Observability preservation under sensor failure

Christian Commault, Jean-Michel Dion, Do Hieu Trinh

► To cite this version:

Christian Commault, Jean-Michel Dion, Do Hieu Trinh. Observability preservation under sensor failure. 45th IEEE Conference on Decision and Control, CDC'06, 2006, France. hal-00128005

HAL Id: hal-00128005

<https://hal.science/hal-00128005>

Submitted on 30 Jan 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Observability preservation under sensor failure

Christian Commault, Jean-Michel Dion and Do Hieu Trinh

Abstract—This paper is concerned with the study of observability in a structured framework. It turns out that the system is structurally observable if and only if the system is output connected and contains no contraction. We focus our attention on the observability preservation under sensor failure. We consider linear observable systems and we wonder if a given system remains observable in case of sensor failure. More precisely we will characterize among the sensors those which are critical and which failure leads to observability loss, those which are useless for observability purpose and the set of those which are useful without being critical. Using a graph approach we classify the sensors with respect to their importance for output connection preservation, contraction avoidance and then observability preservation under sensor failure.

I. INTRODUCTION

In this paper, we consider linear observable systems and we wonder if a given system remains observable in case of sensor failure. More precisely we will characterize among the sensors those which are critical and which failure leads to observability loss, those which are useless for observability purpose and the set of those which are useful without being critical. We study the problem using a graph approach which can be used easily for usual linear systems but will focus our attention on linear structured systems which represent a large class of parameter dependent linear systems. This approach was pioneered by Lin [1]. Generic properties for such systems can be obtained from a graph associated to the system. The dualization of the results given by Lin allows to characterize the generic observability of structured systems. It turns out that the system is structurally observable if and only if the system is output connected and contains no contraction. Starting from these conditions for observability we will here determine the critical sensors, which will be called essential, as well as the useless ones for both conditions. For each condition we will provide with a structural analysis of the admissible conditions and give global results for observability preservation. We will illustrate the results on simple academic examples and point out classical combinatorial algorithms for getting the solutions.

The outline of this paper is as follows. The linear structured systems are presented in section 2. We revisit structural observability conditions in section 3. In section 4 we study the output connection preservation and in section 5 the contraction avoidance. The problem of observability preservation under sensor failure is finally considered in section 6. Some concluding remarks end the paper.

Christian Commault, Jean-Michel Dion and Do Hieu Trinh
Laboratoire d'Automatique de Grenoble. LAG-CNRS, ENSIEG-INPG
BP 46, 38402 Saint Martin d'Hères, France.
Email: {Christian.Commault, Jean-Michel.Dion}@inpg.fr
Do-Hieu.Trinh@lag.ensieg.inpg.fr

II. LINEAR STRUCTURED SYSTEMS

In this part, we recall some definitions and results on linear structured systems. More details can be found in [2].

We consider linear systems with parameterized entries and denoted by Σ_Λ . In this paper, we will only be concerned with observability and we will not take into account input variables.

$$\Sigma_\Lambda \begin{cases} \dot{x}(t) = Ax(t) \\ y(t) = Cx(t) \end{cases}, \quad (1)$$

where $x(t) \in \mathbb{R}^n$ is the state vector and $y(t) \in \mathbb{R}^p$ the measured output vector. A and C are matrices of appropriate dimensions.

This system is called a linear structured system if the entries of the composite matrix $J = \begin{bmatrix} A \\ C \end{bmatrix}$ are either fixed zeros or independent parameters (not related by algebraic equations). $\Lambda = \{\lambda_1, \lambda_2, \dots, \lambda_k\}$ denotes the set of independent parameters of the composite matrix J . For the sake of simplicity the dependence of the system matrices on Λ will not be made explicit in the notation. A structured system represents a large class of parameter dependent linear systems. The structure is given by the location of the fixed zero entries of J .

For such systems, one can study generic properties i.e. properties which are true for almost all values of the parameters collected in Λ [3], [4]. More precisely a property is said to be generic (or structural) if it is true for all values of the parameters (i.e. any $\Lambda \in \mathbb{R}^k$) outside a proper algebraic variety of the parameter space. A directed graph $G(\Sigma_\Lambda) = (Z, W)$ can be easily associated to the structured system Σ_Λ of type (1) where the matrix $\begin{bmatrix} A \\ C \end{bmatrix}$ is structured:

- the vertex set is $Z = X \cup Y$ where X and Y are the state and output sets given by $\{x_1, x_2, \dots, x_n\}$ and $\{y_1, y_2, \dots, y_p\}$ respectively,
- the edge set is $W = \{(x_i, x_j) | A_{ji} \neq 0\} \cup \{(x_i, y_j) | C_{ji} \neq 0\}$, where A_{ji} (resp. C_{ji}) denotes the entry (j, i) of the matrix A (resp. C).

Moreover, recall that a path in $G(\Sigma_\Lambda)$ from a vertex $i_{\mu 0}$ to a vertex $i_{\mu q}$ is a sequence of edges $(i_{\mu 0}, i_{\mu 1}), (i_{\mu 1}, i_{\mu 2}), \dots, (i_{\mu q-1}, i_{\mu q})$ such that $i_{\mu t} \in Z$ for $t = 0, 1, \dots, q$ and $(i_{\mu t-1}, i_{\mu t}) \in W$ for $t = 1, 2, \dots, q$. If $i_{\mu 0} \in X$ and $i_{\mu q} \in Y$, the path is called a state-output path. The system Σ_Λ is said to be output-connected if for any state vertex x_i there exists a state-output path with initial vertex

x_i .

A set of paths with no common vertex is said to be vertex disjoint.

We will also use undirected graphs composed of vertices and undirected edges. In such graphs the undirected paths will be called walks.

Example 1: Let Σ_Λ be the linear structured system defined by its structured matrices.

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \lambda_1 & 0 & 0 \end{bmatrix}, C = \begin{bmatrix} 0 & 0 & \lambda_2 \\ 0 & \lambda_3 & \lambda_4 \end{bmatrix}, \quad (2)$$

Its associated graph $G(\Sigma_\Lambda)$ is given in Figure 1.

Fig. 1. $G(\Sigma_\Lambda)$ for Example 1

III. OBSERVABILITY OF LINEAR STRUCTURED SYSTEMS

The structural controllability or the dual notion of observability has been studied in several papers [1], [5], [6].

Define the concept of contraction which is the dual notion of the dilation defined by Lin in the study of controllability.

Definition 1: Let Σ_Λ be the linear structured system defined by (1) with associated graph $G(\Sigma_\Lambda)$ with vertex set Z and edge set W . Consider a set S made of k_S state vertices. Denote $E(S)$ the set of vertices w_i for $i = 1, \dots, l_S$ of Z , such that there exists an edge (x_j, w_i) of W where $x_j \in S$. S is said to be a contraction if

$$k_S - l_S > 0. \quad (3)$$

Recall the graph characterization of the structural observability, which will be useful later [1], [3].

Proposition 1: Let Σ_Λ be the linear structured system defined by (1) with its associated graph $G(\Sigma_\Lambda)$. The system is structurally observable if and only if:

- 1) The system Σ_Λ is output-connected,
- 2) $G(\Sigma_\Lambda)$ contains no contraction.

Consider again the system Σ_Λ of Example 1. All the state vertices are output connected and its associated graph $G(\Sigma_\Lambda)$ does not have any contraction so Σ_Λ is structurally observable. If the output sensor y_1 fails, the set $S = \{x_2, x_3\}$ is a contraction since $E(S) = \{y_2\}$ and $k_S - l_S = 1$. If the output sensor y_2 fails, x_2 is not any more output connected and $\{x_2\}$ is a contraction.

In the following sections, we will focus our interest on structurally observable systems. We look for the the maximal

number of sensors which can be suppressed while preserving the structural observability of the system. We will tackle this problem in two stages. In a first step, we will study the preservation of the output connection property and give the set of sensors which can or cannot be suppressed to keep this property. In a second step, we will study the contraction avoidance when removing sensors and give the set of sensors which can or cannot be suppressed without creating a contraction.

IV. OUTPUT CONNECTION PRESERVATION

A. Strongly connected components and reduced connection graph

Let Σ_Λ be the linear structured system defined by (1) with its associated graph $G(\Sigma_\Lambda)$. Two vertices v_i and v_j of $G(\Sigma_\Lambda)$ are said to be equivalent if there exists a path from v_i to v_j and a path from v_j to v_i . In this context v_i is assumed to be equivalent to itself. The equivalent classes corresponding to this equivalence relation are called the strongly connected components of $G(\Sigma_\Lambda)$. Standard combinatorial optimization algorithms exist to get the canonical decomposition of the graph into strongly connected components. The output vertices y_i are strongly connected components composed of a unique vertex. The strongly connected components can be endowed with a natural partial order. The strongly connected components C_i and C_j are such that $C_i \preceq C_j$ if there exists an edge (v_j, v_i) where $v_i \in C_i$ and $v_j \in C_j$. The infimal elements with this order are the strongly connected components with no outgoing edges. Notice that the output vertices are such infimal elements.

Proposition 2: Let Σ_Λ be the linear structured system defined by (1) with its associated graph $G(\Sigma_\Lambda)$. Σ_Λ is output connected if and only if all of the infimal components of $G(\Sigma_\Lambda)$ are output vertices.

Proof: Any state vertex of Σ_Λ is connected to one or more infimal components of $G(\Sigma_\Lambda)$. When all of infimal components of $G(\Sigma_\Lambda)$ are output vertices, all state vertices are output connected. Conversely if an infimal component is not an output vertex, the vertices of this component are not output connected. ■

We define pre-infimal components as the strongly connected components which have no path to others components except output vertices.

To study output connection let us now introduce a new undirected graph.

Definition 2: The graph $C(\Sigma_\Lambda) = (Z_c, W_c)$ called *reduced connection graph* is derived from $G(\Sigma_\Lambda)$ as follows:

- the vertex set is $Z_c = V \cup V' \cup Y \cup z$ with $V = \{v_1, v_2, \dots, v_k\}$ where v_i corresponds to the pre-infimal component I_i of $G(\Sigma_\Lambda)$, $V' = \{v'_1, v'_2, \dots, v'_l\}$ where v'_i corresponds to an infimal component of $G(\Sigma_\Lambda)$ which is not an output vertex, Y is the output set $\{y_1, y_2, \dots, y_p\}$ and z is an additional vertex,
- the edge set is $W_c = \{(v_i, y_j) \mid \text{when there exists a path from } I_i \text{ to } y_j\} \cup \{(y_j, z) \text{ for any } j\}$.

$C(\Sigma_\Lambda)$ is a simple way to analyze the output connection of the system Σ_Λ .

Proposition 3: The graph $G(\Sigma_\Lambda)$ is output connected if and only if the reduced connection graph $C(\Sigma_\Lambda)$ is connected.

Proof: If $C(\Sigma_\Lambda)$ is connected there is a walk from each v_i to z in $C(\Sigma_\Lambda)$ and this walk passes through an output y_j . Then in $G(\Sigma_\Lambda)$ all the vertices of a pre-infimal component are output connected, other state vertices being connected to a pre-infimal component will also be output connected. Conversely if $G(\Sigma_\Lambda)$ is not output connected there exists at least one infimal component which is not an output vertex and then the corresponding vertex v'_i is isolated in $C(\Sigma_\Lambda)$, therefore $C(\Sigma_\Lambda)$ is not connected.

Example 2: Let Σ_Λ be the linear observable structured system which associated graph $G(\Sigma_\Lambda)$ is given in Figure 2.

Fig. 2. $G(\Sigma_\Lambda)$ for Example 2

Fig. 3. $C(\Sigma_\Lambda)$ for Example 2

In this example all infimal components are outputs therefore $V' = \emptyset$, we have three pre-infimal components $I_1 = \{x_1\}$, $I_2 = \{x_2\}$, $I_3 = \{x_5\}$. The reduced connection graph $C(\Sigma_\Lambda)$ is given in Figure 3. As shown in the reduced connection graph, $C(\Sigma_\Lambda)$ is connected so $G(\Sigma_\Lambda)$ is output connected.

B. Minimal separators

Definition 3: Let $H(Z, W)$ be an undirected connected graph. For $S \subseteq Z$, $\varepsilon(S)$ denotes the set of connected components of $H(Z \setminus S, \widehat{W})$ where $\widehat{W}$ is the edge set W minus the incident edges to vertices of S .

- S is called a *separator* if the number of connected components of $H(Z \setminus S, \widehat{W})$, is greater than or equal to two i.e. $|\varepsilon(S)| \geq 2$.
- S is called a *minimal separator* if no proper subset of S is a separator of G .

Remark 1: If S of cardinal greater than one is a minimal separator of a connected graph H , for any vertex $s \in S$, $\{S \setminus s\}$ is not a separator, i.e. the suppression of $\{S \setminus s\}$ does not disconnect H .

The reduced connection graph $C(\Sigma_\Lambda)$ captures all the information concerning output connection. We will use minimal separators only made of output vertices on this graph to analyse the connection preservation under sensor failure.

C. Connection preservation under sensor failure

In this subsection we will focus our attention on separators which are only made of output vertices.

Proposition 4: Let Σ_Λ be the linear observable structured system defined by (1) with its associated graph $G(\Sigma_\Lambda)$ and its reduced connection graph $C(\Sigma_\Lambda)$. Output connection of Σ_Λ is preserved if and only if there exists at least one non failing sensor in each minimal separator of $C(\Sigma_\Lambda)$ included in Y .

Proof: Disconnecting $C(\Sigma_\Lambda)$ is equivalent to suppress completely at least one minimal separator. When we do not suppress completely any minimal separator (i.e. we keep at least one non failing sensor in each separator), $C(\Sigma_\Lambda)$ remains connected so $G(\Sigma_\Lambda)$ from Proposition 3 is still output connected ■

To find out the set of minimal separators $SM \subseteq Y$ in a $C(\Sigma_\Lambda)$, we can follow the following algorithm:

Algorithm 1: Listing all minimal separators

- 1) For all v_i :
 - $S(v_i) := \{y_j | (v_i, y_j) \in W_c\}$
- 2) For all $S(v_i)$:
 - If there is no k such that $S(v_k) \subset S(v_i)$, then $S(v_i)$ is a minimal separator.

The construction of the algorithm comes from the fact that $S(v_i)$ is the set of neighbor vertices of v_i , so it is a separator for v_i . Moreover, if $S(v_i)$ does not contain any other separator, it is minimal.

Remark 2: With the set of minimal separators SM , we can classify the sensors with respect to their importance for the output connection of the system:

- The *essential sensors* $\bar{E}$ coincide with the minimal separators of cardinality one. If we loose one of these sensors, the output connection is lost.

- The sensors $\bar{L}$ which are not in any minimal separator are called *useless sensors*. These sensors do not play any role for the output connection of the system.
- The *useful sensors* $\bar{F}$ are those which belong to at least one minimal separator. Each one individually may fail without loosing output connection but they may contribute to output connection in case of failure of the other sensors.

Consider again Example 2. In the set of output vertices Y , we found two minimal separators $S_1 = \{y_2, y_3\}$ and $S_2 = \{y_4\}$. So $\bar{L} = \{y_1, y_5\}$ are useless sensors because they do not belong to any minimal separator. The useful sensors are $\bar{F} = \{y_2, y_3, y_4\}$ and $\bar{E} = \{y_4\}$ is an essential sensor. To keep this system output connected, we have to ensure at least that $\{y_2, y_4\}$ or $\{y_3, y_4\}$ have no failure.

When we have many minimal separators in Y , to keep the system output connected, we have to keep at least one sensor in each minimal separator - see Proposition 4. To minimize the number of sensor to conserve is a NP-complete problem. This problem is well know as the "hitting set" problem [7]. But we the size of this problem is strongly reduced with respect to the size of Σ_Λ .

V. CONTRACTION AVOIDANCE

A. The bipartite graph

In this subsection, we will analyse in a finer way contractions. This will be performed using a bipartite graph associated with the system Σ_Λ . We consider a linear structured system Σ_Λ of type (1) as previously. Let us now introduce the bipartite graph $B(\Sigma_\Lambda)$ as follows.

The bipartite graph of this system is $B(\Sigma_\Lambda) = (B^+, B^-; W')$ where the sets B^+ and B^- are two disjoint vertex sets and W' is the edge set. The vertex set B^+ is given by X^+ , the vertex set B^- is given by $X^- \cup Y$, with $X^+ = \{x_1^+, \dots, x_n^+\}$ the first set of state vertices, $X^- = \{x_1^-, \dots, x_n^-\}$ the second set of state vertices and $Y = \{y_1, \dots, y_p\}$ the set of output vertices. Notice that here we have split each state vertex x_i of $G(\Sigma_\Lambda)$ into two vertices x_i^+ and x_i^- . The edge set W' is described by $W_A \cup W_C$ with $W_A = \{(x_j^+, x_i^-) | A_{ij} \neq 0\}$ and $W_C = \{(x_j^+, y_i) | C_{ij} \neq 0\}$. In the latter, for instance $A_{ij} \neq 0$ means that the (i, j) -th entry of the matrix A is a parameter (structurally nonzero).

A *matching* in a bipartite graph $B = (B^+, B^-; W')$ is an edge set $M \subseteq W'$ such that the edges in M have no common vertex. We denote by $\partial^+ M$ (resp. $\partial^- M$) the set of vertices in B^+ (resp. B^-) incident to the edges in M . A matching M is called maximal if it has a maximum cardinality. The cardinality of a matching, i.e. the number of edges it consists of, is also called its size. The maximal matching problem is the problem of finding a matching of maximal cardinality. This problem can be solved using very efficient algorithms based on alternate augmenting chains or ideas of maximum flow theory [8]. This notion allows a simple characterization of the generic rank of a structured matrix in terms of its

bipartite graph [9] which in our problem can be stated as follows.

Proposition 5: Let Σ_Λ be the linear structured system defined by (1) with its associated bipartite graph $B(\Sigma_\Lambda)$. There is no contraction in $G(\Sigma_\Lambda)$ if and only if there exists a size n matching in $B(\Sigma_\Lambda)$.

With a matching M we associate a graph $B_M = (B^+ \cup B^-; \bar{W}')$ so that: $(v, w) \in \bar{W}' \Leftrightarrow (v, w) \in W'$ or $(w, v) \in M$. Denote $S^+ = B^+ \setminus \partial^+ M$ and $S^- = B^- \setminus \partial^- M$. Denote $v \rightsquigarrow w$ for $\{v; w\} \in \bar{W}'$ if there is a path from v to w in B_M .

B. The Dulmage-Mendelsohn (DM) decomposition

The DM-Decomposition allows to decompose a bipartite graph B into unique partially ordered irreducible bipartite subgraphs $B_i = (B_i^+, B_i^-; W'_i)$ called the DM-components:

Algorithm 2: DM-Decomposition [3]

1. Find a maximum matching M on $B = (B^+, B^-; W')$.
2. Let $B_0 = \{B_0^+ \cup B_0^-\} = \{v \in B^+ \cup B^- | w \rightsquigarrow v \text{ on } B_M \text{ for some } w \text{ in } S^+\}$ where $B_0^+ = \{B^+ \cap B_0\}$ and $B_0^- = \{B^- \cap B_0\}$. B_0 is called the minimal inconsistent part or horizontal tail of B .
3. Let $B_\infty = \{B_\infty^+ \cup B_\infty^-\} = \{v \in B^+ \cup B^- | v \rightsquigarrow w \text{ on } B_M \text{ for some } w \text{ in } S^-\}$ where $B_\infty^+ = \{B^+ \cap B_\infty\}$ and $B_\infty^- = \{B^- \cap B_\infty\}$.
4. Let V_i ($i=1, \dots, r$) be the strong components of the graph obtained from B_M by deleting the vertices of $B_0 \cup V_\infty$ and the edges incident thereto.
5. Define the partial order $\prec$ on $\{B_i | i = 0, 1, \dots, r, \infty\}$ as follows: $B_i \prec B_j \Leftrightarrow v_j \rightsquigarrow v_i$ on B_M for some $v_i \in B_i$ and $v_j \in B_j$.

Fig. 4. $B(\Sigma_\Lambda)$ for Example 2

Example 3: Consider the system Σ_Λ which associated graph $G(\Sigma_\Lambda)$ is depicted in Figure 2 and consider the

associated bipartite graph $B(\Sigma_\Lambda)$. We will find the DM-Decomposition of $B(\Sigma_\Lambda)$. First of all, a maximum matching is defined by $\{x_6^+ \rightarrow x_3^-; x_5^+ \rightarrow y_4; x_4^+ \rightarrow x_2^-; x_3^+ \rightarrow x_1^-; x_1^+ \rightarrow y_2; x_2^+ \rightarrow y_3\}$, so that we have $S^+ = \emptyset$ and $S^- = \{y_5; x_5^-; y_1; x_4^-; x_6^-\}$. Then $B_0 = \emptyset$ and $B_\infty = \{x_4^-; x_6^-; y_5; x_4^+; x_2^-; x_5^-; y_1; x_1^+; y_2; x_3^+; x_1^-; x_2^+; y_3\}$. Finally, $B_1 = \{x_6^+; x_3^-\}$ and $B_2 = \{x_5^+; y_4\}$ as shown in the Figure 4.

C. Contraction avoidance

Proposition 6: Let Σ_Λ be the linear structured system defined by (1) with its associated graph $G(\Sigma_\Lambda)$ and associated bipartite graph $B(\Sigma_\Lambda)$. $G(\Sigma_\Lambda)$ has not any contraction if and only if the DM-Decomposition of $B(\Sigma_\Lambda)$ has no minimal inconsistent part B_0 .

Proof: Follows directly from the DM-Decomposition properties and from the definition of a contraction ■

We will look for the outputs which can be suppressed without creating contractions, i.e. the DM-Decomposition of $B(\Sigma_\Lambda)$ has no minimal inconsistent part B_0 or there exists a matching of size n in $B(\Sigma_\Lambda)$. So all output vertices which are not covered by the matching of size n can fail.

Proposition 7: Let Σ_Λ be the linear structured system defined by (1) with associated graph $G(\Sigma_\Lambda)$ and associated bipartite graph $B(\Sigma_\Lambda)$. All output vertices belonging to strong components B_i , ($i = 1, \dots, r$) in the DM-Decomposition of $B(\Sigma_\Lambda)$ are essential to avoid contraction.

Proof: Follows the DM-Decomposition properties, the number of vertices of B_i^+ equals to the number of vertices of B_i^- . If we suppress one vertex of B_i^- , it will create a contraction. ■

Proposition 8: Consider Σ_Λ the linear structured system defined by (1) with associated graph $G(\Sigma_\Lambda)$ and associated bipartite graph $B(\Sigma_\Lambda)$. Let B_∞ be the minimal inconsistent part of the DM-Decomposition of $B(\Sigma_\Lambda)$. Decompose B_∞ in connected bipartite subgraphs.

In a connected bipartite subgraph, if there exists a maximal matching which does not cover any output vertex, these output vertices are useless to avoid contraction. Otherwise, these output vertices are useful.

Proof: A completer

Remark 3: With the help of the DM-decomposition we can classify the sensors with respect to their importance for the contraction avoidance:

- The *essential sensors* $\hat{E}$ coincide with the set of outputs which belong to strong components B_i . If we loose one of these sensors, we create a contraction.
- The set of useless sensors of Proposition 8 is denoted by $\hat{L}$. These sensors do not play any role for the contraction avoidance of the system.
- The set of useful sensors of Proposition 8 is denoted by $\hat{F}$. Each one individually may fail without creating a contraction but they may contribute to avoid contraction in case of failure of other sensors.

To check in such a connected bipartite subgraph $B^* = (B^+, B^-; W')$, if there exists a maximal matching which does not cover any output vertex, we can use the following algorithm:

Algorithm 3: 1. For any arc $(v^+, v^-) \in W'$, assign it a weight 0 if v^- is a state vertex, a weight 1 if v^- is an output vertex.

2. Find a maximal matching of minimal weight in B^* . Note W_m the weight of this matching i.e. the sum of all the arc weights it consists of.

3. If $W_m = 0$ the maximal matching does not cover any output vertex. Otherwise, $W_m > 0$, any maximal matching covers at least one output vertex.

To find a maximal matching of minimal weight in a bipartite graph, we can use the algorithm developed in [10] with running time $O(n * (m + n \log n))$.

Consider again Example 2 which associated graph $G(\Sigma_\Lambda)$ is depicted in Figure 2. The DM-Decomposition of associated bipartite graph $B(\Sigma_\Lambda)$ is depicted in Figure 4.

The essential sensor is $\hat{E} = y_4$ because $y_4 \in B_2$. In B_∞ , we have two connected bipartite subgraphs composed of $\{y_5; x_4^+; x_2^-; x_5^-\}$ and $\{y_1; x_1^+; y_2; x_3^+; x_1^-; x_2^+; y_3\}$. In the first one, we have a maximal matching $\{x_4^+; x_2^-\}$ which does not cover the output y_5 so $\hat{L} = \{y_5\}$ is a useless sensor for contraction avoidance. In the second one, any maximal matching covers at least two of three outputs $\{y_1, y_2, y_3\}$ so the set of useful sensors for contraction avoidance is $\hat{F} = \{y_1, y_2, y_3\}$.

VI. OBSERVABILITY PRESERVATION

To keep the system observable while suppressing sensors, the system with remaining sensors must satisfy output connectivity property and absence of contraction. Proposition 4 and Remark 2 give the set of sensors which can be suppressed or can not be suppressed to keep the output connectivity property. Proposition 7 and Proposition 8 give the set of sensors which can be suppressed without creating a contraction.

It is clear that a sensor which is essential for output connection or contraction avoidance is essential for observability preservation of the system. A sensor which is useless for output connection and contraction avoidance is useless for observability preservation of the system. Finally, we have the following Theorem:

Theorem 1: Let Σ_Λ be the linear structured system defined by (1). Using the notations of Remarks 2 and 3, to preserve the observability of the system:

- The set of essential sensors, i.e. the sensors which cannot fail, is given by $E = \bar{E} \cup \hat{E}$
- The set of useless sensors, i.e. the sensors which do not play any role for observability, is given by $L = \bar{L} \cap \hat{L}$
- The set of useful sensors, i.e. which may fail individually without observability loss but may contribute to observability in case of sensor failures, is given by $F = (Y \setminus L) \setminus E$.

Reconsider the observable system in Example 2. In this Example, we have $\bar{L} = \{y_1, y_5\}$, $\bar{F} = \{y_2, y_3, y_4\}$ and $\bar{E} = \{y_4\}$. By Example ??, we found $\hat{E} = y_4$, $\hat{L} = \{y_5\}$ and $F = \{y_1, y_2, y_3\}$. Finally, we have:

- The essential sensors $E = \bar{E} \cup \hat{E} = \{y_4\}$
- The useless sensors $L = \bar{L} \cap \hat{L} = \{y_5\}$
- The useful sensors $F = Y \setminus E \setminus L = \{y_1, y_2, y_3\}$

VII. CONCLUDING REMARKS

In this paper, we revisited the observability analysis for structured systems. Using a graph approach we focused our attention on observability preservation under sensor failure and classified the sensors with respect to their criticality concerning observability. The use of standard algorithms or decompositions for reducing the problem complexity allows to get easily tractable algorithms. The proposed approach is well suited for structural analysis prior to observer design and the numerical implementation is simple. It allows to determine the sensors which are essential i.e. those whose failure will lead to observability loss, the sensors which are useless for observability preservation and those which are useful without being essential. The output connection analysis is valid for standard linear systems and is potentially interesting for nonlinear systems when considering linearized models of fixed structure and varying parameters around a set point. We think that this approach can also be useful for solving fault detection and diagnosis problem in case of sensor failure.

REFERENCES

- [1] C. Lin, "Structural controllability," *IEEE Trans. Automat. Control*, vol. 19, no. 3, pp. 201–208, June 1974.
- [2] J. M. Dion, C. Commault, and J. van der Woude, "Generic properties and control of linear structured systems: a survey," *Automatica*, vol. 39, no. 7, pp. 1125–1144, 2003.
- [3] K. Murota, *Systems Analysis by Graphs and Matroids*, ser. Algorithms and Combinatorics. Springer-Verlag New-York, Inc., 1987, vol. 3.
- [4] K. Reinschke, *Multivariable control : A graph-theoretic approach*, ser. Lect. Notes in Control and Information Sciences. Springer-Verlag, 1988, vol. 108.
- [5] R. Shields and J. Pearson, "Structural controllability of multiinput linear systems," *IEEE Trans. Automat. Contr.*, vol. 21, pp. 203–212, 1976.
- [6] K. Glover and L. Silverman, "Characterization of structural controllability," *IEEE Trans. Automat. Contr.*, vol. 21, pp. 534–537, 1976.
- [7] M. Garey and D. Johnson, *Computers and intractability: A guide to the theory of NP-Completeness*. W.H. Freeman And Company, 1999.
- [8] J. Hopcroft and R. Karp, "An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs," *SIAM J. on Comp.*, vol. 2, pp. 225–231, 1973.
- [9] C. Commault, J. M. Dion, and J. van der Woude, "Characterization of generic properties of linear structured systems for efficient computations," *Kybernetika*, vol. 38, no. 5, pp. 503–520, 2002.
- [10] H. N. Gabow, "Data structures for weighted matching and nearest common ancestors with linking," in *SODA '90: Proceedings of the first annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 1990, pp. 434–443.