

# Structural Identification: The Computation of the Generic McMillan Degree

N. Karcanias E. Sagianos E. Milonidis

Control Engineering Research Centre, School of Engineering and Mathematics , City University  
Northampton Sq. , London EC1V 0HB, UK

**Abstract**—The McMillan degree of a transfer function model is one of the most important structural characteristics of a system. In this paper the problem of identifying the generic McMillan degree of a rational matrix is considered. The transfer function matrices of interest are those referred to as *Structured Transfer Function (STF)* matrices and have certain elements fixed to zero, some elements being constant and other elements expressing some identified dominant dynamics of the system. For the family of STF matrices the problem of determining the generic McMillan degree is considered using genericity arguments and an optimisation procedure based on path properties of nonnegative integer matrices. A novel approach is introduced that exploits the structure of integer matrices and this leads to an efficient new algorithm for computation of the generic value of the McMillan degree. Links are made to standard problems of optimisation and in particular to the *optimal assignment problem*. The problem examined here belongs to the general area of *Structural Identification* where the evaluation of structural characteristics of STF models is under investigation with robust computational methods. Such problems are of interest to large scale system studies.

## I. INTRODUCTION

The study of system properties based on ill-defined models is a topic of great interest especially in the context of early design of large scale systems, such as process systems. The main interest is to deploy the structural characteristics to predict the true system properties which are defined on ill-defined models [9]. The problems may be tackled in both the frequency domain and the state-space domain. The properties which are dependent on the structure will be referred to as the structural or generic system properties. The structural properties are important in that they are generically possessed by all the systems which may have different parametric values but share the same underlying graph structure [10], [4]; therefore the study of the structural properties is relevant not only to one particular system but to a class of systems.

The subject of our investigation is the study of McMillan degree on special types of transfer functions referred to as *Structural Transfer Functions (STF)* such models are large dimension transfer functions with certain elements fixed to zero, some elements being constant, and other elements expressing the dominant dynamics of the system, which have been identified by some preliminary modelling effort. Structural transfer function matrices frequently arise as models in the early process design stages [3] and they are usually of very large dimensions. Standard methods for computing structural characteristics of these models are not appropriate and new efficient techniques are required which exploit the fact that numerical values of the constants are

not fixed, but only the nature of dominant dynamics. The current paper is focused on the McMillan degree structural characteristic, but the problem area includes other forms of structural characteristics such as infinite pole-zero structure, minimal indices, etc; such issues however are not considered here.

It is a fact that in the early design stages the values of the parameters in the elements of the transfer function are not known exactly. Yet it is desirable to have some knowledge on the McMillan degree since it indicates the complexity of the system. Given the structure of the transfer function matrix and the type of the non-zero entries of the matrix, the evaluation of the McMillan degree of such systems will be termed as generic evaluation of the McMillan degree, and the McMillan degree will be termed as generic McMillan degree of the given structured uncertain model. The McMillan degree of a rational matrix can be calculated from the orders of the denominators of the matrix in Smith-McMillan form [5]. So algorithms can be designed to first transform the rational matrix into Smith-McMillan form by using unimodular transformations and then find the sum of the orders of the denominators. As pointed out in [8], this method is impractical in terms of computations to obtain the Smith-McMillan form. An alternative has been suggested [8], that is to obtain the pole polynomial as the least common multiple of the minors of all orders. The order of this common multiple gives the McMillan degree. This may also be used for computation of the generic form of the Smith-McMillan form of the given structure system, as well as the unstable McMillan degree. This method does not require the transformation of the rational matrix into Smith-McMillan form and computationally is more practical. An original treatment of the problem was given in [6], where general searching methods for the maximal weight of nonnegative, natural matrices were used. This paper develops two methods for the study of the problem. The first uses the reduction of the generic McMillan degree computation to finding the maximum weight of natural matrices, and develops a structural methodology based on the new notion of column irreducibility (motivated by the related notion of the polynomial matrices) that leads to an efficient computational procedure based on tests of Boolean independence of vectors and notions of complexity of weighted Boolean matrices. The second links the generic McMillan degree computation problem to a standard problem of optimisation, known as the optimal assignment problem, which is briefly discussed at the end. Note that both approaches have strong interpretation in

graph-theoretic terms and it is possible to combine them for the development of more efficient computational procedures.

## II. PROBLEM STATEMENT AND BACKGROUND RESULTS

In this section we define the basic notions, state the problem and review the basic results. The notion of the structured transfer function is introduced below.

**Example (2.1):** An example of a  $3 \times 3$  structured proper rational transfer function matrix  $H(s)$  is given below

$$H(s) = \begin{bmatrix} A_1^2 A_2 & A_1 & A_3 \\ A_3 & A_2^2 & A_1 A_2 A_3 \\ A_1 & A_4 & A_1^2 \end{bmatrix} \quad (1)$$

where the elements  $A_i, i = 1, \dots, 4$  are repeated patterns representing, for instance, constant terms, the first order or second order dynamics,

$$A_1 = \frac{c_1}{s + \alpha_1}, A_2 = \frac{c_2}{s^2 + b_1 s + b_2}, \dots$$

where the  $\alpha_1, b_1, b_2$  etc are fixed and the  $c_i$  are constants which take generic values. By using partial fraction expansion we can decompose the transfer function matrix in the following manner:

$$\begin{aligned} H(s) &= \begin{bmatrix} A_1^2 A_2 & A_1 & A_3 \\ A_3 & A_2^2 & A_1 A_2 A_3 \\ A_1 & A_4 & A_1^2 \end{bmatrix} \\ &= \underbrace{\begin{bmatrix} A_1^2 & A_1 & 0 \\ 0 & 0 & A_1 \\ A_1 & 0 & A_1^2 \end{bmatrix}}_{H_1(s)} + \underbrace{\begin{bmatrix} A_2 & 0 & 0 \\ 0 & A_2^2 & A_2 \\ 0 & 0 & 0 \end{bmatrix}}_{H_2(s)} \\ &+ \underbrace{\begin{bmatrix} 0 & 0 & A_3 \\ A_3 & 0 & A_3 \\ 0 & 0 & 0 \end{bmatrix}}_{H_3(s)} + \underbrace{\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & A_4 & 0 \end{bmatrix}}_{H_4(s)} \end{aligned} \quad (2)$$

and the matrices  $H_i(s)$  will be called simple structured matrices. In general, we define the structured and simple structured matrix as:

**Definition (2.1):** The **structured transfer function matrix** of a system is a transfer function matrix whose entries consist of elementary dynamical terms. The elementary dynamical terms represent the basic dynamics of the system which may appear in more than one entries depending on the structure of the system. If the structured matrix consists of entries with the same elementary dynamical term, then it is called **simple**.

A structured transfer function matrix, can always be decomposed into a set of simple structured transfer function matrices by use of partial fraction expansion method for each of the dynamic terms.

**Remark (2.1):** If  $\{\lambda_1, \lambda_2, \dots, \lambda_p\}$  are the fixed pole locations of a structured transfer function  $H(s)$ , then  $H(s)$  may always be expressed as

$$H(s) = H_1(s) + H_2(s) + \dots + H_p(s)$$

where  $H_i(s)$  are simple structured transfer functions, corresponding to  $\lambda_i$  fixed pole. For the so-defined structured transfer function matrices, we define the generic McMillan degree as

**Definition (2.2):** The **generic McMillan degree** of the structured transfer function  $H(s) \in \mathbb{R}_{pr}^{m \times l}(s)$  is the McMillan degree when the gain parameters of the entries take generic values.

**Remark (2.2):** In the computation of any minor of a generic rational matrix there is no pole zero cancellation occurring and this is a mere consequence of the assumption of genericity.

By Definition (2.2) and Remark (2.1) and the definition of the McMillan degree based on the minors, we have the result:

**Proposition (2.1):** The generic McMillan degree of the structured transfer function matrix  $H(s) \in \mathbb{R}_{pr}^{m \times l}(s)$  is equal to the sum of the generic McMillan degree of the matrices  $H_1(s), H_2(s), \dots$ . That is, if  $\delta_{gm}(H_i)$  denotes the generic McMillan degree of a structured transfer function matrix  $H_i(s)$ , then

$$\delta_{gm}(H) = \delta_{gm}(H_1(s)) + \delta_{gm}(H_2(s)) + \dots \quad (3)$$

□

**Remark (2.3):** The evaluation of the generic McMillan degree of a structured transfer function matrix  $H(s)$  is reduced to finding the generic McMillan degrees of the simple structured matrices  $H_i(s)$ .

In the following we look into the methods of computing the generic McMillan degree of the simple structured transfer function matrices. First we define the concepts of order, path and weight.

**Definition (2.3):** Given a simple structured matrix  $H_i(s) \in \mathbb{R}_{pr}^{m \times l}$ ,  $m \leq l$ , the **order** of an entry in the matrix is the power of the fundamental dynamics; an **independent path** is a sequence of  $m$  elements selected from the matrix with no two elements from the same column or the same row. The **length** of a path is the number of the elements in the path and the **weight** of a path is defined to be the sum of the orders of the elements in the path. The **maximal weight** of all the independent paths of a matrix is denoted by  $\gamma(H_i)$ .

**Remark (2.4):** The constant terms of the structured transfer function matrix do not contribute to the weight and thus do not contribute to the generic McMillan degree. □

In the following we study the simple structured matrices  $H_i(s)$ . Because only the non-zero dynamic elements need to be considered, and the non-zero entries represent the same dynamic unit with different orders, for simplicity of notation, we use the orders of the dynamics of the entries only. For the example (2.1), the matrix  $H_i(s)$  is simplified to be

$$H_i(s) \begin{bmatrix} A_1^2 & A_1 & 0 \\ 0 & 0 & A_1 \\ A_1 & 0 & A_1^2 \end{bmatrix} \rightarrow I_1 = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 2 \end{bmatrix}$$

In general, a map can be defined between a simple structured matrix  $H_i(s)$  and an integer matrix  $I_i$  such that

the entries of the integer matrix correspond to the orders of the entries in  $H_i(s)$ . The maximal weight of  $H_i(s)$ , or equivalently of  $I_i$  will be called simply the weight of the matrix. Concerning the relationship between the generic McMillan degree and the weight of the paths, we have [6]:

**Proposition (2.2):** *The generic McMillan degree of the simple structured matrix  $H_i(s)$  is equal to the maximal weight,*

$$\delta_{gm}(H_i) = \gamma(H_i)$$

The integer matrix  $I_i$  can also be represented as a bipartite graph and the problem of determining its maximal weight is equivalent to a maximal weight matching problem [2]. Maximal weight matching problems in bipartite graphs are also known as assignment problems [1]. In the sequel we are presenting an alternative method for the computation of the maximal weight of  $I_i$ .

### III. REPRESENTATIONS, IRREDUCIBILITY AND WEIGHT OF NATURAL MATRICES

Let  $A \in \mathbb{N}^{m \times p}$ , where  $\mathbb{N}$  is the set of nonnegative integers. Without loss of generality we may assume  $m \geq p$ . The matrix  $A$  may be expressed as:

$$A = [\alpha_1, \alpha_2, \dots, \alpha_p], \alpha_i \in \mathbb{N}^m, i \in p \quad (4)$$

For every column  $\alpha_i$  we define as its *content* the ordered set of distinct values of the numbers in  $\alpha_i$  as

$$U(\alpha_i) = \{\delta_{1i} > \delta_{2i} > \dots > \delta_{\sigma(1)i}\} \quad (5)$$

and  $\delta_{1i}$  is the weight of the  $i$ th column. Using this notation we may represent the matrix  $A$  as shown below:

$$A = A_1 \begin{bmatrix} \delta_{11} & 0 & \dots & 0 \\ 0 & \delta_{12} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \delta_{1p} \end{bmatrix} + \dots \quad (6)$$

where  $\mu = \max\{\sigma(i), i \in p\}$ , the matrices  $A_1, A_2, \dots, A_\mu$  are Boolean matrices and in the  $\Delta_k$  matrices,  $\delta_{jk}$  take the  $k$ th value from  $U(\alpha_j)$  with  $\delta_{jk} = 0$  if  $k > \sigma(j)$ . We shall refer to the decomposition (6) as the *Weighted Boolean Representation* of  $A$ ; the matrices  $A_i$  are referred to as *Boolean coefficients* and the  $\Delta_i$  as *order matrices*. Condition (6) may also be rewritten as

$$A = \underbrace{[A_1 \dots A_\mu]}_{\langle A \rangle} \begin{bmatrix} \Delta_1 \\ \dots \\ \vdots \\ \dots \\ \Delta_\mu \end{bmatrix} \quad (7)$$

and  $\langle A \rangle$  is referred to as the  $\alpha$ -*Weighted Boolean representation*. Every column of  $\langle A \rangle$  is Boolean and is characterised by an index  $\delta_{ki}$ , where  $\delta_{ki}$  is the  $i$ th value of the matrix  $\Delta_k$ . Alternatively, every column of  $A$  can be written as

$$\alpha_i = \tilde{A}_i [\delta_{1i} \delta_{2i} \dots \delta_{\sigma(i)i}]^T = \tilde{A}_i \tilde{\delta}_i^T \quad (8)$$

so  $A$  can be expressed as

$$A = \underbrace{[\tilde{A}_1, \tilde{A}_2, \dots, \tilde{A}_p]}_{\{A\}} \begin{bmatrix} \tilde{\delta}_1 & 0 & \dots & 0 \\ 0 & \tilde{\delta}_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \tilde{\delta}_p \end{bmatrix} \quad (9)$$

and  $\{A\}$  is referred to as the  $b$ -*Weighted Boolean representation*. It should be noted that the  $b$ -representation is more economical since the  $a$ -representation may contain also some zero columns. In the following, both representations will be used.

A matrix  $A \in \mathbb{N}^{m \times p}$  is called *column irreducible*, if the matrix  $A_1$  of the weighted Boolean representation, referred to as *high coefficient matrix* has full structural rank; otherwise, the matrix is called *column reducible*. The set of indices  $\{\delta_{11}, \delta_{12}, \dots, \delta_{1p}\}$  is referred to as the set of *column weight*, or *column degrees* and the number

$$\delta(A) \triangleq \delta = \sum_{i=1}^p \delta_{1i} \quad (10)$$

is called the *complexity* of  $A$ . The irreducible, reducible classification of natural matrices has the following implications.

**Remark (3.1):** From the definition of the weight of the matrix we have that the complexity  $\delta$  of  $A$  is always an upper bound for the weight  $\gamma(A)$ , i.e

$$\gamma(A) \leq \delta \quad (11)$$

**Theorem (3.1):** *Let  $A \in \mathbb{N}^{m \times p}$ ,  $m \leq p$  and assume the representation defined by (7). The following properties hold true:*

1) *If  $A$  is irreducible, then its weight  $\gamma(A) = \gamma$  is equal to the complexity  $\delta(A) = \delta$ , i.e.*

$$\gamma = \delta = \sum_{i=1}^p \delta_{1i} \quad (12)$$

2) *If  $A$  is reducible, then  $\gamma(A) < \delta$ . Furthermore, if*

$$A = \{H_k : H_k \in \mathbb{N}^{m \times q}, q < p\}$$

*denote the set of all submatrices of  $A$  made up from subsets of its columns, such that  $H_k$  is irreducible, then*

$$\sigma = \max\{\delta(H_k), \forall H_k \in A\} \leq \gamma(A) < \delta \quad (13)$$

□

We shall refer to the number  $\sigma$  introduced above as the *index* of  $A$ .

**Remark (3.2):** Let us assume that the path that gives the weight  $\gamma(A)$  is associated with columns  $\{i_1, i_2, \dots, i_\mu\}$  of the set  $\{1, 2, \dots, \mu\}$  and that

$$\gamma(A) = \gamma_{1i_1} + \gamma_{2i_2} + \dots + \gamma_{\mu i_\mu} \quad (14)$$

Clearly,  $\gamma_{1i_1} \in U(\alpha_{i_1}), \dots, \gamma_{\mu i_\mu} \in U(\alpha_{i_\mu})$  and the Boolean matrix which is defined as the submatrix of  $\{A\}$  that corresponds to indices  $(\gamma_{1i_1}, \gamma_{2i_2}, \dots, \gamma_{\mu i_\mu})$  has full structural rank. The search for the value of  $\gamma(A)$  is thus

reduced to finding a submatrix  $\hat{A}$  of  $\{A\}$  which has the properties:

- 1) Only one column may be possibly selected from each of the blocks  $\hat{A}_1, \hat{A}_2, \dots, \hat{A}_p$  to form  $\hat{A}$ .
- 2) The resulting matrix  $\hat{A}$  has full structural rank.
- 3) The complexity of the matrix is maximal.

Remark (3.2) indicates the basics of a new algorithm that leads to the computation of  $\gamma(A)$  in an efficient way. The essentials of this new algorithm are given in the next section.

#### IV. ALGORITHM FOR DETERMINING THE WEIGHT OF A NATURAL MATRIX

Consider the matrix  $A \in \mathbb{N}^{m \times p}$  and assume that its columns are ordered according to descending weight and form the table of the column contents, i.e.

$$\begin{array}{cccc} \text{col}(1) & \text{col}(2) & \cdots & \text{col}(p) \\ \hline \delta_{11} & \delta_{12} & \cdots & \delta_{1p} \\ \delta_{21} & \delta_{22} & \cdots & \delta_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \delta_{\sigma(1)1} & \delta_{\sigma(2)2} & \cdots & \delta_{\sigma(p)p} \end{array} \quad (15)$$

where  $\delta_{11} \geq \delta_{12} \geq \cdots \geq \delta_{1p}$ . The search for the weight  $\gamma(A)$  involves the following steps:

##### STEP(I) (Preliminary Step)

Define the  $b$ -Boolean representation of the matrix  $A$  as shown in (9) and test column irreducibility by finding the structural rank of the high coefficient matrix.

- a) If  $A$  is column irreducible, then

$$\gamma(A) = \sum_{i=1}^p \delta_{i\mu} = \delta \quad (16)$$

and the search stops.

- b) If  $A$  is reducible, then  $\gamma(A) < \delta$  and compute the index of  $A$  denoted by  $\sigma(A) \triangleq \sigma$  (eqn (13)). The search for  $\gamma(A)$  then continues and involves the following steps.

##### STEP(II) (Generation of Allocation Matrices)

The objective of this step is to generate the set of all  $p$ -term sequences  $(x_1, x_2, \dots, x_p)$  such that  $x_i \in U(\alpha_i), i \in p$ . For this we will define the matrices  $B$  and  $C$ , called *allocation matrices* in the contents of this work.

**Definition (4.1):** Let  $B \in \mathbb{N}^{m \times p}$  be the matrix created according to the table of column contents 15, i.e. by the ordered set of each column:

$$B \triangleq [U(a_1), U(a_2), \dots, U(a_p)]$$

For the determination of the independent path, we will define the matrix  $C$  as follows:

**Definition (4.2):** Let  $C \in \mathbb{N}^{m \times p}$  be the *row allocation matrix* or *arrangement matrix*, given by:

$$C[\gamma_{ij}] = [\sigma_j(i)j] \quad i = 1, 2, \dots, m \quad j = 1, 2, \dots, p$$

where  $\sigma_j(i)j$  is the arrangement of the  $i$ -th maximal element of the  $j$ -th column. Since we have  $j$  columns,

we will have the same number of arrangement functions.

**Remark(4.1):** The matrix  $C$  denotes the row position of the corresponding value of  $B$  in matrix  $A$ , i.e.  $\gamma_{ij}$  is the row position of the element  $b_{ij}$  in the  $j$ -th column of matrix  $A$ .

**Remark(4.2):** Through a *greedy* search in matrix  $C$  we can obtain an independent path, which will consist a sub-optimal solution for the generic McMillan degree. We can use however this solution as a lower bound, and this concludes the second step of the algorithm.

If we denote by  $\sigma(A)$  the value for the lower bound, we now have that the generic McMillan degree  $\gamma(A)$  of  $A$  is:

$$\delta(A) > \gamma(A) \geq \sigma(A)$$

**STEP(III) (Searching Routine)** The objective here is to find the McMillan degree with the minimum number of steps. The searching routine is based on matrix  $D$ , defined as follows:

**Definition(4.3)** Let  $D \in \mathbb{N}^{m \times p}$  be the *loss allocation matrix*, given by:

$$D = [d_{ij}] = [b_{1j} - b_{ij}] \quad i = 1, 2, \dots, m \quad j = 1, 2, \dots, p$$

i.e.  $D$  is the matrix that occurs when we subtract the  $i$ -th row from the first row of  $B$ . If we include the first row of  $B$ , then the following holds true:

**Remark(4.3):** There is a one-to-one correspondence between the matrices  $B, C$ , and  $D$ . The amount  $d_{ij}$  is the difference from the maximum complexity  $\delta(A)$  if we choose the element from the  $j$ -th column and the  $\gamma_{ij}$ -th row of  $A$ .

Having the matrices  $A, B, C$ , and  $D$ , the maximum complexity of  $A$  given by  $\delta(A)$ , and a lower bound for the generic McMillan degree  $\sigma(A)$ , the search for the generic McMillan degree  $\gamma(A)$  is now defined through the following procedure:

- i) Search in matrix  $D$  for the minimum amount given by the sum:

$$d_1 = d_{i_1 1} + d_{i_2 2} + \dots + d_{i_p p}.$$

- ii) From the matrices  $B$  and  $C$  we check if the entries given by the coordinates  $\{(i_1 1), (i_2 2), \dots, (i_p p)\}$ , correspond to a matrix with full rank.

- a) If the corresponding coefficient matrix has full rank, then the algorithm has finished and the generic McMillan degree is given by

$$\delta(A) - d_1$$

and the row coordinates of the independent path are given by:

$$\{\gamma_{i_1 1}, \gamma_{i_2 2}, \dots, \gamma_{i_p p}\}$$

b) If the corresponding coefficient matrix is rank deficient, then there is no independent path in  $A$  with a complexity greater than  $\delta(A) - d$ .

iii) Return to the loss allocation matrix  $D$  and find the next minimum value  $d_2 > d_1$ .

The above result provides a systematic procedure that leads to the value of  $\gamma(A)$  in a very small number of steps. The above procedure uses the minimal possible number of steps since it is based on ordering and exploits fully the property of irreducibility which is behind the determination of  $\gamma(A)$ .

**Example (4.1):** Given is the  $10 \times 5$  integer matrix dimensions:

$$A = \begin{bmatrix} 1 & 0 & 0 & 1 & 2 \\ 0 & 2 & 4 & 10 & 0 \\ 4 & 1 & 8 & 5 & 9 \\ 1 & 0 & 6 & 6 & 2 \\ 8 & 2 & 9 & 2 & 1 \\ 2 & 1 & 2 & 3 & 0 \\ 1 & 3 & 1 & 4 & 0 \\ 6 & 4 & 7 & 5 & 0 \\ 3 & 0 & 0 & 3 & 0 \\ 2 & 1 & 3 & 5 & 8 \end{bmatrix}$$

We will create the matrices  $B$  and  $C$  and check for column irreducibility:

$$B = \begin{bmatrix} 8 & 4 & 9 & 10 & 9 \\ 6 & 3 & 8 & 6 & 8 \\ 4 & 2 & 7 & 5 & 2 \\ 3 & 2 & 6 & 5 & 2 \\ 2 & 1 & 4 & 5 & 0 \\ 2 & 1 & 3 & 4 & 0 \\ 1 & 1 & 2 & 3 & 0 \\ 1 & 0 & 1 & 3 & 0 \\ 1 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 5 & 8 & 5 & 2 & 3 \\ 8 & 7 & 3 & 4 & 10 \\ 3 & 2 & 8 & 3 & 1 \\ 9 & 5 & 4 & 8 & 4 \\ 6 & 3 & 2 & 10 & 5 \\ 10 & 6 & 10 & 7 & 2 \\ 1 & 10 & 6 & 6 & 6 \\ 7 & 1 & 7 & 9 & 7 \\ 4 & 4 & 1 & 5 & 8 \\ 2 & 9 & 9 & 1 & 9 \end{bmatrix}$$

**STEP I (Preliminary Step):**

The upper bound for the weight is given by the sum of the elements of the first row of  $B$  :

$$\delta(A) = \sum_{j=1}^p b_{1j} = 8 + 4 + 9 + 10 + 9 = 40$$

We check the first row of matrix  $C$  for irreducibility. Since we have  $[5 \ 8 \ 5 \ 2 \ 3]$ ,  $A$  is reducible. We check the third column of  $C$  to find an acceptable combination. If we replace  $c_{13}$  with  $c_{43}$ , we obtain the combination:  $[5 \ 8 \ 4 \ 2 \ 3]$ , which is acceptable. Replacing in  $B$  the

corresponding entries  $b_{13}$  with  $b_{43}$  we have a lower bound, given by the sum:

$$\sigma(A) = \sum_{j=1}^p b_{1j} = 8 + 4 + 6 + 10 + 9 = 37$$

Therefore the condition for the maximal possible weight of  $A$  is going to be:

$$37 \leq \gamma(A) < 40$$

**STEP II (Generation of the Difference Matrix):**

We generate the matrix  $D$  of dimensions  $10 \times 5$ , as follows:

$$D = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ -2 & -1 & -1 & -4 & -1 \\ -4 & -2 & -2 & -5 & -7 \\ -5 & -2 & -3 & -5 & -7 \\ -6 & -3 & -5 & -5 & -9 \\ -6 & -3 & -6 & -6 & -9 \\ -7 & -3 & -7 & -7 & -9 \\ -7 & -4 & -8 & -7 & -9 \\ -7 & -4 & -9 & -8 & -9 \\ -8 & -4 & -9 & -9 & -9 \end{bmatrix}$$

We begin now the search for the *minimum loss* according to the values of  $D$  that are closer to zero. The first set of values is:  $\{d_{22}, d_{23}, d_{25}\} = -1 = d_1$ . Therefore we will replace the corresponding elements in the first row of  $C$  according to the ones indicated by the above set.

$$\begin{aligned} [5 \ 7 \ 5 \ 2 \ 3] & \text{ not acceptable} \\ [5 \ 8 \ 3 \ 2 \ 3] & \text{ not acceptable} \\ [5 \ 8 \ 5 \ 2 \ 10] & \text{ not acceptable} \end{aligned}$$

and since there are no other possible combinations with a loss of  $-1$ , we repeat the same procedure for the next value in  $D$  closer to zero, which is given by the set:  $\{d_{21}, d_{32}, d_{33}, d_{42}, d_{22} + d_{23}, d_{22} + d_{25}, d_{23} + d_{25}\} = -2 = d_2$ . The corresponding combinations formed by the elements of matrix  $C$  are:

$$\begin{aligned} [8 \ 8 \ 5 \ 2 \ 3] & \text{ not acceptable} \\ [5 \ 2 \ 5 \ 2 \ 3] & \text{ not acceptable} \\ [5 \ 8 \ 8 \ 2 \ 3] & \text{ not acceptable} \\ [5 \ 5 \ 5 \ 2 \ 3] & \text{ not acceptable} \\ [5 \ 7 \ 3 \ 2 \ 3] & \text{ not acceptable} \\ [5 \ 7 \ 5 \ 2 \ 10] & \text{ not acceptable} \\ [5 \ 8 \ 3 \ 2 \ 10] & \text{ acceptable} \end{aligned}$$

So after only 10 iterations we obtain an acceptable combination. The weight  $\gamma(A)$  is now given by the difference:

$$\gamma(A) = \delta(A) - |d_2| = 40 - 2 = 38$$

and the algorithm stops.

**STEP III (Final Step):**

The McMillan degree of  $A$  is given as the sum:

$$a_{51} + a_{82} + a_{33} + a_{24} + a_{105} = 38,$$

which is indeed an independent path within the matrix  $A$ .

## V. THE GENERIC McMILLAN DEGREE AND THE OPTIMAL ASSIGNMENT PROBLEM

An alternative approach for the computation of the generic McMillan degree may be developed by using standard results from integer optimisation problems. Here we provide a connection between the problem of finding the generic McMillan degree, and a series of optimisation problems, more known as *optimal assignment problems*.

**Definition (5.1) [7]:** The *assignment problem* is defined on an integer matrix as the problem to choose  $n$  elements - one from each row and column - of an  $n \times n$  matrix  $C = [c_{ij}]$  so that the sum of the elements chosen is maximum.

The optimal assignment problem frequently appears in Operational Research as the problem of having to assign  $n$  workers to  $n$  jobs, or  $n$  machines to  $n$  tasks. A more mathematical description can be given, if we introduce the matrix  $X = [x_{ij}]$ , where  $x_{ij} = 0$  if  $c_{ij} = 0$  and  $x_{ij} = 1$  if  $c_{ij} \neq 0$ . Then the optimal assignment problem is to maximise the expression

$$\sum_{i=1}^n \sum_{j=1}^n x_{ij} c_{ij}$$

by choosing the matrix  $X = [x_{ij}]$  in such a way that the entries are independent. It is therefore obvious that the optimal assignment problem is equivalent to finding the McMillan degree of an  $n \times n$  matrix, or in general of any  $m \times n$  matrix, which we can transform to a square one, by adding zero columns (or rows). The independent entries of the matrix  $X$  will be equivalent to the independent path we defined earlier and the optimal assignment is equivalent to the McMillan degree.

There are known methods for solving this kind of problems, and the most popular is the *Hungarian method* [7] [1]. This method involves modifying the matrix  $X$  in such a way that it matches a permutation of the identity matrix. Such methods provide an alternative way of looking at the problem and their performance to the case of large dimension systems has to be evaluated.

## VI. CONCLUSIONS

The computation of the McMillan degree of a structured transfer function matrix has been considered using properties of column irreducibility of natural matrices. The proposed algorithm avoids the general searching methods suggested in [6] and determines the optimal solution in a small number of steps. The problem considered here is equivalent to a "maximal weight matching" problem of graph theory for which alternative solutions exist. The alternative approach for the study of such problems is their formulation as "optimal assignment problems", for which a number of algorithms exist. The comparison of this new, algebraically

based algorithm to the standard methodologies, is under investigation. There are strong indications that exploring the structural criteria based on the reduceness properties of the variants of the "optimal assignment" algorithms may be developed which explore the sparse structure of the matrices and thus lead to algorithms with reduced complexity. From the systems viewpoint, it is also interesting to explore the link of these procedures with the graph theoretic properties of the problem.

## REFERENCES

- [1] V.K. Balakrishnan. *Network Optimisation*. Chapman & Hall Mathematics, 1995.
- [2] J.W. Van der Woude. The generic dimension of a minimal realization of an ar system. *MCSS*, 8:50-64, 1995.
- [3] EPIC. Early process design integrated with control. In *Esprit II Project 2090*.
- [4] V. Hovelaque, C. Commault, and J.M. Dion. Proceedings of european control conference, ecc'97, brussels, belgium. In *Poles and Zeros in Linear Structured Systems*, 1997.
- [5] T. Kailath. *Linear Systems*. Prentice-Hall, 1980.
- [6] N. Karcanas, X.Y. Shan, and E. Milonidis. Structural identification process in early process design: The generic mcmillan degree problem. In *Proceedings of 4th IEEE Med. Symposium, MCDA'96, Chania, Crete, Greece*, 1996.
- [7] E.L. Lawler, J.K. Lenstra, A.H.G. Rinnoy Kan, and D.B. Simmons, editors. *The Travelling Salesman Problem, A guided tour of combinatorial optimisation*. John Wiley & Sons, 1985.
- [8] A.G.J. MacFarlane and N. Karcanas. Poles and zeros of linear multivariable systems: A survey of the algebraic, geometric and complex value theory. *Int. J. of Control*, 24:33-74, 1976.
- [9] M. Morari and G. Stephanopoulos. Studies in the synthesis of control structures of chemical processes, parts i and ii. *AICHE*, 26, 1980.
- [10] K. Reinschke. *Multivariable Control: A graph Theoretical Approach*. Springer-Verlag, 1988.