathcal{F}_a: \begin{bmatrix} u \\ u_c \end{bmatrix} = \begin{bmatrix} \hat{f} & \hat{c}^T \\ \hat{b} & \hat{A} \end{bmatrix} \begin{bmatrix} y \\ y_c \end{bmatrix}
\]

(21)
Pole assignment problem to an augmented system

\[ \mathcal{S}_a : \begin{bmatrix} \dot{x} \\ \dot{\hat{x}} \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \hat{x} \end{bmatrix} + \begin{bmatrix} b \\ 0 \end{bmatrix} u \begin{bmatrix} y \\ y_c \end{bmatrix} = \begin{bmatrix} c^T \\ 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{\hat{x}} \end{bmatrix} \]

(22)

Thus the pole assignment problem by dynamic output feedback is essentially the same as pole assignment by constant output feedback, and hence can be attacked with the graph-theoretic approach of § 3.

We assume that \( \mathcal{S} \) is structurally controllable and observable, that is, it has no structurally fixed modes. Let \( \mathcal{S}(f) \) be the digraph of the closed-loop system consisting of \( \mathcal{S} \) and the (scalar) constant output feedback

\[ \mathcal{F} : u = fy \]

Then, \( \mathcal{S} \) having no structurally fixed modes is equivalent to the following two conditions (Pichai et al. 1984):

(a) \( \mathcal{S}(f) \) contains a cycle family \( \mathcal{CF} \) of width \( n \);

(b) \( \mathcal{S}(f) \) is strongly connected, i.e. each state vertex reaches every other either in \( \mathcal{S} \), or through the feedback edge \((y, u)\).

We further assume that each cycle in \( \mathcal{CF} \) has a vertex in common with some input–output path in \( \mathcal{S} \). This is a crucial assumption that enables us to define the auxiliary variables \( \tilde{f}_c \) in (13) using simple polynomials \( \psi_k \) as will become clear in the following development.

We now choose the order of the controller \( \mathcal{S} \) to be \( \tilde{n} = n - 1 \), and fix its structure as

\[ \tilde{A} = \begin{bmatrix} 0 & 0 & \ldots & 0 & \hat{a}_{n-1} \\ 1 & 0 & \ldots & 0 & \hat{a}_{n-2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 1 & \ldots & 0 & \hat{a}_{n-3} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \ldots & 1 & \hat{a}_1 \end{bmatrix}, \quad \tilde{b} = \begin{bmatrix} \tilde{b}_{n-1} \\ \tilde{b}_{n-2} \\ \vdots \\ \hat{b}_1 \end{bmatrix} \]

(23)

\[ \tilde{c}^T = [0 \ 0 \ \ldots \ 0 \ 1], \quad \tilde{f} = f \]

where \( \hat{a}_i, \tilde{b}_i, i = 1, 2, \ldots, n - 1 \), and \( \tilde{f} \) are variable feedback gains. Thus, of the \( n^2 \) elements of \( \mathcal{F}_a \) in (21), \( n^2 - (2n - 1) \) are fixed at 0 or 1 with the remaining \( 2n - 1 \) left as variable parameters.

With \( \mathcal{S} \) chosen as above, the closed-loop digraph \( \mathcal{D}_a(\mathcal{F}_a) \) which corresponds to the system \( \mathcal{S}_a(\mathcal{F}_a) \) has the structure shown in Fig. 4.

We now prove the following result about the pole assignability of \( \mathcal{S}_a(\mathcal{F}_a) \).

Theorem 4

Suppose that \( \mathcal{S}(f) \) contains a cycle family of width \( n \), each cycle of which has a vertex in common with some input–output path in \( \mathcal{S} \). Then \( \mathcal{D}_a(\mathcal{F}_a) \) satisfies the conditions of Theorem 2 with \( n \) replaced by \( n_a = 2n - 1 \).
Figure 4. Illustration of the closed-loop system digraph $\mathcal{G}_r(Q_0)$. 

\[ \begin{align*} 
\hat{a}_{n-1} & \quad \cdots \quad \hat{a}_2 \quad \hat{a}_1 \quad \hat{a}_0 \quad \hat{a}_1 \quad \hat{a}_2 \quad \cdots \quad \hat{a}_{n-1} \\
\hat{x}_{n-1} & \quad \cdots \quad \hat{x}_2 \quad \hat{x}_1 \quad \hat{x}_0 \quad \hat{x}_1 \quad \hat{x}_2 \quad \cdots \quad \hat{x}_{n-1} \\
\hat{y}_{n-1} & \quad \cdots \quad \hat{y}_2 \quad \hat{y}_1 \quad \hat{y}_0 \quad \hat{y}_1 \quad \hat{y}_2 \quad \cdots \quad \hat{y}_{n-1} \\
\cdots & \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \\
\hat{d}_{n-1} & \quad \cdots \quad \hat{d}_2 \quad \hat{d}_1 \quad \hat{d}_0 \quad \hat{d}_1 \quad \hat{d}_2 \quad \cdots \quad \hat{d}_{n-1} \\
\end{align*} \]
Proof

Referring to Fig. 4, we first note that $\mathcal{D}(\mathcal{F}_a) = (\mathcal{V}, \mathcal{E})$, where $\mathcal{D} = (\mathcal{V}, \mathcal{E})$ is the digraph of $\mathcal{F}_a$, and $\mathcal{E}_f$ is the digraph associated with the fixed parameters of $\mathcal{D}$, and $\mathcal{E}_f$ is the set of (variable) $f$-edges corresponding to the feedback parameters $\bar{a}_i, \bar{b}_i, i = 1, 2, ..., n - 1$, and $\bar{f}$. Thus $\mathcal{D}_a(\mathcal{F}_a)$ has $n_a = 2n - 1$ state vertices, which is exactly the same as the number of $f$-edges. We show that these $f$-edges can be suitably ordered so as to satisfy the conditions of Theorem 2.

We first observe that $f$-cycles in $\mathcal{D}_a(\mathcal{F}_a)$ are of one of the following forms:

$$
\mathcal{C}_i = \{(y, u), (u, y)\}
$$

$$
\mathcal{C}_{i_1} = \{(\hat{y}_1, \hat{u}_1), (\hat{u}_1, \hat{y}_1)\}
$$

$$
\mathcal{C}_{i_2} = \{(y, \hat{u}_i), (\hat{u}_i, \hat{y}_i), (\hat{y}_i, u, (u, y))\}
$$

where $(u, y)$ denotes a path in $\mathcal{D}$ and $(\hat{u}_i, \hat{y}_i)$ denotes a path in $\mathcal{D}$. Consequently, no $f$-cycle in $\mathcal{D}_a(\mathcal{F}_a)$ contains more than one $f$-edge, satisfying condition (i) of Theorem 2. Also, only $f_1 = (y, u)$ and an $f$-edge $f_{i_1} = (\hat{y}_1, \hat{u}_1), i = 1, 2, ..., n - 1$, can appear in disjoint $f$-cycles (of forms $\mathcal{C}_i$ and $\mathcal{C}_{i_1}$, respectively). It is not difficult to see that for every such pair $(f_1, f_{i_1})$, $d_e = (\hat{y}_1, u)$ and $f_{i_1} = (y, \hat{u}_1)$ form a unique pair which satisfies condition (ii) of Theorem 2.

To continue the proof we need the following result.

Fact 4

$\mathcal{D}(f)$ has a subgraph $\mathcal{D}$ with the following properties:

(a) $\mathcal{D}$ contains a unique cycle family $\mathcal{C}_F$ of width $n$.

(b) Each cycle in $\mathcal{C}_F$ has a vertex in common with some input–output path.

(c) $\mathcal{D}$ is minimal in the sense that removal of any edge violates (a) or (b) above.

Proof of Fact 4

Pick an arbitrary cycle family $\mathcal{C}_F$ of width $n$ in $\mathcal{D}(f)$, and a minimal set $\mathcal{D}_a$ of additional $d$-edges such that each cycle in $\mathcal{C}_F$ has a vertex in common with some input–output path in the subgraph $\mathcal{D}$ formed by $\mathcal{C}_F$ and these additional $d$-edges. Include $f$ into $\mathcal{D}$, if not already included. If $\mathcal{D}$ contains another cycle family $\mathcal{C}_F$ of width $n$, then one of the cycles in $\mathcal{C}_F$ contains a $d$-edge which is not included in $\mathcal{C}_F$. The subgraph $\mathcal{D}$ of $\mathcal{D}$ obtained by removing this particular $d$-edge still contains a cycle family of width $n$ each cycle of which has a vertex in common with some input–output path. Replace $\mathcal{D}$ by $\mathcal{D}$. $\mathcal{C}_F$ by $\mathcal{C}_F$, and repeat the same argument. Each time by deleting a $d$-edge from $\mathcal{D}$ and modifying $\mathcal{C}_F$, we eventually obtain a subgraph which satisfies properties (a) and (b). Finally, removing some $d$-edges from $\mathcal{D}_a$ if not needed for (b), minimality of $\mathcal{D}$ with respect to the properties (a) and (b) is guaranteed. This completes the proof.

We note that $\mathcal{D}$ in Fact 4 may or may not contain the $f$-edge $(y, u)$. We continue with the proof of Theorem 3 by considering the two cases separately.
Case I: $\mathcal{D}$ does not include the $f$-edge $(y, u)$

In this case, $\mathcal{G}$ is a $d$-cycle family of width $n$. Let $\mathcal{D}_a(\mathcal{F}_a)$ be the digraph obtained from $\mathcal{D}_a(\mathcal{F}_a)$ by replacing $\mathcal{D}$ with $\mathcal{D}$. Since $\mathcal{D}_u(\mathcal{F}_u)$ is obtained from $\mathcal{D}_a(\mathcal{F}_a)$ by removing some $d$-edges of $\mathcal{D}$, it suffices to complete the proof for $\mathcal{D}_a(\mathcal{F}_a)$, because $\mathcal{D}_u(\mathcal{F}_u)$ still satisfies conditions (i) and (ii) of Theorem 2, and if it also satisfies condition (iii), then so does $\mathcal{D}_a(\mathcal{F}_a)$.

Let $\{\mathcal{P}_1, \mathcal{P}_2, ..., \mathcal{P}_s\}$ be a family of input–output paths in $\mathcal{D}$ such that any $d$-cycle in $\mathcal{G}$ has a vertex in common with some $\mathcal{P}_l$. Define $\mathcal{G}_j$ to be the subfamily of all cycles in $\mathcal{G}$ which has no vertex in common with any $\mathcal{P}_l$, $l > j$.

The definition of $\mathcal{D}$ and $\mathcal{G}$ is illustrated in Fig. 5 for a simple digraph $\mathcal{D}$. Note that $\mathcal{G}_0 = 0$, $\mathcal{G}_1 = \mathcal{G}$, and $\mathcal{G}_{j-1} \subset \mathcal{G}_j$, $j = 1, 2, ..., s$. We further define the integers $\alpha_j$ and $\beta_j$ as the number of state vertices in $\mathcal{D}$ and $\mathcal{G}$, i.e. $\alpha_j = \gamma(\mathcal{D}_j)$ and $\beta_j = \gamma(\mathcal{G}_j)$, $j = 1, 2, ..., s$, and let $\alpha_{s+1} = \beta_{s-1} = \beta_0 = 0$ for convenience. It is easy to see that $\alpha_j$ and $\beta_j$ satisfy

(a) $1 \leq \beta_1 < \beta_2 < ... < \beta_s = n$

(b) $\alpha_j + \beta_{j-1} \leq \alpha_{j+1} + \beta_j - 1$, $1 \leq j \leq s$

We partition the integers $\{1, 2, ..., n_s = 2n - 1\}$ into two groups at $s$ levels as shown in Table 1, where Group $A/Level 0$ is empty if $\alpha_1 = 1$, and Group $A/Level s$ is empty if $\alpha_s + \beta_{s-1} = n$.

<table>
<thead>
<tr>
<th>Level</th>
<th>Group A</th>
<th>Group B</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1, ..., $\alpha_1 - 1$</td>
<td>$\alpha_1, ..., \alpha_1 + \beta_1 - 1$</td>
</tr>
<tr>
<td>1</td>
<td>$\alpha_1 + \beta_1, ..., \alpha_2 + 2\beta_1 - 1$</td>
<td>$\alpha_2 + 2\beta_1, ..., \alpha_2 + \beta_1 + \beta_2 - 1$</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>p</td>
<td>$\alpha_p + \beta_{p-1} + \beta_p, ..., \alpha_{p+1} + 2\beta_p - 1$</td>
<td>$\alpha_{p+1} + 2\beta_p, ..., \alpha_{p+1} + \beta_p + \beta_{p+1} - 1$</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>s-1</td>
<td>$\alpha_{s-1} + \beta_{s-2} + \beta_{s-1}, ..., \alpha_s + 2\beta_{s-1} - 1$</td>
<td>$\alpha_s + 2\beta_{s-1}, ..., \alpha_s + \beta_{s-1} + \beta_s - 1$</td>
</tr>
<tr>
<td>s</td>
<td>$\alpha_s + \beta_{s-1} + \beta_s, ..., 2n - 1$</td>
<td></td>
</tr>
</tbody>
</table>

Table 1.

We now define the feedback edges $f_k$ and the associated cycle families $\mathcal{G}_k^*$, $1 \leq k \leq 2n - 1$, for $\mathcal{D}_u(\mathcal{F}_u)$ as follows.

(a) If $k \in \text{Group } A/\text{Level } p$, i.e. if

$$\alpha_p + \beta_{p-1} + \beta_p \leq \alpha_{p+1} + 2\beta_p - 1$$

then

$$f_k = (\hat{y}_1, \hat{u}_{k-\beta_p})$$

$$\mathcal{G}_k^* = \{(\hat{u}_{k-\beta_p}, \hat{y}_1), (\hat{y}_1, \hat{u}_{k-\beta_p})\} \cup \mathcal{G}_p$$

(b) If $k \in \text{Group } B/\text{Level } p$, i.e. if

$$\alpha_{p+1} + 2\beta_p \leq k \leq \alpha_{p+1} + \beta_p + \beta_{p+1} - 1$$
Figure 5. Definition of $\mathcal{F}_1$ and $\mathcal{F}_2$.

\[ \mathcal{CF}_0 = \emptyset \]
\[ \mathcal{CF}_1 = \{C_2\} \]
\[ \mathcal{CF}_2 = \mathcal{CF} = \{C_1, C_2, C_3, C_4\} \]

Note that in case (a)

\[ \gamma(\mathcal{CF}_k) = \gamma((\hat{u}_{k-\beta_p}, \hat{y}_1)) + \gamma(\mathcal{CF}_p) \]
\[ = (k - \beta_p) + \beta_p = k \]

and in case (b)

\[ \gamma(\mathcal{CF}_k) = \gamma((\hat{u}_{k-\beta_p}, \hat{y}_1)) + \gamma(\mathcal{CF}_{p+1}) + \gamma(\mathcal{CF}_p) \]
\[ = (k - \alpha_{p+1} - \beta_p) + \alpha_{p+1} + \beta_p = k \]

so that $\mathcal{CF}_k$ is an $f$-cycle family of width $k$ in $\mathcal{F}_k(\mathcal{D})$. By definition, it includes $f_k$ and no other $f$-edge, satisfying condition (iii)(a) of Theorem 2. Finally, to prove condition (iii)(b), let $\mathcal{F}_g$ be a simple $f$-cycle family of width $k$, which includes some $f$-edge $f_i$ with $l \leq k$. We consider all possibilities for $k$ and $l$, as follows.

1. $k \in \text{Group A or B/Level } p$, $l \in \text{Group A/Level } q$, $q \leq p$. In this case, $\mathcal{CF}_k = \{(\hat{u}_{l-\beta_q}, \hat{y}_1), (\hat{y}_1, \hat{u}_{l-\beta_q})\} \cup \mathcal{CF}_d$, where $\mathcal{CF}_d$ is a $d$-cycle family in $\mathcal{D}$. If $\mathcal{CF}_d$ contains a $d$-edge which does not belong to $\mathcal{CF}_p$ (remember that $k \in \text{Level } p$), then that $d$-edge does not belong to any $\mathcal{CF}_r$, $r \leq p$, either. Since any $j \leq k$ is at some level $r \leq p$, and $\mathcal{CF}_g$ includes $\mathcal{CF}_r$, this particular $d$-edge appears in no $\mathcal{CF}_j$, $j \leq k$, and condition (iii)(b) is satisfied. If $\mathcal{CF}_d$ does not contain such a $d$-edge, then minimality of $\mathcal{D}$ implies $\mathcal{CF}_d \subset \mathcal{CF}_p$. Then, $\gamma(\mathcal{CF}_d) \leq \beta_p$, and we must have
can be shown, as in case I, to lead to a contradiction that

contradicting the assumption that

Again, if \( \text{GF}_d \) contains a d-edge which does not occur in \( \text{GF}_p \), or if \( t \geq p + 2 \), then condition (iii)(b) is satisfied. On the other hand, if \( t > p \), then \( \tilde{G} \), contains a d-edge which has no vertex in common with any \( \text{GF}_r \), \( r < p \), and again condition (iii)(b) is satisfied. The only remaining possibility is the case when \( \text{GF}_d \subset \text{GF}_p \) and \( t \leq p \). This case, however, can be shown, as in case 1, to lead to a contradiction that \( \gamma(\tilde{G}_k) < k \).

(2) \( k \in \text{Group } A/\text{Level } p, l \in \text{Group } B/\text{Level } q, q < p \). In this case, \( \tilde{G}_k = \{ (\tilde{u}_{t-a_{t-1}-a_t}, \tilde{w}_t), (\tilde{w}_t, y), (y, \tilde{w}_{t-a_{t-1}-a_t}) \} \cup \text{GF}_d \), where \( \tilde{G} \) is some input-output path in \( \tilde{G} \), and \( \text{GF}_d \) is some d-cycle family in \( \tilde{G} \). As in case 1, if \( \text{GF}_d \) contains a d-edge which does not occur in \( \text{GF}_p \), then condition (iii)(b) is satisfied. However, it is impossible because \( \gamma(\tilde{G}_k) \leq l - \beta_q + \beta_p \), with equality holding only if \( \text{GF}_d = \text{GF}_p \). This, however, is

impossible because

\[(a) \text{ if } q = p, \text{ then either } \gamma(\tilde{G}_k) < l \leq k \text{ or } \gamma(\tilde{G}_k) \leq l < k \text{ (because } \gamma(\tilde{G}_k) = l = k \text{ can occur only if } \text{GF}_k = \text{GF}_k \);\]

\[(b) \text{ if } q < p, \text{ then } l - \beta_q + \beta_p \leq a_{t-1} + \beta_q + \beta_p - 1 < a_{t-1} + \beta_p + \beta_p - 1 \leq k - 1,\]

contradicting the assumption that \( \gamma(\tilde{G}_k) = k \).

(3) \( k \in \text{Group } B/\text{Level } p, l \in \text{Group } B/\text{Level } q, q \leq p \). \( \tilde{G}_k \) is as in case 2. Again, if \( \text{GF}_d \) contains a d-edge which does not occur in \( \text{GF}_p \), or if \( t \geq p + 2 \), then condition (iii)(b) is satisfied. Otherwise, \( \text{GF}_d = \text{GF}_k \) and \( t \leq p + 1 \), then \( \gamma(\tilde{G}_k) < k \) unless \( l = k \) and \( \text{GF}_d = \text{GF}_p \), in which case \( \tilde{G}_k = \text{GF}_k \), and if \( t \leq p \), then again \( \gamma(\tilde{G}_k) < k \), both contradicting the assumption on \( \text{GF}_k \).

As a result, condition (iii)(b) is also satisfied, and the proof is complete for Case 1.

Case II: The f-edge \((y, u)\) is included in \( \tilde{G} \)

In this case, \( \tilde{G} \) is an f-cycle family of width \( n \), which includes the f-cycle \( \{ (u, y), (y, u) \} \) being some input–output path in \( \tilde{G} \). Let the family of the remaining d-cycles of \( \tilde{G} \) be denoted by \( \tilde{G}' \). Let \( \{ J_1, J_2, \ldots, J_s \} \), \( J_s \neq \tilde{G} \), be a minimal family of input–output paths in \( \tilde{G} \) such that any d-cycle in \( \tilde{G} \) has a vertex in common with some \( J_s \) and let \( J_{s+1} = \tilde{G} \). We now define subfamilies of \( \tilde{G}_j, 1 \leq j \leq s \), the same way as \( \tilde{G}'_j \)s are defined in Case I, but with respect to \( \tilde{G} \) and \( \{ J_j \} \), rather than \( \tilde{G} \) and \( \{ J_j \} \), and similarly define integers \( a_s \) and \( \beta_j, 1 \leq j \leq s \), in terms of \( \tilde{G} \) and \( \tilde{G} \). With these definitions, the proof follows the same lines as the proof of Case I, except that the integers at Level s are modified as shown in Table 2, where, obviously, \( a_{s+1} + \beta_s = n \). This completes the proof of Theorem 4.

Now several remarks are in order. The first remark is about our assumption that each cycle in the cycle family \( \tilde{G} \) of width \( n \) has a vertex in common with some input–output path. Obviously, this assumption is not essential for structural pole assignability of \( \tilde{G} \) using a dynamic output feedback controller \( \tilde{G} \). However, it is needed for proving generic pole assignability using Theorem 2, and it has been observed by Şefik (1989) that it might be possible to remove this assumption by

<table>
<thead>
<tr>
<th>T</th>
<th>Level</th>
<th>Group A</th>
<th>Group B</th>
</tr>
</thead>
<tbody>
<tr>
<td>s</td>
<td>( a_s + \beta_{s-1} + \beta_s + \beta_{s+1} + \beta_s + \beta_s - 1 )</td>
<td>( a_{s+1} + \beta_s + \beta_{s+1} + \beta_{s+2} + \ldots + 2n - 1 ).</td>
<td></td>
</tr>
</tbody>
</table>
modifying Theorem 2 to include more general cases when $\psi_k$ of (13) contains linear terms in addition to a single quadratic term. The second remark is concerned with the restriction of $\mathcal{S}$ to be a single-input/single-output system. Provided Theorem 2 is modified to remove the assumption mentioned above, this restriction can easily be relaxed. One way of doing this is to employ a preliminary constant output feedback to reduce the system to a single-input/single-output system without destroying structural controllability and observability, and then design $\hat{\mathcal{S}}$. A more efficient way, which also allows generic pole assignment using a smaller order dynamic compensator is to imitate the well-known results due to Brasch and Pearson (1970) and Ahmari and Vacroux (1973) in a structural setting. This, however, requires a structural interpretation of controllability and observability indices of $\mathcal{S}$, which is not a straightforward task.

5. Conclusion

Brasch and Pearson’s (1970) renowned result which states that all the poles of a controllable and observable system can be assigned arbitrarily using a dynamic feedback compensator of order $L = \min \{L_c, L_o\} - 1$ where $L_c$ and $L_o$ are the controllability and observability indices of the system, is overly sufficient in most cases. This can be explained by the fact that their algebraic criterion does not take into account the structure of the system, which actually plays the most important role in the solvability of the problem.

In this paper we realize a qualitative investigation of the problem of arbitrary pole assignability, depending merely on the structural information about the system under consideration.

Our results detect a class of system structures for which constant output feedback is sufficient to place all the poles at desired locations. Many examples can be found for which, according to Brasch and Pearson’s (1970) or Kimura’s (1978) results, dynamic compensation is needed for pole assignment, while by our results constant output feedback is sufficient for the job.

Note that it may be possible to broaden the class of systems pole assignable by Theorem 2 by modifying the theorem somehow to include more general cases specified by Corollary 1. On the other hand, Corollary 1 is still a special case of some other result, namely Lemma 1, which probably contains hints on how to characterize a larger class of pole assignable structures.

Appendix

The choice algorithm

In this section we present an algorithm to check the existence of a set of $n$ $f$-edges $f_1, f_2, ..., f_n$ in $\mathcal{D}(\mathcal{F})$ which satisfies conditions of Theorem 2, and to identify one such set if there exists any. The algorithm accepts as input

11: $n$, the number of state vertices in $\mathcal{D}(\mathcal{F})$;
12: $f = (f_1, f_2, ..., f_v)$ a set of all $f$-edges, $v \geq n$;
13: for each $1 \leq k \leq n$, a list of all $f$-cycle families $\{\mathcal{C}_F \}$ of width $k$, each $\mathcal{C}_F$ being specified as a product of the parametric weights of all the edges appearing in $\mathcal{C}_F$;

and produces as output
O1: a positive or negative response regarding the existence of a required set of $f$-edges, and if the response is positive;

O2: the chosen subset $f_j = (f_1^*, f_2^*, ..., f_n^*)$ of $f$ (here we use a starred notation for the variable $f$-edges to distinguish between the orderings of $f$ and $f_j$);

O3: $\mathcal{C}_k^n$, the list of particular simple $f$-cycle families defined by $f_k^n$, $1 \leq k \leq n$;

O4: the fixed values (0 or 1) assigned to the $f$-edges in $f_c = f - f_c$.

The algorithm tries to construct an arborescence (a directed tree) $\mathcal{T} = (\mathcal{V}, \mathcal{E}, \mathcal{F})$ having a longest path of length $2n$ by examining all cycle families $\mathcal{C}_k$, $k = 1, 2, ..., n; s = 1, 2, ..., n_k^*$; and all $f$-edges $f_k^n$, $t = 1, 2, ..., n_k^*$, appearing in each $\mathcal{C}_k$. It halts with a positive response as soon as such a longest path is constructed, and with a negative response if no such path can be formed. The vertices of $\mathcal{T}$ are arranged in $n + 1$ levels, each of which, except level 0, is further divided into two sublevels. The vertices at the first sublevels constitute $\mathcal{V}_c$, and are called the $c$-vertices, while the vertices at the second sublevels constitute $\mathcal{V}_f$, and are called the $f$-vertices. Each $c$-vertex at level $k$ is the son of some $f$-vertex at level $k - 1$, and corresponds to an $f$-cycle family $\mathcal{C}_k$, of width $k$, while each $f$-vertex at level $k$ is the son of some $c$-vertex $\mathcal{C}_k$, at the same level, and corresponds to an $f$-edge that occurs in $\mathcal{C}_k$, $1 \leq k \leq n$. Level 0 contains a single $f$-vertex, denoted by $f_0^*$, which is the root of $\mathcal{T}$. The algorithm proceeds as follows.

Suppose that a path $\mathcal{P}_{k-1}$ of length $2(k - 1)$ is constructed from $f_k^*$ to some $f$ vertex $f_{k-1}^*$ at level $(k - 1)$, with some $f$-edges of $\mathcal{D}(\mathcal{T})$ assigned to the branches and $f$-vertices on $\mathcal{P}_{k-1}$ are assigned to the $f$ vertices of $\mathcal{P}_{k-1}$. If no such $\mathcal{C}_k$ exists, terminate the path $\mathcal{P}_{k-1}$, and search for an unexplored $f$-vertex at level $(k - 1)$ to replace $f_{k-1}^*$. If there exist one or more such cycle families, construct a $c$-vertex for each of them and extend a branch from $f_{k-1}^*$ to these $c$-vertices. Pick any one of these $c$-vertices, say $\mathcal{C}_k^s$, $s = 1, 2, ..., n_k^*$, and label it as $\mathcal{C}_k^*$.

Corresponding to each $f$-edge that occurs in $\mathcal{C}_k^* = \mathcal{C}_k^s$ construct an $f$-vertex, $f_k^n$, $t = 1, 2, ..., n_k^s$, extend a branch from $\mathcal{C}_k^*$ to each $f_k^n$, and assign all other $f$-edges in $\mathcal{C}_k^*$ to the branch $(\mathcal{C}_k^*, f_k^n)$ of $\mathcal{T}$. Pick one of the $f$-vertices, say $f_k^n$, and check if the assignment $f_k^n = f_k^n$ violates the conditions of Theorem 2. If not, set $f_k^n = f_k^n$, and repeat the whole procedure with $k - 1, f_{k-1}^*$, and $\mathcal{P}_{k-1}$ replaced by $k, f_k^n$, and $\mathcal{P}_k$. If the assignment $f_k^n = f_k^n$ violates the conditions of Theorem 2, terminate the path from $f_k^n$ to $f_k^n$, and pick another unexplored $f$-vertex to replace $f_k^n$. If none of $f_k^n$ can be chosen as $f_k^n$, go back to the upper sublevel to replace $\mathcal{C}_k^*$ with another unexplored $c$-vertex $\mathcal{C}_k$. If all the paths through all $\mathcal{C}_k$ are terminated, search for an unexplored $f$-vertex at level $(k - 1)$ to replace $f_{k-1}^*$. In checking whether the assignment $f_k^n = f_k^n$ violates the conditions of Theorem 2, the $f$-edges of $\mathcal{D}(\mathcal{T})$ that are assigned to any branch of $\mathcal{P}_k$ are assumed to be fixed at some non-zero value (at 1, for convenience), and all $f$-edges other than these and $f_j^*, 1 \leq j \leq k$ can be fixed or variable, as appropriate.

With this introduction, we now state the choice algorithm to identify $f_c = (f_1^*, f_2^*, ..., f_n^*)$, where we adopt the following notation:

- $k$: index to scan the levels of $\mathcal{T}$, $0 \leq k \leq n$
- $n_k^*$: number of distinct $f$-cycle families of width $k$ in $\mathcal{D}(\mathcal{T})$
- $s_k$: index to scan the $c$-vertices of $\mathcal{T}$ at level $k$, $0 \leq s_k \leq n_k^*$
- $\mathcal{C}_k^*$: the $c$-vertex chosen at level $k$
The algorithm

Step 1. Set $k \leftarrow 1$, and construct vertex $f_k^*$ of $\mathcal{F}$.

Step 2. Add the c-vertices $\mathcal{CF}_{k-1}$, and the branches $(f_{k-1}^*, \mathcal{CF}_{k-1})$ to $\mathcal{F}$, $1 \leq s \leq n_k^*$, and set $s_k \leftarrow 0$.

Step 3. Set $s_k \leftarrow s_k + 1$. If $s_k \leq n_k^*$ go to Step 5.

Step 4. Set $k \leftarrow k - 1$. If $k = 0$, stop. No choice of $f_k$ is possible. Otherwise, go to Step 7.

Step 5. If $\mathcal{CF}_{k-1}$ contains an f-edge corresponding to an f-vertex $f_k^*$ of $\mathcal{F}$, $i$, $j \leq k - 1$, terminate the path from $f_k^*$ to $\mathcal{CF}_{k-1}$, and go to Step 3. Otherwise, let $\mathcal{CF}_k = \mathcal{CF}_{k-1}$.

Step 6. Add the f-vertices $f_{kt}$ and the branches $(\mathcal{CF}_k, f_{kt})$ to $\mathcal{F}$, $1 \leq t \leq n_n^*$, and set $t_k \leftarrow 0$.

Step 7. Set $t_k \leftarrow t_k + 1$. If $t_k > n_n^*$, go to Step 3.

Step 8. If $k_{kt}$ is assigned to any branch $(\mathcal{CF}_k, f_k^*)$ of $\mathcal{F}$, $1 \leq j \leq k - 1$, terminate the path from $f_k^*$ to $k_{kt}$, and go to Step 7. Otherwise, assign all the f-edges in $\mathcal{CF}_k$, except $k_{kt}$, to the branch $(\mathcal{CF}_k, f_{kt})$.

Step 9. Delete all f-edges of $\mathcal{D}(\mathcal{F})$ except $k_{kt}$, those that correspond to the f-vertices $f_k^*$, $1 \leq j \leq k - 1$, and those that are assigned to the branches $(\mathcal{CF}_k, f_k^*)$, $1 \leq j \leq k - 1$. Convert all f-edges that are assigned to the branches $(\mathcal{CF}_k, f_k^*)$ to d-edges by choosing their weights to be unity. If $f_k^*, f_{kt}^*, ..., f_{k-1}^*$ and $k_{kt}$ do not satisfy the conditions of Theorem 1 for the remaining subgraph and with $n$ replaced by $k$, terminate the path from $f_k^*$ to $k_{kt}$ and go to Step 7. Otherwise, let $f_k^* = k_{kt}$.

Step 10. If $k < n$, set $k \leftarrow k + 1$, and go to Step 2.

Step 11. Let $f_k = (f_k^*, f_{kt}^*, ..., f_{k-1}^*)$. Convert all the remaining f-edges of $\mathcal{D}(\mathcal{F})$ into d-edges by fixing their weights to 1 if they are assigned to some branch $(\mathcal{CF}_k, f_k^*)$ of $\mathcal{F}$, $1 \leq k \leq n$, and to 0 otherwise. Stop.

Example 1

Consider the digraph $\mathcal{D}(\mathcal{F})$ of Fig. 6, in which units weight is assigned to any d-edge adjacent from an input or to an output vertex. In this example, this causes no loss of generality as every input or output vertex has a unique edge adjacent from or to itself. We want to identify an $f_k = (f_k^*, f_{kt}^*, f_{k-1}^*)$, if any exists. We have $n = 4$ and $f = (f_1, f_2, f_3, f_4, f_5, f_6)$, and the list of all f-cycle families $\{\mathcal{CF}_k\}$ of width $k$, $1 \leq k \leq n$, is given in Table 3.

Figure 7 shows the arborescence constructed during the application of the choice algorithm. The algorithm stops with a positive response identifying $f_k = (f_6, f_3, f_1, f_2)$ while fixing $f_4 = f_5 = 0$. The corresponding choice of particular
Figure 6. $\mathcal{D}(\mathcal{F})$ of Example 1.

Figure 7. Arborescence generated by the chosen algorithm in Example 1.
The usefulness of our choice algorithm: in a classical approach, in order to place all the poles of a system possessing a digraph $\mathcal{D}(\mathcal{F})$, as given in Fig. 6, at desired locations, one would attempt to use a dynamic output compensator, whereas we showed above that constant output feedback was sufficient for the job.

<table>
<thead>
<tr>
<th>$k$</th>
<th>$\mathcal{C}_k$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>${f_6}$</td>
</tr>
<tr>
<td>2</td>
<td>${d_5, f_3}$</td>
</tr>
<tr>
<td>3</td>
<td>${d_5, f_3}$</td>
</tr>
<tr>
<td>4</td>
<td>${d_5, f_3}$</td>
</tr>
</tbody>
</table>

Table 3. $f$-Cycle families in $\mathcal{D}(\mathcal{F})$ of Fig. 6.

This example illustrates the significance of Theorem 1 and Theorem 2 and hence the usefulness of our choice algorithm: in a classical approach, in order to place all the poles of a system possessing a digraph $\mathcal{D}(\mathcal{F})$, as given in Fig. 6, at desired locations, one would attempt to use a dynamic output compensator, whereas we showed above that constant output feedback was sufficient for the job.

REFERENCES


