Connectivity condition for structural properties using a graph theoretical approach: Probabilistic reliability assessment

Manal Dakil, Christophe Simon, Taha Boukhobza

Connectivity condition for structural properties using a graph theoretical approach: Probabilistic reliability assessment

M. Dakil [*,**] **C. Simon** [*,**] **T. Boukhobza** [*,**]

[*] *Université de Lorraine, Centre de Recherche en Automatique de Nancy, UMR 7039, Vandoeuvre-lès-Nancy, F-54506, France*
[**] *CNRS, Centre de Recherche en Automatique de Nancy,UMR 7039, Vandoeuvre-lès-Nancy, F-54506, France*
{*manal.dakil, christophe.simon, taha.boukhobza*}*@univ-lorraine.fr*

Abstract: In a structured system graphically represented, the satisfaction of structural properties depends on graphical conditions. Among these conditions, we choose to study one of the most commonly used conditions, the so-called connectivity condition. The validity of this condition depends on the validity of some edges in the system graphical representation. The corresponding edges reflect the operating state of the physical system components. To study the validity of the connectivity condition, this paper describes a methodology which provides a Boolean expression of the connectivity condition based on the components operating state. As an application of the proposed Boolean expression, we can define the reliability characteristics of the connectivity condition and for how much time it remains satisfied knowing the reliability characteristics of the system components.

Keywords: Structural properties; connectivity condition; graph-theoretical approach; reliability.

1. INTRODUCTION

Systems properties such as controllability, observability, disturbance rejection ... are subject of many studies in automatic control due to the major role they play in control systems theory using algebraic and geometric tools. The most important properties depend on the system structure more than the values of its internal parameters [Willems, 1986]. In most cases, the parameters of the studied system model are not always numerically fixed in the early steps of its life-cycle. Thus, the use of a generic representation can be considered which do not directly depend on the system parameters. This model is based on matrices containing “zero”and “non-zero”parameters and it is useful for the graphical representation which reflects the relationship between the variables of the system studied. This allows to study the system and its structural properties in a simple and intuitive way. [Lin, 1974] is the first one which study the structured systems where the author provides their graphical representation as a directed graph. The paper gives the condition for the structural controllability of single-input systems based on the existence of cycles, paths forming “cactus”. Based on this original work, many other works have been developed to study more structural properties by graphical approaches (see [Dion et al., 2003, Boukhobza and Hamelin, 2011b, Boukhobza, 2010]).

All the cited papers are concerned by the verification if a structured system satisfies structural properties or not, but they do not give enough informations about how long these structural properties are still satisfied for this system and if their satisfaction can tolerate one or several failures which can occur on the system components.

In dependability engineering, reliability analysis is an important engineering task in many sectors. By analyzing failures of components and their propagation in systems, reliability analysis addresses several problematic like safety, risk analysis, sustainability in all steps of the system lifecycle. It characterizes the ability of the system to perform its functions during a specified period of time under certain conditions. Reliability depends on the functional architecture of the studied system and the characteristics of the system components. It is usually expressed in term of the reliability of its subsystems or components. Thus, reliability is relied to the occurrence of some undesirable events (failures) on the system components and the impact of these events on the whole system. We can find a lot of usual tools of reliability [Villemeur, 1992] or Bayesian networks in [Langseth and Portinale, 2007]. [Kaufmann et al., 1977] also propose a graph theoretical approach based on a function useful to compute system reliability.

Components failures can impact structural properties and system reliability, that is why many researchers developed works merging the reliability theories and automatic control studies. [Doguc and Ramirez-Marquez, 2009] provides a method for constructing a Bayesian network model for estimating system reliability. In [Conrard et al., 2012] the authors are interested in Fault Tolerant Control systems (FTC) and they introduce the parameter Fault Tolerant Level (FTL) which characterizes the number of tolerable failures in a system. [Staroswiecki, 2006] solves the general fault tolerance problem through a structural analysis. The author studies the structural observability in good/failed operating state and the link with two basic dependability concepts named critical faults and reliability is studied based on a graph-theoretical approach. In [Maza et al., 2012], the authors characterize the reliability of controllability knowing the reliability of the actuators using a graph theoretical approach. They consider that only actuators can fail.

In structured systems, each structured property is satisfied when it verifies some graphical conditions. The so-called connectivity, complete matching, distance and linking conditions are the most used graphical conditions. For example, structural controllability and observability in linear systems should verify the connectivity and complete matching conditions [Dion et al., 2003] and state and input observability necessitates connectivity, complete matching and distance conditions for structured switching systems[Boukhobza and Hamelin, 2011a].
In this paper, we choose to study the connectivity condition since it is necessary for the satisfaction of several structural properties. The originality of the paper is to combine structural analysis and reliability. The provided method is based on graphical approach and consists on formulating the connectivity condition depending on the validity of some edges in the graphical representation of the studied system. Each edge is linked to certain components in the system by a specific function. Thus, the connectivity condition is expressed as a Boolean expression based on the events (failures) that can occur on the system components. Contrary to other works such as [Maza et al., 2012] which consider that external components (actuators and sensors) are the only components that can fail and [Commault and Dion, 2007] where the authors classify the sensors/actuators as essential, useful and useless, we assume that connectivity condition depends on the events on both internal and external components. In a second step, we are not interested just by the number of tolerable failures as in [Conrard et al., 2012], but in the reliability of the connectivity condition. Thus, the provided Boolean expression is used to compute the reliability of the connectivity condition knowing the reliability of the system components. In this paper, we propose to compute reliability using a common tool of dependability engineering, it is Bayesian networks.

The paper is organized as follows. Section 2 is dedicated to the problem statement of connectivity conditions which is the main idea of the study. In Section 3, we recall some graphical definitions and notations useful for the rest of the paper. Section 4 details the developed method to elaborate the graphical connectivity conditions that determine the maintaining or not of the connectivity between two sets of vertices. From this, in Section 5, the reliability of this graphical condition is computed knowing the reliability of the system components.

2. PROBLEM STATEMENT

Any structured system must satisfy some structural properties in order to carry out its mission in good conditions. Each structural property is compliant with some graphical conditions (connectivity, complete matching, distance, linking, . . .) which are related to the type of the studied system (linear, bilinear, switching system,. . .).

Connectivity condition is one of the most used graphical conditions. It should be verified in many structural properties, such as observability, controllability,. . . Using a directed graph (denoted digraph), the compliance of this condition can be verified graphically in relation with the validity of some edges or paths in the digraph, and each edge is related to the system components operating conditions. An edge (v_i, v_j) is valid if the corresponding components is not failed. Thus, connectivity condition will be formalized as a Boolean expression based on the edges validity corresponding to the components availability.

In order to state the connectivity condition, let us consider a digraph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of vertices, and $\mathcal{E}$ is the set of edges. The connectivity condition between two sets of vertices $V_1 \subseteq \mathcal{V}$ and $V_2 \subseteq \mathcal{V}$ in the digraph $\mathcal{G}$ consists on the validity of at least one path from each elements of V_1 to at least one element of V_2.

In order to keep satisfied a structural property depending on the connectivity condition, we should make sure that the condition is verified. This condition remains valid if some edges constituting the digraph remain valid, which means that the components associated to these edges are not down (or failed). In this paper, our goal is to provide a Boolean expression for the connectivity condition. To keep this condition verified, the corresponding Boolean expression must be $true$. The validity of this condition is based on the validity or not of some edges which are related to the system components in the digraph.
Based on the operating conditions, the obtained Boolean expression allows us to assess the probability to maintain a structural property verified thanks to the components reliability.

3. DEFINITIONS AND GRAPHICAL REPRESENTATION

The structure of a structured system can be represented by a graph based on a knowledge model of the studied system. This graph shows the relationship between the system variables.

- A digraph $\mathcal{G}$ is constituted of a set of elements called "vertices" and a set of ordered pairs of vertices called "edges". We denote $\mathcal{V}$ the vertex set and $\mathcal{E}$ the edge set. We will often write $\mathcal{G}(\mathcal{V}, \mathcal{E})$. For an edge $(v_i, v_j) \in \mathcal{E}$ connecting two vertices $v_i \in \mathcal{V}$ and $v_j \in \mathcal{V}$, the first vertex v_i is its beginning vertex and the second vertex v_j is its end vertex.
- A path passing through the vertices $v_{r_0}, \ldots, v_{r_i}$ is denoted $p = v_{r_0} \to v_{r_1}, \ldots, \to v_{r_i}$ where for $j = 0, 1, \ldots, i-1$ and $(v_{r_j}, v_{r_{j+1}}) \in \mathcal{E}$.
- When a path p passes through the vertices v_i and v_j, the edge (v_i, v_j) is covered by the path p, we denote $(v_i, v_j) \in p$.
- Paths are disjoints if they have no common vertex.
- Paths are simple if they do not pass several times by the same vertex.
- $\mathcal{V}_1$ and $\mathcal{V}_2$ are two sets of vertices. A path p is called $\mathcal{V}_1 - \mathcal{V}_2$ path if its beginning vertex is in $\mathcal{V}_1$ and its end vertex is in $\mathcal{V}_2$. Moreover, if the only vertices of p belonging to $\mathcal{V}_1 \cup \mathcal{V}_2$ are its beginning vertices and end vertices, then p is called a $\mathcal{V}_1 - \mathcal{V}_2$ direct path.
- A set of ℓ $\mathcal{V}_1 - \mathcal{V}_2$ disjoint paths forms a $\mathcal{V}_1 - \mathcal{V}_2$ linking. A maximum linking is a linking consisting of a maximum number of paths.
- The set of vertices noted $V_{ess}(\mathcal{V}_1, \mathcal{V}_2)$ includes the vertices contained in all the maximum $\mathcal{V}_1 - \mathcal{V}_2$ linking. These vertices are called essential vertices in a maximum $\mathcal{V}_1 - \mathcal{V}_2$ linking.
- $Pred(V_1)$ is the set of all the vertices denoted v_i predecessors of $v_j \in V_1$, $i.e.$ $\forall v_j \in V_1$ there exist vertices v_i such as there is at least one $v_i - v_j$ path.

- $\varphi : \mathcal{E} \rightarrow \mathcal{F}(C)$, φ is a function from $\mathcal{E}$ to $\mathcal{F}(C)$ where $\mathcal{E}$ is a set of edges states and $\mathcal{F}(C)$ is a set of Boolean expressions of logical AND, OR and NOT based on the components events. This function φ is useful to affect to each edge state in the Boolean expression corresponding to the connectivity condition another Boolean expression based on the system components events (failures).
  For example, let us consider a system with two components c_1 and c_2 in parallel represented by the edge $(\mathbf{x}_1, \mathbf{x}_2)$ in the corresponding digraph. This edge is valid when c_1 or c_2 is not faulty *i.e.* $\overline{C}_1 \vee \overline{C}_2$ with $\overline{C}_i$ represents the failure of the component c_i.

4. MAIN RESULTS

The connectivity condition is a graphical condition which evaluates the connectivity between two sets of vertices. Several structural properties require the validity of this condition in addition to other graphical conditions. Indeed, to satisfy these structural properties, the connectivity condition should be verified systems. That is why the connectivity condition is interesting to study.

As explained in Section 2, we have to connect, in a digraph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, each element of a vertices set $V_1 \subseteq \mathcal{V}$ to at least one element of a vertices set $V_2 \subseteq \mathcal{V}$ where $V_1 \neq \emptyset$ and $V_2 \neq \emptyset$. The elements of V_1 and V_2 are defined according to the property studied. For instance, we have to ensure the connectivity between the state vertices and the output vertices for the state observability. To satisfy this condition, we provide the following approach.
Let us define two sets of vertices noted V_0 and V_T which are represented as follows:

$$\begin{cases} V_0 = \bigcup\limits_{v_i \in V_1} \left(V_{ess}\left(\{v_i\}, V_2\right) \setminus \{v_i\}\right) \\ V_T = V_1 \setminus (V_0 \cup V_2) \end{cases} \tag{1}$$

Using V_T, we will not handle all the vertices in V_1. All the elements of V_1 are connected to V_2 if the elements of V_T are connected to V_2. From this, we propose to write the connectivity condition from V_1 to V_2 as follows: $CC(V_1, V_2) = Exp(V_T, V_2)$ where $Exp(V_T, V_2)$ is a function which computes the Boolean expression corresponding to the connectivity condition between V_T and V_2. For the computation of $Exp(V_T, V_2)$, we propose a recursive formulation which provides a Boolean expression for the connectivity condition as following:

$$\begin{aligned} Exp(V_T, V_2) = & \bigvee_{p_j \in \mathcal{P}(\{v_i\}, V_2)} \Big(\big(\bigwedge_{(v_\ell, v_k) \in p_j} (v_\ell, v_k) \big) \\ & \wedge Exp(V_T \setminus \{v_m \text{ covered by } p_j\}, V_2) \Big) \end{aligned} \tag{2}$$

where $\mathcal{P}(\{v_i\}, V_2)$ is the set of all simple and direct $\{v_i\} - V_2$ paths which connect each element v_i of V_T to at least one element of V_2.
We choose to study the connectivity of any vertex v_i of V_T. The remaining vertices, which are not yet connected to V_2, are handled in the last part of Equation (2) *i.e.* $Exp(V_T \setminus \{v_m \text{ covered by } p_j\}, V_2)$ until having one element of V_T to handle. Note that the parameters of $Exp(V_T, V_2)$ change and it is never called with the same arguments. When we have only one element $\{v_i\}$ of V_T to connect to V_2, the corresponding expression is given below:

$$Exp(\{v_i\}, V_2) = \bigvee_{p_j \in \mathcal{P}(\{v_i\}, V_2)} \Big(\bigwedge_{(v_l, v_k) \in p_j} (v_l, v_k) \Big) \tag{3}$$

Obviously, if $v_i \in V_2$, v_i is always connected to V_2, then $Exp(\{v_i\}, V_2) = 1$ and if $v_i \notin Pred(V_2)$ then $Exp(\{v_i\}, V_2) = 0$.
The detail of the previous expressions $Exp(V_T, V_2)$ and $Exp(\{v_i\}, V_2)$ is given by Algorithms 1 and 2.

Algorithm 1 Computation of $CC(V_1, V_2)$

$V_0 \leftarrow \bigcup\limits_{v_i \in V_1} \left(V_{ess}\left(\{v_i\}, V_2\right) \setminus \{v_i\}\right)$
$V_T = V_1 \setminus (V_0 \cup V_2)$
$CC(V_1, V_2) \leftarrow Exp(V_T, V_2)$

Algorithm 2 Computation of $M \leftarrow Exp(V, V_2)$

if $V = \emptyset$ **then**
  $M \leftarrow 1$
else if $card(V) = 1$ **then** $\qquad \triangleright\ V = \{v_i\}$
  if $v_i \in V_2$ **then**
    $M \leftarrow 1$
  else if $v_i \notin Pred(V_2)$ **then**
    $M \leftarrow 0$
  else
    $M \leftarrow \bigvee\limits_{p_j \in \mathcal{P}(v_i, V_2)} \Big(\bigwedge\limits_{(v_l, v_k) \in p_j} (v_l, v_k) \Big)$
  end if
else if $card(V) > 1$ **then**
  Choose any $v_i \in V$
  if $v_i \in V_2$ **then**
    $M \leftarrow 1$
  else if $v_i \notin Pred(V_2)$ **then**
    $M \leftarrow 0$
  else
    $M \leftarrow 0$
    for $1 < j \leq card(\mathcal{P}(\{v_i\}, V_2))$ **do**
      if $(p_j \in \mathcal{P}(\{v_i\}, V_2) \wedge card(V \setminus \{v_m \text{ covered by } p_j\}) \geq 1)$ **then**
        $M \leftarrow M \vee \Big(\bigwedge\limits_{(v_\ell, v_k) \in p_j} (v_\ell, v_k) \Big) \wedge Exp(V \setminus \{v_m \text{ covered by } p_j\}, V_2)$
      else
        $M \leftarrow M \vee \Big(\bigwedge\limits_{(v_\ell, v_k) \in p_j} (v_\ell, v_k) \Big)$
      end if
    end for
  end if
end if
return M

Proposition 1: Let us consider a digraph $\mathcal{G}(\mathcal{V}, \mathcal{E})$. The connectivity condition between each element of $V_1 \subseteq \mathcal{V}$ to at least one element of $V_2 \subseteq \mathcal{V}$ is satisfied iff the Boolean expression $CC(V_1, V_2)$ computed according to Algorithms 1 and 2, and based on the edges validity is equal to 1.

Proof: First, let us prove that in order to connect the element of V_1 to V_2, it is necessary and sufficient to connect all the elements of V_T to V_2. Since $V_T \subseteq V_1$, connecting all the elements of V_1 to at least an element of V_2, implies that all the elements of V_T are connected to V_2. So, connecting all the elements of V_T is necessary to ensure the connectivity condition

$CC(V_1, V_2)$. Otherwise, we have that $V_1 = V_T \cup V_0$, according to the definition of V_0, we have that $v_j \in V_0$ implies that $\exists\, v_i \in V_T$ such as $v_j \in V_{ess}(\{v_i\}, V_2)$. Therefore, if v_i is connected to V_2, then v_j is also connected to V_2. Each element $v_j \in V_0$ is connected to V_2, if every element $v_i \in V_T$ is connected to V_2. So, it is necessary and sufficient to connect the elements of V_T to V_2 in order to connect the elements of V_1 to V_2.

Obviously, the connectivity of each element $v_\ell \in V_2$ is always verified. So, $\forall v_\ell \in V_2$, $Exp(v_\ell, V_2) = 1$. Otherwise, there is no path that connects the element v_m which are not predecessors of V_2 to V_2. Therefore, $\forall v_m \notin Pred(V_2)$, $Exp(\{v_m\}, V_2) = 0$.

The computation of $CC(V_1, V_2)$ is mainly based on the computation of $Exp(V, V_2)$ given in Algorithm 1 and 2 where $V \subseteq V_1$. When $card(V) = 1$, there is only one element v_i predecessor of V_2 in the set V. v_i can be connected to V_2 through a $\{v_i\} - V_2$ path in the paths set $\mathcal{P}(v_i, V_2)$. Therefore, Exp is a logical OR between the elements p_j of the paths set $\mathcal{P}(\{v_i\}, V_2)$ knowing that the path p_j is a logical AND between its edges (v_l, v_k). So,

$$Exp(\{v_i\}, V_2) = \bigvee_{p_j \in \mathcal{P}(\{v_i\}, V_2)} \Big(\bigwedge_{(v_l, v_k) \in p_j} (v_l, v_k) \Big).$$

Assume that $\forall n_0 \leq card(V)$, for all subsets $V_0^k \subseteq V$ of cardinality $\leq n_0$, $Exp(V_0^k, V_2)$ is the right expression for the validity of the connectivity condition *i.e.* all the elements in V_0^k are connected to V_2 if $Exp(V_0^k, V_2)$ is equal to "1". Let us prove that the Boolean expression Exp is true $\forall V_1^k \subseteq V$ of cardinality $n_0 + 1$. The corresponding expression Exp is:

$$Exp(V_1^k, V_2) = \bigvee_{p_j \in \mathcal{P}(\{v_i\}, V_2)} \Big(\Big(\bigwedge_{(v_\ell, v_k) \in p_j} (v_\ell, v_k) \Big) \wedge Exp(V \setminus \{v_m \text{ covered by } p_j\}, V_2) \Big).$$

p_j covers at least v_i and $V \setminus \{v_m \text{ covered by } p_j\}$ which is of cardinality $\leq n_0$, then $Exp(V \setminus \{v_m \text{ covered by } p_j\}, V_2)$ gives the correct expression according to the recurrence assumption because $card(V \setminus \{v_m \text{ covered by } p_j\}) \leq n_0$. If the path p_j is valid, then all the vertices covered by p_j are connected to V_2, therefore, we have to connect the remaining vertices which are not yet connected to V_2 using the expression: $Exp(V \setminus \{v_m \text{ covered by } p_j\}, V_2)$.
From this, the first part of the expression $Exp(V_1^k, V_2)$ is the right expression to connect v_i to V_2 through a path $p_j \in \mathcal{P}(\{v_i\}, V_2)$, and the second part is the right expression to connect the element not connected to V_2. Therefore, for all $V_1^k \subseteq V$ of cardinality $n_0 + 1$, $Exp(V_1^k, V_2)$ is correct.

We proved that if the expression $Exp(V, V_2)$ is correct for a vertices set V of cardinality n_0, then it is correct for V of cardinality n_0+1. We also proved that $Exp(V, V_2)$ is correct for $V = \{v_i\}$ where the cardinality of V is $n_0 = 1$. From this, the expression is correct for the vertices set V of cardinality 2 and so on. So the expression $Exp(V, V_2)$ is correct for any vertices set V of cardinality n, therefore, for the vertices set V_1. $\triangle$

Example: In order to illustrate the proposed method, let us consider an example of system represented by its digraph in Figure 1. This figure represents a system which is quite simple to better understand the aim of the provided approach.
For this example, we are interested in the structural state observability of this system. In this case, we have to connect each element x_i of the state vertices to at least one output y_i. Therefore, $V_1 = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5\}$, $V_2 = \{\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3\}$, $V_0 = \{\mathbf{x}_5\}$ and $V_T = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4\}$.

Let us give the Boolean expression $Exp(V_T, V_2)$ starting in

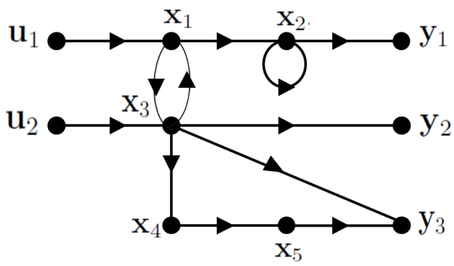

Figure 1. The digraph of example

connecting the vertex $\mathbf{x}_1$ to V_2 as follows:

$$\begin{aligned} Exp(V_T, V_2) = & ((\mathbf{x}_1, \mathbf{x}_2) \wedge (\mathbf{x}_2, \mathbf{y}_1) \wedge Exp(\{\mathbf{x}_3, \mathbf{x}_4\}, V_2)) \\ & \vee((\mathbf{x}_1, \mathbf{x}_3) \wedge ((\mathbf{x}_3, \mathbf{y}_2) \vee (\mathbf{x}_3, \mathbf{y}_3)) \wedge Exp(\{\mathbf{x}_2, \mathbf{x}_4\}, V_2)) \\ & \vee((\mathbf{x}_1, \mathbf{x}_3) \wedge (\mathbf{x}_3, \mathbf{x}_4) \wedge (\mathbf{x}_4, \mathbf{x}_5) \wedge (\mathbf{x}_5, \mathbf{y}_3) \wedge Exp(\{\mathbf{x}_2\}, V_2)). \end{aligned}$$

and so on for the computation of $Exp(\{\mathbf{x}_3, \mathbf{x}_4\}, V_2)$, $Exp(\{\mathbf{x}_2, \mathbf{x}_4\}, V_2)$ and $Exp(\{\mathbf{x}_2\}, V_2)$.

From this, the connectivity is ensured between V_1 and V_2 iff the Boolean expression given by Equation (4) is equal to "1".

$$\begin{aligned} CC(V_1, V_2) = & \big((\mathbf{x}_2, \mathbf{y}_1) \wedge (\mathbf{x}_4, \mathbf{x}_5) \wedge (\mathbf{x}_5, \mathbf{y}_3) \wedge \\ & ((\mathbf{x}_1, \mathbf{x}_2) \vee (\mathbf{x}_1, \mathbf{x}_3)) \wedge \big((\mathbf{x}_3, \mathbf{y}_2) \vee (\mathbf{x}_3, \mathbf{y}_3) \vee (\mathbf{x}_3, \mathbf{x}_4)\big)\big) \\ & \vee\big((\mathbf{x}_1, \mathbf{x}_2) \wedge (\mathbf{x}_2, \mathbf{y}_1) \wedge (\mathbf{x}_3, \mathbf{x}_1) \wedge (\mathbf{x}_4, \mathbf{x}_5) \wedge (\mathbf{x}_5, \mathbf{y}_3)\big) \end{aligned} \tag{4}$$

5. RELIABILITY OF THE CONNECTIVITY CONDITION

As automatic control is involved in many applications, extending the notion of reliability to structural properties of systems is an important issue. It allows anticipating failures, loss of properties and consequences of failures on the mission assigned of the system. As structural properties rely on basic conditions, computing the reliability of these elementary conditions based on the reliability of the involved components is required. In the recent literature [Weber et al., 2012, Pourret et al., 2008], there is a growing interest to model the reliability of complex industrial systems using Bayesian Networks (BN). This modeling method seems to be very relevant in the context of complex systems [Langseth, 2008]. Even if the system proposed in this paper is of low size and complexity, the purpose is to address large scale systems. So, computing the reliability of a complex condition is challenging.

We propose to use Dynamic Bayesian Networks (DBN) to compute the reliability of the connectivity condition. BN perform the factorization of variables joint distribution based on the conditional (in)dependencies. The principles of this modeling tool are deeply explained in [Jensen, 1996, Pearl, 1988]. Let us define some basic notions of BN. BNs are Directed Acyclic Graphs (DAG) used to represent uncertain knowledge. Those graphs are distinct from the directed graphs $\mathcal{G}(\mathcal{V}, \mathcal{E})$ representing the system state space model. DAG is defined as a couple:

$DAG = ((N, A), P)$, where (N, A) represents the graph; N is a set of nodes; A is a set of arcs and P represents the set of probability distributions that are associated to each node. When a node is not a root node, *i.e.* when it has some parent nodes, the probability distribution is a conditional probability distribution that quantifies the probabilistic dependency between this node and its parents. A discrete random variable Z is represented by a node $Z \in N$ with a finite number of mutually exclusive states $S_Z : \{s_1^Z, \ldots s_M^Z\}$. The vector $P(Z)$ denotes a probability distribution over these states as Equation (5):

$$P(Z) = \left[P(Z = s_1^Z) \ldots P(Z = s_m^Z) \ldots P(Z = s_M^Z)\right] \quad (5)$$

with $P(Z = s_m^Z) \geq 0$ and $\sum_{m=1}^{M} P(Z = s_m^Z) = 1$ and where $P(Z = s_m^Z)$ is the marginal probability of node Z being in state s_m^Z.

A DBN is a BN taking into account the temporal dimension. At each time step k ($k \geq 0$), a variable X is represented by a node X_k. Thus, each time step k is represented by a set of nodes N_k including all the variables of this time slice k. The qualitative dependency between a node X_k and a node Y_{k+1} is represented by a directed arc linking the two nodes. In our problem the nodes X_k and Y_{k+1} represent the same variable. This arc, denoting a transition function, is defined by a conditional probability table (CPT) as follows:

$$\mathrm{F}[X_k](Y_{k+1}) = \begin{bmatrix} f[A_1^{X_k}](A_1^{Y_{k+1}}) & \ldots & f[A_1^{X_k}](A_{Q_Y}^{Y_{k+1}}) \\ \ldots & \ldots & \ldots \\ f[A_{Q_X}^{X_k}](A_1^{Y_{k+1}}) & \ldots & f[A_{Q_X}^{X_k}](A_{Q_Y}^{Y_{k+1}}) \end{bmatrix} \quad (6)$$

where $A_i^{X_k}$ is the i-th state of X_k and $A_j^{Y_{k+1}}$ is the j-th state of Y_{k+1}.

DBN are supposed to be:

- Stationary: $F[X_k](Y_{k+1})$ does not depend on k.
- Markovian: $F(Y_{k+1})$ depends only on the distributions of its parent nodes. Thus, the future time step is conditionally independent of the past given the present time slice [Murphy, 2002].

To model reliability, the root nodes represent the basic components reliability involved in the reliability of the connectivity condition. The nodes connected by the temporal arcs represent the components reliability at two consecutive steps.

As developed in Section 4, we consider that the connectivity condition is expressed as a Boolean expression from a combination of logical $\wedge$ and $\vee$ linking the edges state. Indeed, in example of Section 4 the connectivity condition given by Equation (4) as Boolean expression based on the digraph edges can be written in another way from a logical combination of the events (failures) that can occur on the system components. We recall that φ is a function which associates to each edge state in the Boolean expression an expression of events on the system components. For the system provided in the previous example, function φ is illustrated by Table 1.

Therefore, using function φ and Table 1, Equation (4) can be written as given by Equation (7). We recall that $\overline{C_i}$ characterizes the good operating of the component c_i (no event on c_i).

$Edgestate_i$	C_i	$Edgestate_i$	C_i
$(\mathbf{x_1}, \mathbf{x_2})$	$\overline{C_1} \wedge \overline{C_2}$	$(\mathbf{x_3}, \mathbf{y_3})$	$\overline{C_5}$
$(\mathbf{x_2}, \mathbf{y_1})$	$\overline{C_2}$	$(\mathbf{x_3}, \mathbf{x_4})$	$\overline{C_6} \vee \overline{C_3}$
$(\mathbf{x_1}, \mathbf{x_3})$	$\overline{C_1}$	$(\mathbf{x_4}, \mathbf{x_5})$	$\overline{C_7}$
$(\mathbf{x_3}, \mathbf{x_1})$	$\overline{C_3}$	$(\mathbf{x_5}, \mathbf{y_3})$	$\overline{C_8}$
$(\mathbf{x_3}, \mathbf{y_2})$	$\overline{C_4}$		

Table 1. Edges and corresponding components events logical expression

$$CC(V_1, V_2) = \overline{C_1} \wedge \overline{C_2} \wedge \overline{C_7} \wedge \overline{C_8} \wedge (\overline{C_3} \vee \overline{C_4} \vee \overline{C_5} \vee \overline{C_6}) \quad (7)$$

Any Boolean expression can be handled in a probabilistic way by a BN through corresponding CPT. Table 2 defines the CPT for $\wedge$ and $\vee$ for 2 inputs where "Up" de is the good operating state and "$Down$" is the failed state according to the binary state assumption.

A	B	$C = A \wedge B$ Up	$C = A \vee B$ Up
Up	Up	1	1
Up	$Down$	0	1
$Down$	Up	0	1
$Down$	$Down$	0	0

Table 2. CPT for $\wedge$ and $\vee$ logical operators

Table 2 contains only the Up state for $\wedge$ and $\vee$ and, obviously, $Down$ is the opposite state.

We assume that each components c_i has a constant failure rate λ_i. The components failure rates are considered different from each other and equal to $\lambda_i = i \cdot 10^{-3}$ (per hour). Then, the reliability follows an exponential distribution. To model such reliability distribution in a DBN, the CPT given in Table 3 is defined. Figure 2 represents the DBN model of the connectivity condition. The dynamic aspect of the DBN concerns the way the components may fail, thus affecting the connectivity. The top node CC contains the probability distribution of the connectivity condition.

$X(k)$	$X(k+1)$ Up	$Down$
Up	$e^{-\lambda_i . t}$	1-$e^{-\lambda_i . t}$
$Down$	0	1

Table 3. Temporal CPT

Figure 3 shows the reliability evolution of the provided connectivity condition function depending on time. We can notice from the this figure that after 200 hours of operation, there is a low probability (less than 0.1) that the connectivity condition is still verified.

6. CONCLUSION AND PERSPECTIVES

This paper deals with the connectivity condition which is compliant with many structural properties in structured systems. The connectivity condition is studied in a generic way, by proposing a method which defines a Boolean expression based on the edges state that reflect the operating state of all the system components. This Boolean expression must be valid to keep the connectivity condition verified. Thus, the Boolean expression allows us to compute the reliability of the connectivity condition knowing the reliability of the corresponding components. In this paper, we used two kinds of graphs, a directed graph $\mathcal{G}$ representing the state space model of the system

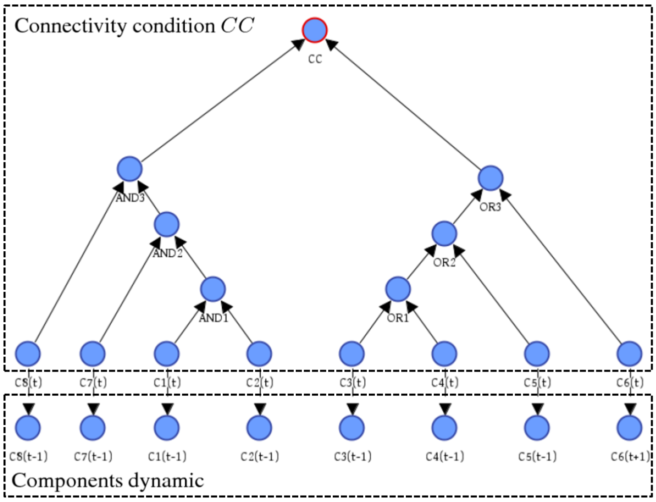

Figure 2. DBN of the connectivity condition

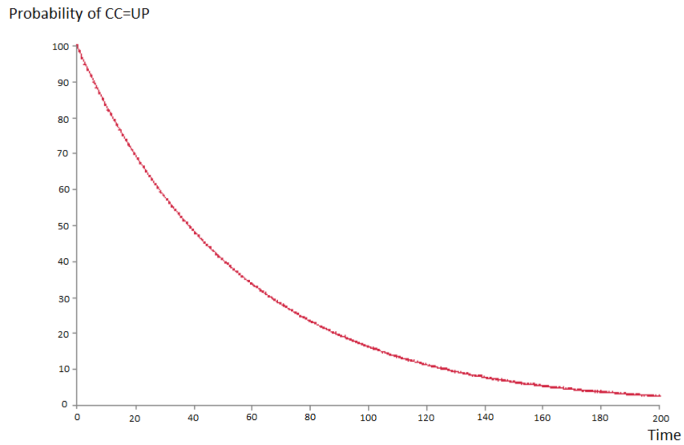

Figure 3. Reliability of the connectivity condition

and the relationship between its variables, and directed acyclic graphs DAG as a tool to compute the connectivity condition reliability. An example has been studied to show the application of the proposed methodology.

As a complementary study to the proposed work, other graphical conditions (complete matching, linking, distance) can be developed in the same way, in order to evaluate all the structural properties for several kinds of systems.

REFERENCES

T. Boukhobza. Partial state and input observability recovering by additional sensor implementation: a graph-theoretic approach. *International Journal of Systems Science*, 41(11): 1281 – 1291, 2010.

T. Boukhobza and F. Hamelin. Observability of switching structured linear systems with unknown input: a graph-theoretic approach. *Automatica*, 47(2):395–402, 2011a.

T. Boukhobza and F. Hamelin. Observability analysis and sensor location study for structured linear systems in descriptor form with unknown inputs. *Automatica*, 47(12):2678–2683, 2011b.

C. Commault and J. M. Dion. Sensor location for diagnosis in linear Systems: a structural analysis. *IEEE Transactions on Automatic Control*, 52(2):155–169, 2007.

B. Conrard, V. Cocquempot, and S. Mili. Fault-Tolerant system design in multiple operating modes using a structural model. In Grall Bérenguer and Guedes-Soares, editors, *Advances in Safety, Reliability and Risk Management*, 2012.

J. M. Dion, C. Commault, and J. W. van der Woude. Generic properties and control of linear structured systems: a survey. *Automatica*, 39(7):1125–1144, 2003.

O. Doguc and J.E. Ramirez-Marquez. A generic method for estimating system reliability using bayesian networks. *Reliability Engineering and System Safety*, 94:542–550, 2009.

F.V. Jensen. *An Introduction to Bayesian Networks*. London, 1996.

A. Kaufmann, D. Grouchko, and R. Cruon. *Mathematical models for the study of the reliability of systems*. New York : Academic Press, 1977.

H. Langseth. *Bayesian Network in Reliability: The Good, the Bad and the Ugly. Advances in Mathematical Modeling for Reliability*. 2008.

H. Langseth and L. Portinale. Bayesian networks in reliability. *Reliability Engineering and System Safety*, 92:92–108, 2007.

C. T. Lin. Structural controllability. *IEEE Transactions on Automatic Control*, AC-19(3):201–208, 1974.

S. Maza, C. Simon, and T. Boukhobza. The impact of actuator failures on the structural controllability of linear systems : A graph theoretical approach. *I.E.T. Control Theory & Applications*, 6:412–419, 2012.

K. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, Dept. Computer Science. UC, Berkeley, 2002.

J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988. ISBN 0-934613-73-7.

O. Pourret, P. Nam, and B Marcot, editors. *Bayesian Networks: A Practical Guide to Applications*. Wiley & Sons, 2008.

M. Staroswiecki. *Observability and the Design of Fault Tolerant Estimation Using Structural Analysis*, chapter 353, pages 257–278. Advances in Control Theory and Application. Springer, 2006.

A. Villemeur. *Reliability, availability, maintainability and safety assessment: methods and techniques*. from French Edition by Cartier A., T. & M.C, L, 1992.

P. Weber, G. Medina-Oliva, C. Simon, and B. Iung. Overview on bayesian networks applications for dependability, risk analysis and maintenance areas. *Engineering Applications of Artificial Intelligence*, 25, 2012.

J. L. Willems. Structural controllability and observability. *System & Control Letters*, 8(1):5–12, 1986.