Monitoring of a Milk Manufacturing Workshop Using Chronicle and Fault Tree Approaches

Anis MHALLA[1,2], Simon COLLART DUTILLEUL[2], Etienne CRAYE[2], Mohamed BENREJEB[1]

[1] LAboratoire de Recherche en Automatique,
Ecole Nationale d'Ingénieurs de Tunis, BP 37,
le Belvédère, 1002 Tunis, Tunisie
anis.mhalla@enim.rnu.tn, mohamed.benrejeb@enit.rnu.tn

[2] Laboratoire d'Automatique, Génie Informatique et Signal
Ecole Centrale de Lille, Cité Scientifique BP 48,
59651 Villeneuve d'Ascq, France
simon.collart_dutilleul@ec-lille.fr, etienne.craye@ec-lille.fr,

Abstract: Developments presented in this paper are devoted to the monitoring of manufacturing job-shops with time constraints and with assembling tasks. A new method for monitoring combining chronicles with fault tree analysis is proposed. The purpose of the proposed approach is to explain in details what is happening on the system and to help operators identifying failures in order to avoid a damage of the process or an accident with human beings. Finally, to demonstrate the effectiveness and accuracy of the monitoring approach, an application to a milk production unit is outlined. The results show that the monitoring approach allows keeping on producing, by on-line diagnosis, while providing correct quality of the manufactured products.

Keywords: milk manufacturing unit, monitoring, chronicle, fault tree, diagnosis, time constraints

1. Introduction

Supervision and monitoring are a set of solutions allowing managing the process in order to correctly react in case of failure. The problem we deal with is the supervision of complex discrete event systems such as telecommunication networks, electricity distribution networks and manufacturing workshops. The Manufacturing systems can also be subject to staying time constraints (maximum cooking time of a product in a furnace, overheating of milk bottles in a hydromat …). This type of constraints does not only affect the performances of the system but also its functional validity (burnt pieces, unusable milk...). Hence, we need a specific performance evaluation, because an "as-soon-as-possible" operation is invalid in the general case.

Two classical approaches in monitoring such systems are knowledge based techniques that directly associate a diagnosis to a set of symptoms, for example expert systems [1], or chronicle recognition systems [2], and model-based techniques which rely on a behavioural model of the system [3]. The main weakness of the second approach is the difficulty to represent the behavioral model of complex systems. Therefore, we focus on expertise-based approaches which are known to be better suited to that kind of system than model-based techniques.

In this paper, we propose a new method for monitoring manufacturing workshops with time constraints which combine chronicles with fault tree analysis. The monitoring system is used as a model for diagnosis. The proposed approach allows to roughly identify what is happening on the system (failed and critical components) and to help operators to identify failures in order to avoid damage to the equipment or an accident with human beings.

A fault tree describes how a set of events can concur in order to cause a top event. However, to address the problem of detection, we complement those fault trees with a dynamic model of system behaviour. This system can capture the behavioural transformations that occur in complex systems as a hierarchy of state machines, by chronicle recognition. The chronicle, proposed in [4], describes a situation that is worth identifying within the diagnosis context. It is made up of a set of events and temporal constraints between those events. As a consequence, this formalism fits particularly well problems that have a temporal dimension. Then, monitoring the system consists of analyzing flows of events and recognizing constraints composing the chronicles.

The work has been motivated by an application that aims at monitoring the behaviour of a milk manufacturing workshop. In such system, each operation is associated to a time interval. Its lower bound indicates the minimum time needed to execute the operation. The upper bound sets the maximum time to not exceed otherwise the quality of the product is deteriorated.

This paper is organised as follows. In Section 2 we discuss the form of the monitoring model and its development process and we recall the principles of the chronicle recognition approach. A more detailed description of this distributed approach is given in [4], [5] and [14]. Section 3 begins by presenting the milk production workshop. The monitoring approach of manufacturing systems is considered. An illustrative example is outlined and the results are discussed. Finally, conclusions of this work are given.

2. Monitoring Task

2.1 Modelling

In order to help the human agent (or supervisor) in charge of managing a manufacturing workshop, i.e., detecting failures and deciding reconfiguration/repair actions [6], a monitoring task is needed. The purpose of the monitoring task is to detect, localise, and identify problems that occur on the system. These problems can be physical (an equipment is down, a cable is cut) or logical (a station is rebooting, a logical connection is down). Our purpose is to explain in details what is happening on the system and what a supervisor agent needs to know in order to avoid damage to the equipment or an accident with human beings.

Several works deal with the supervision of manufacturing job-shops with maximum time constraints [7], [8]. For example, [7] proposed a filtering mechanism of sensors signals integrating the robustness values. This mechanism aims at generating symptoms for the diagnosis. It allows avoiding control freezing if the time disturbance is in the robustness intervals. Therefore, the filtering mechanism allows detecting a failure symptom. When a symptom of an abnormal functioning is claimed by the filtering mechanism, a series of algorithms are generated in order to build a theory dealing with localization problem.

The general form of the proposed monitoring model is shown in Figure 1. Three models can be distinguished:

- Structural model
- Distributed detection model
- Behavioural model

A/ Structural model

The spine of the monitoring model is a hierarchy of abstract state-machines, which is developed around the structural decomposition of the system. The structural part of that model records the physical decomposition of the system into basic blocks (left hand side of Figure 1). This part of the model shows the topology of the system in terms of components and connections.

B/ Distributed detection model

The aim of the detection function is to recognize, in a distributed way, specific evolutions of the monitored process. These evolutions are represented by means of chronicles. A chronicle [9] is defined like a set of events and a set of time constraints between these events.

In a manufacturing system, every set of sensors providing useful information must have its own monitoring system. When these systems are connected, we have a distributed monitoring system. The detection architecture, Figure 1, is subdivided into monitoring systems (sites) S_i. Every site (diagnoser) manages a physical zone (set of sensors and resources) of the manufacturing system. The multiple covering of the manufacturing system zone, by different monitoring systems, facilitates the failure diagnosis.

A chronicle is distributed into several sub-chronicles associated to the different monitoring sites. Each site (diagnoser) performs locally a partial observation of the system and communicates with other diagnosers in order to get necessary information not available locally or to take decisions relative to the diagnosis with distributed way. Therefore, the detection

function of each site has to recognize an evolution of the monitored process using a set of sub-chronicles. The recognition is performed through the verification of the time constraint associated to a considered sub-chronicle.

C/ Behavioural model

The dynamic part of the monitoring model is a hierarchy of state machines that determine the behaviour of the system and its subsystems (right hand side of Figure 1). The model identifies the abnormal functional states of the system and its subsystems by chronicle recognition. Therefore, the distributed detection function monitors the system evolutions through the recognition of chronicles. If the chronicle is recognized, a diagnosis text describing the recognized situation is generated.

The chronicle recognition consists of checking that all the constraints of the chronicle

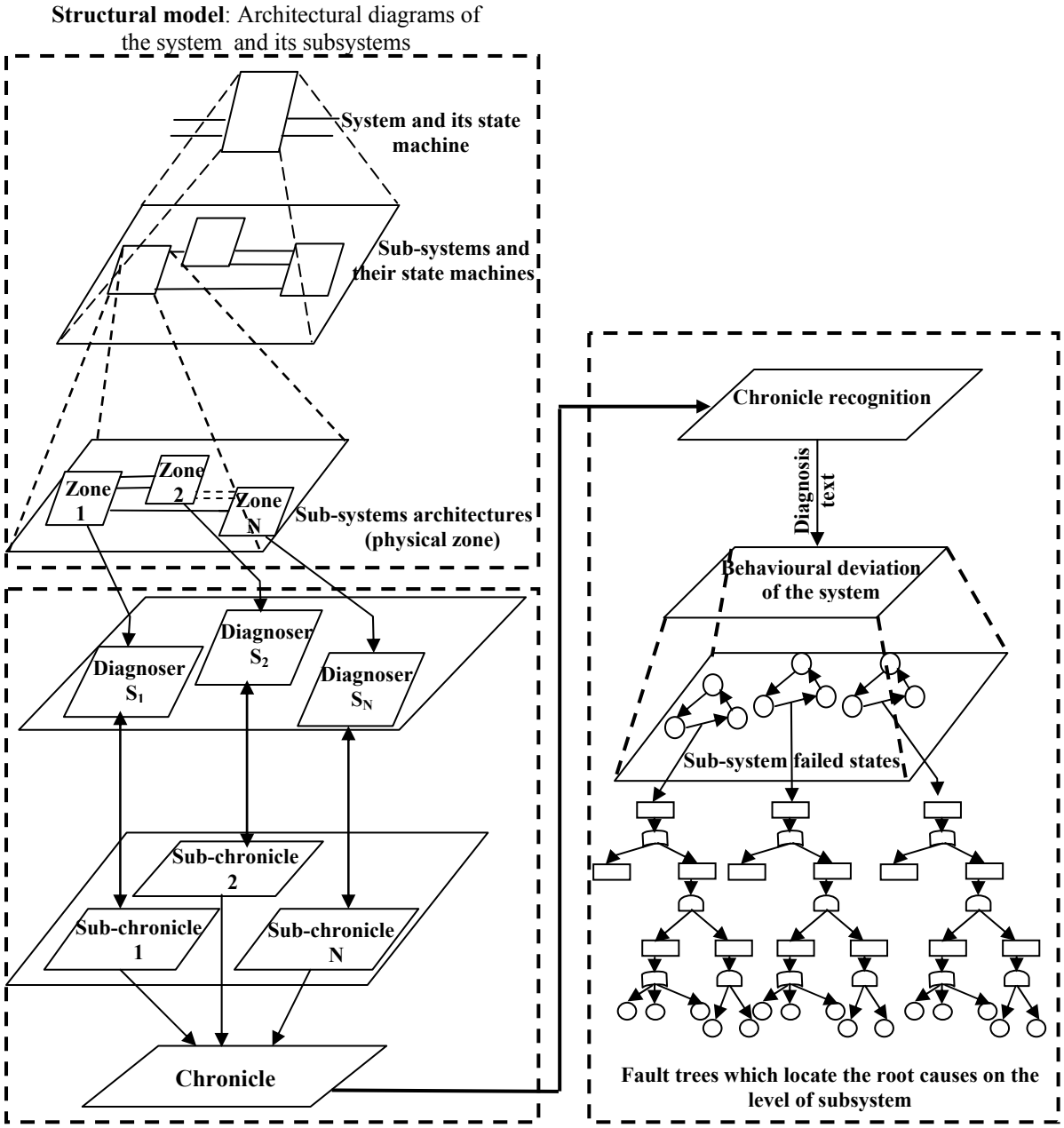

Figure 1. General form of monitoring model

are satisfied according to the durations between the occurrences of the associated events [10]. If the chronicles represent normal evolution of the monitored system, the no recognition of the chronicle allows pointing out a behavioural deviation of the system, the detection function based on this discrepancy principle can then detect a failure symptom. If the monitored time constraints are associated to erroneous evolutions of the process, the chronicle recognition leads to failure symptom detection.

As Figure 1 shows, the lower layers of the behavioural model identify abnormal functional states, in other words states where the subsystems deviate from their expected normal behaviour. As we progressively move from the leaf nodes towards the higher layers of the behavioural model, the model shows how logical combinations or sequences of lower-level subsystem failures propagate upwards and cause functional failure at higher levels of the behavioural model.

Abnormal functional states at the low-levels of the design (behavioural model) represent the top events of fault trees which record the causes and propagation of failure through the architectures of the corresponding subsystems. Classical fault tree analysis techniques can be used to derive those fault trees.

2.2 Monitoring process

The monitoring process is performed in two stages. The first stage is the fault detection, which gathers the information required about the causes and consequences of failures in system sections. This information is stored and used in the second stage (identification stage) of application in order to identify component failures that cause observed symptoms. This paper considers the diagnostic application of the fault tree and chronicle methods. Details of the fundamentals of each procedure are stated.

Application of these steps to a milk manufacturing unit is detailed in Section 3.

2.2.1 Fault detection and identification process

A/ Fault detection method

Typically, a diagnosis system, Figure 2, is a process of symptom pattern recognition. A good real-time diagnosis system would be the one which can [11]:

- model complex symptom patterns,
- easily generate a symptom pattern library,
- recognize the symptom patterns matching the observed situation.

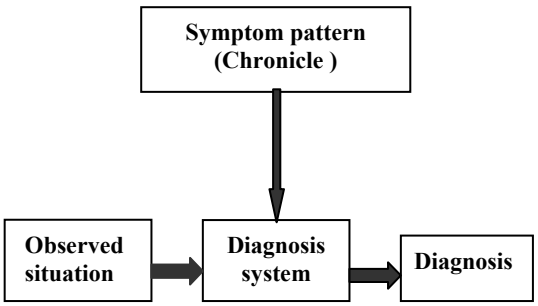

Figure 2. Diagnosis system [4]

A symptom pattern is a scenario of events occurring in a specific topologic state in a given time period. The format needed to model a temporal scenario is called a "chronicle".

A-1/ Chronicle syntax

This section presents the chronicle formalism. A chronicle is composed of a set of events, a set of temporal constraints linking a pair of events, and the diagnosis text describing the recognized situation [11].

Basic definition

Preliminary definitions, useful for the rest of this paper, are given in order to explain the distributed detection principles.

Definition 1[12]: An event is a stimulus to which the system can react by a state change. An event can occur after a message sent by the process at the beginning or at the end of any operation. There are two types of events:

- **Mandatory events**. Mandatory events are the main events of a chronicle. The arrival of all the mandatory events is necessary for recognizing chronicles.
- **Forbidden events**. In case of forbidden event occurrence during the time window corresponding to the time constraints, the chronicle will not be recognized.

Definition 2 [13]: An occurrence date is the time corresponding to an event issued from the process. Let O be the occurrence function

which associates to each event "e_i" its occurrence date $O(e_i)$, then:

$$O : E \rightarrow Q^+$$
$$e_i \rightarrow O(e_i)$$

Where, Q^+ is the set of positive rational numbers and E is the set of events ($e_i \in E$).

Definition 3 [13]: A constraint is a relationship expressed by a duration between events occurrences. Two types of constraints can be distinguished:

- *local constraints* link events dated by a same site,
- *global constraints* link events dated by different diagnosers.

A-2/ Time constraint verification with delay consideration

The problem induced by the chronicle recognition is the verification of a global constraint.

Let us denote by $\Delta \in [\delta_m, \delta_M]$; where δ_m , $\delta_M \in Q^+$) are the delay bounds existing between the different sites. The problem to be solved can be summarized as follows [13], [14]:

Given:

- the global constraint $C_{B,A}$ linking e_B to e_A,
- the occurrence dates of e_B and e_k on the site S_B,
- the bounds δ_m and δ_M of the transmission delay from S_A to S_B.

Is it possible to establish if the occurrence date $O(e_A)$ satisfies the global constraint $C_{B,A}$?

We have:

$$d_{B,A} \leq O(e_B) - O(e_A) \leq f_{B,A} \quad (1)$$

and

$$O(e_B) - O(e_k) + O(e_k) - O(e_A) = \Phi + \Delta.$$

As $\delta_m \leq \Delta \leq \delta_M$, we obtain:

$$\Phi + \delta_m \leq O(e_B) - O(e_A) \leq \Phi + \delta_M \quad (2)$$

with $\Phi \in]-\infty, +\infty[$.

The verification of the interval constraint consists, by means of the measurable duration Φ, of looking for the durations $O(e_B) - O(e_A)$ that verify both [12]:

$$\begin{cases} d_{B,A} \leq O(e_B) - O(e_A) \leq f_{B,A} \\ \Phi + \delta_m \leq O(e_B) - O(e_A) \leq \Phi + \delta_M \end{cases}$$

In order to quantify the set of possible durations $\Phi \in \Re$ ($\Re$ is the universe of discourse), a graphical representation of real values allowing to verify the constraint $C_{B,A}$ is proposed (Figure 3) [14].

Considering a bounded delay, the possibility to check a time constraint belongs to [0, 1]. Consequently the verification of time constraints is fuzzy. These results make it possible to highlight zones of certainty for the detection function.

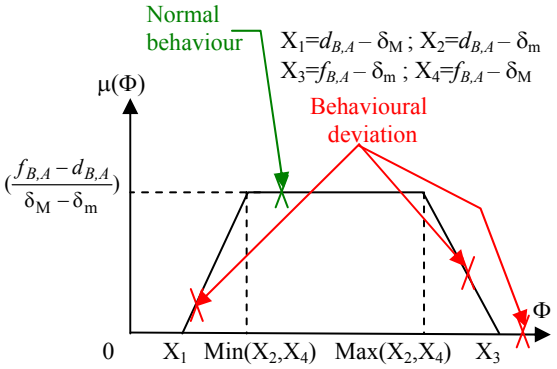

Figure 3. Possibility function for an interval constraint with $X_1 \geq 0$, $X_2 \geq 0$, $X_3 \geq 0$ and $X_4 \geq 0$

A-3/ Distributed failure detection method

Using the chronicle method for distributed recognition of particular evolutions in a manufacturing system, involves the following steps (modeling and preparation stage) [14]:

Step 1 — Sub-system identification: the first task is to divide the system up into sections (sub-systems). There are physical zones that can change the system behaviour through the occurrence of component failures.

Step 2 — Sensor identification: the monitoring architecture is subdivided into several monitoring systems (sites). Each site manages a physical zone (set of sensors and resources) of the manufacturing system. The redundant covering of the manufacturing system zone by different monitoring systems facilitates the failure diagnosis.

Step 3 —Distributed detection function: the distributed detection consists in recognizing a deviation from the normal (expected)

functioning. It monitors the system evolutions through the recognition of chronicles, constituted by specific sequences of events linked by time constraints.

Step 4 — Scenario formation: all system scenarios are developed. The scenarios consider all possible deviations in each section. Deviations can occur due to the failure of components. In fact, the approach of the detection function consists in enumerating a set of scenarios (events sequences) bringing the system to erroneous situations (failure). A finishing scenario indicates a failure situation. Therefore, one or several scenarios that lead to the same failure situation are called a chronicle.

2.2.2 Fault tree diagnostic method

Once the chronicle is recognized, the diagnosis text is generated. The diagnosis text determines failed states (the deviations from the normal function) of the system and its subsystem. To establish the causality of failures on the sub-systems that can affect the system status, fault trees are constructed.

Fault tree analysis has been around as a reliability assessment technique since the 1970's. It is concerned with the analysis of failures and provides a diagrammatic description of the various causes of a specified system failure in terms of the failure of its components [15]. It is commonly used to evaluate the reliability of complicated systems in many fields, such as nuclear plants [16], chemical works, manufacturing industry, pipelines [17], and control systems [18].

The instrument of fault trees has provides some improvement because of their fast building and calculation. A summary of the most important papers on the subject can be found in [19]. A generalization of the concept of fault tree is presented in [20].

Using the fault tree method for fault diagnostics involves the following step (identification of components failure):

Step 5 — Constructing fault trees for observable system deviations:

For each of sub-sections, fault trees are constructed. The behaviour of the system can be monitored by sensors located at specific points. Fault trees are constructed to represent the failure modes at these locations. These failure logic diagrams are developed down to section component failures, deviations in process variables that are inputs to the section, and operating modes of the system.

3. Monitoring of Milk Manufacturing Workshop

3.1 Presentation of the workshop

Figure 4, shows a milk manufacturing unit composed of five machines (M_1, M_2, M_3, M_4, M_5) and six conveyors (T_1, T_2,T_3, T_4, T_5, T_6), where [14]:

- M_1 is a bottle filling machine,
- M_2 is a milk bottle capper,
- M_3 is a time/date stamp,
- M_4 is a labelling machine,
- M_5 is a packaging machine.

For simplicity, we disregard the nature of the precise operations performed in the milk production unit, and therefore we represent a model of a generic workshop.

To manufacture the products (bottles of 1000 ml), empty bottles are placed on the conveyor T_1 to supply the bottle filling machine M_1. The filled bottles are transported towards the capping machine M_2 by the conveyor T_2. After capping, the bottles arrive directly on T_3. This conveyor carries the bottles to the machine M_3 (time/date stamp) to print the manufacturing date and end date of consumption.

Once this task is completed, the bottles move towards the labelling machine M_4 via the conveyor T_4. After, the bottles are transferred to the packaging machine M_5, where they will be wrapped by welding in a group of 6. Lastly, the finished products are deposited on the conveyor T_6 towards the stock of finished products SA.

3.2 Monitoring of milk manufacturing using chronicles and fault tree

Steps 1—sub-system identification

The milk production unit has been split into five sections as shown in Figure 4. The overall aim of the system is to control that the

production cycle of the bottles proceeded well. Five sections can be distinguished [14]:

Section 1 : bottle filling machine and conveyor T_1.
Section 2 : milk bottle capper and conveyorT_2.
Section 3 : time/date stamp machine and conveyor T_3.
Section 4 : labelling machine and conveyor T_4.
Section 5 : packaging machine and two conveyors T_5 and T_6.

Steps 2— Sensor identification

Within the framework of the distributed detection, we are interested exclusively in the sensors which are at the origin of the information necessary to the monitoring mechanisms.

Each diagnoser S_i (Site), Figure 4, has a partial observation of the monitored system, since each diagnoser is associated to a subset of process components and to related subset of observable events. The diagnoser monitors the evolutions through the flow of observable events generated by the subsystem.

deteriorated. Consequently, the detection monitors the system evolutions through the verification of time constraints.

Figure 4, gives operating, transfer durations and operating delay Δi associated to each operation (filling, capping, transfer...); these durations represent interval constraints.

When operating and transfer durations are included in the mentioned intervals, the production cycle of the bottles proceeded well. When the interval constraints are exceeded, it is possible that the time granted to the manufacture of a bottle has been delayed; consequently a failure symptom is detected. We are going to express that in the following.

Steps 3— Distributed detection function

The distributed detection of time constraints violation can be applied to the monitoring of systems whose constraints between events occurrences are comparable to communication delays, as for example the computer network systems, telecommunication networks, manufacturing system [4], [21] [22].

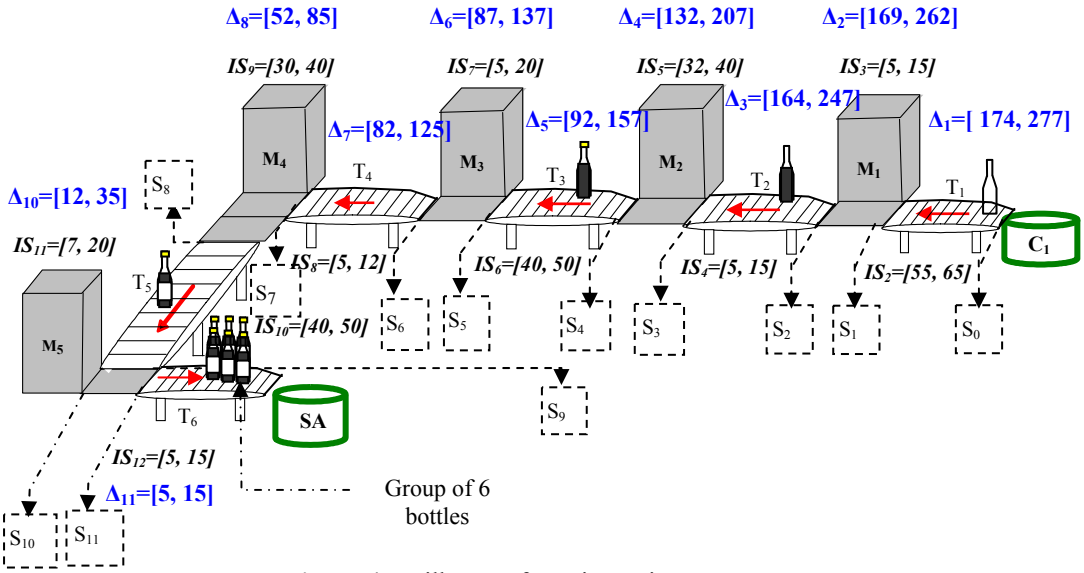

Figure 4. Milk manufacturing unit

In manufacturing workshops with time constraints, a time interval is associated to each operation ($[a_i, b_i]$ with *u.t*: unit time). Its lower bound indicates the minimum time needed to execute the operation and the upper bound fixes the maximum time to not exceed otherwise the quality of product is

When the checked constraints are associated with a normal behaviour of the monitored system, a high value corresponds to the expectation of a normal behavior. On the contrary, a low value implies the possibility of failure (behavioral deviation).

The monitoring of the milk production unit is done by observing the set of events. Therefore, we associate a diagnoser to each event and the detection function is distributed on these different sites:

e_{C1} : Beginning of containers transfer to the bottle filling machine ″M_1″(diagnoser S_0).

$e_{C1\rightarrow M1}$: End of transfer of the containers of the cyclone "C_1" to the bottle filling machine ″M_1″(diagnoser S_1).

e_{M1} : End of filling operation (diagnoser S_2).

$?_{M1\rightarrow M2}$: End of transfer of the bottles from the bottle-filling machine ″M_1″ to the Milk bottle capper ″M_2″ (diagnoser S_3).

e_{M2} : End of capping operation (diagnoser S_4).

$e_{M2\rightarrow M3}$: End of transfer of the bottles from the capper ″M_2″ to the time/date stamp ″M_3″ (diagnoser S_5).

e_{M3} : End of pointing operation (diagnoser S_6).

$e_{M3\rightarrow M4}$: End of transfer of the bottles from time/date stamp ″M_3″ to the labelling labelling machine ″M_4″ (diagnoser S_7).

e_{M4} : End of labelling operation (diagnoser S_8).

$e_{M4\rightarrow M5}$: End of transfer of the bottles from the labelling machine ″M_4″ to packaging machine ″M_5″ (diagnoser S_9).

e_{M5} : End of packaging operation (diagnoser S_{10}).

$e_{M5\rightarrow SA}$: End of transfer of the bottles of packaging machine ″M_5″ to stock″SA″ (diagnoser S_{11}).

Let us suppose that we want to monitor the duration between the two events, e_{C1} (the beginning of transfer of the containers to the bottle filling machine is indicated by the generation of an event e_{C1}) and $e_{M5\rightarrow SA}$ (the end of transfer of the bottles of packaging machine "M_5" to stock "SA" is indicated by the generation of an event $e_{M5\rightarrow SA}$).

To monitor this duration, it is necessary to check the timing constraint linking the occurrences of the two events e_{C1} and $e_{M5\rightarrow SA}$.

This timing constraint is a global one, therefore the verification of this constraint can be done through the measure of the operating and transfer durations associated to the intermediate events (each event is associated to each operations).

As previously mentioned, the global constraint to compute is an interval constraint type defined by:

$C_{C1,\ M5\rightarrow SA}$:

$$d_{C1,\ M5\rightarrow SA} \leq O(e_{C1}) - O(e_{M5\rightarrow SA}) \leq f_{C1,M5\rightarrow SA}$$

with:

$$d_{C1,\ M5\rightarrow SA} = \sum_{i=2}^{12} a_i$$

$$f_{C1,\ M5\rightarrow SA} = \sum_{i=2}^{12} b_i$$

The minimum time ($d_{C1,\ M5\rightarrow SA}$) granted to the production of a milk bottle is 229 *u.t* whereas the maximum time ($f_{C1,\ M5\rightarrow SA}$) is 342 *u.t*.

Step 4 — Scenario formation

A/ Scenario: delay of packaging operation

Let us suppose that:

$\Phi_{C1\rightarrow M5} = \Phi_{C1\rightarrow M1} + \Phi_{M1} + \Phi_{M1\rightarrow M2} + \Phi_{M2} + \Phi_{M2\rightarrow M3} + \Phi_{M3} + \Phi_{M3\rightarrow M4} + \Phi_{M4} + \Phi_{M4\rightarrow M5} + \Phi_{M5}$,

where: $\Phi_{C1\rightarrow M1}$= 115 *u.t*, Φ_{M1}= 10 *u.t*, $\Phi_{M1\rightarrow M2}$= 10 *u.t*, Φ_{M2}= 36 *u.t*, $\Phi_{M2\rightarrow M3}$= 45 *u.t*, Φ_{M3}= 13 *u.t*, $\Phi_{M3\rightarrow M4}$ = 8 *u.t*, Φ_{M4} = 35 *u.t* and $\Phi_{M4\rightarrow M5}$= 45 *u.t*.

Suppose that the transfer duration linking the occurrences of the events e_{C1} and $e_{C1\rightarrow M1}$ is 115 *u.t* ($\Phi_{C1\rightarrow M1}$= 115 *u.t)* and that the operating durations associated to the packaging machine is 40 *u.t* (Φ_{M5} = 40 *u.t*). It is sure that the production cycle of a milk bottle will be delayed (Figure 5) according to the measured duration $\Phi_{C1\rightarrow M5}$ and to the transfer durations of the bottles of the packaging machine "M_5" to the stock "SA" ($\Delta_{11} \in [5, 15]$). Consequently, the global constraint ($C_{C1,\ M5\rightarrow SA}$) is violated. In fact, a late detection of a time constraint violation can imply the propagation of the failure symptom and can induce some catastrophic consequences on the functioning of the system. Therefore, the considered approach

uses the additional information provided by the occurrences of the intermediate events and allows detecting a failure symptom when a constraint is violated.

The results, Figure 5, highlight areas of certainty for the detection function: as the constraints traduce a normal functioning of the monitored system, a low degree of possibility ($\mu(\Phi_{C1\rightarrow M5})= 0$) induces the detection of a failure symptom.

The chronicle recognition allows detecting a failure symptom when a constraint is violated. When a symptom of an abnormal functioning is claimed, a diagnosis text is generated. The diagnosis text determines failed states (deviations from the normal function) of the system (milk unit failure) and its subsystem (failure of packaging machine, failure of bottles filling machine, …).

$X_1= d_{C1,\ M5\rightarrow SA} - \delta_M = 214s$; $X_2= d_{C1,\ M5\rightarrow SA} - \delta_m = 224s$
$X_3= f_{C1,\ M5\rightarrow SA} - \delta_m = 337s$; $X_4= f_{C1,\ M5\rightarrow SA} - \delta_M = 327s$

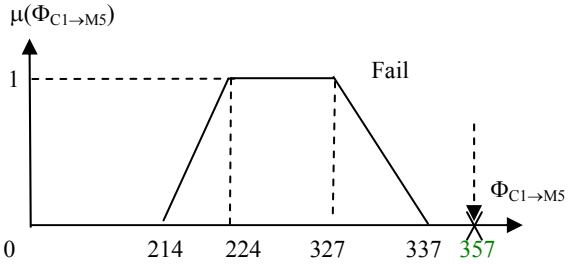

Figure 5. Possibility function considering $\Phi_{C1\rightarrow M5}$

When a symptom of an abnormal functioning is claimed by the chronicle mechanism, it is imperative to localize the failure by using fault tree as a modelling tool.

3.3 Fault tree of the milk manufacturing unit

Step 5 — construct fault trees for observable system deviations

In the case of a system comprising a large number of components, failure may occur due to various failure combinations involving one or more components. This relationship between component and system failure is represented in a fault tree.

To establish the causality of failures on the sub-systems (packaging machine) that can affect the system status, a fault tree, Figure 6, was constructed and the milk unit failure was defined as the top event of the fault tree. This diagnostic tree was comprised of 19 basic events. When the sub-systems deviate from their expected normal behaviour, the fault tree provides information regarding the failure state of its components.

The causes of such deviations, such as a failure of the bottle filling machine, a failure of the milk bottle capper, etc., were determined in the structure of the fault trees that were constructed by traversing the structural model of the system. Nodes of those fault trees were also augmented with monitoring expressions and the trees were then used in real-time for the diagnosis of root causes of failure. According to Figure 6, the logical expression of top event (F_0) of the fault tree is:

$$F_0 = F_1 + F_2 + F_3 + F_4 + F_5$$
$$= [(d_a \times d_b)+(d_c + d_d)]+[(d_e + d_f) + d_g]+[(d_h \times d_i)+(d_j + d_k)]+[d_l + d_m + d_n + d_o] +[(d_p + d_q)+(d_r + d_s)]$$
$$= [ds_0+ds_1] + [ds_2+d_g] + [ds_3+(d_j + d_k)] + [d_l + d_m + d_n + d_o] + [ds_4 + ds_5]$$

with:

$ds_0= (d_a \times d_b)$; $ds_1= (d_c + d_d)$; $ds_2 = (d_e + d_f)$;

$ds_3= (d_h \times d_i)$; $ds_4= (d_p + d_q)$; $ds_5 = (d_r + d_s)$;

We define the vectors of:

- Basic failures (associated to basic events):

$D = [d_a; d_b; d_c; d_d; d_e; d_f; d_g; d_h; d_i; d_j; d_k; d_l; d_m; d_n; d_o; d_p; d_q; d_r; d_s]$

- Intermediate failures (combination of basic failures):

$\overline{D} = [ds_0; ds_1; ds_2; ds_3; ds_4; ds_5]$

Let us take the example of a failure of the packaging machine; the proposed fault tree allows localizing, and identifying problems that occur on the packaging machine. These problems are physical (failure of the mechanical fingers, failure of sealing bar …). Therefore the fault tree explains in details what is happening with the system and what a supervisor agent needs to know in order to avoid a damage of the process or an accident with human beings.

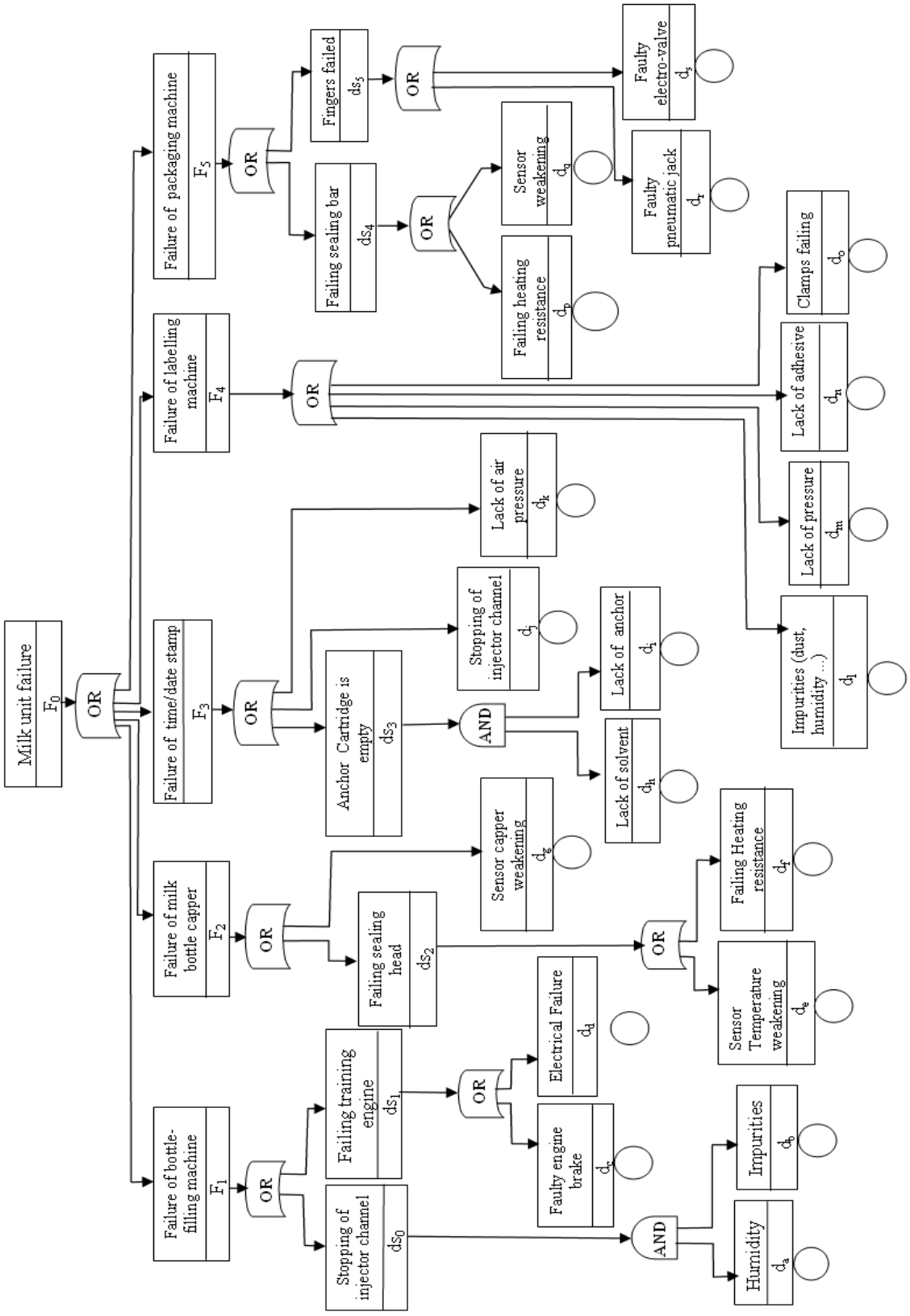

Figure 6. Fault tree of milk manufacturing unit

4. Conclusion

This paper deals with supervision in critical time manufacturing job-shops. In such systems, operation times are included between a minimum and a maximum value. In the proposed manufacturing workshop the determining parameter for quality and cost is

the time, which must belong to a very strict validity interval. The monitoring allows to keep on producing, by on-line diagnosis, while providing correct quality of the manufactured products.

A monitoring mechanism integrating design models and safety analysis is described. Its purpose is to explain in details what is happening on the system and to help operators identifying failures in order to avoid a damage of the process or an accident with human beings.

The proposed monitoring model can capture the symptoms of failure on process components as violations of constraints. From those symptoms, the monitoring process can detect single or multiple (dependent) failures and then diagnose the single or multiple causes of such failures by using fault tree. Consequently, the proposed monitoring approach which combines chronicles and fault tree techniques allows evaluating abnormal functioning and the reliability of manufacturing systems with time constraints.

The distributed detection allows detecting failure symptoms and performing an early diagnosis. The failure symptom may occur due to various failure combinations involving one or more components. This relationship between component and system failure is represented by a fault tree.

It is often difficult to estimate precise failure probability of the components due to insufficient data or vague characteristics of the events. Therefore, in the absence of accurate data, it may be necessary to work with rough estimates of probabilities, and the failure probabilities are treated as random variables with known probability distributions. Fault Tree Analysis FTA [23], [24] might be the only way to predict the reliability of a manufacturing milk unit when little quantitative information is available.

The monitoring mechanism helps the monitor to prevent false alarms, with the aid of chronicles recognition. Furthermore, the chronicle helps the monitor to recognize when a condition should be considered normal and when it should be interpreted as an indication of a real disturbance. The monitor can also determine the functional effects of failures: This is quite useful for the maintenance task.

It is interesting as further research to incorporate the issues of maintenance and repair strategies into the proposed monitoring approach by identifying scenarios with serious consequences and critical components; By knowing the scope of a failure and by being able to determine its permanency, the monitor can apply successive corrective measures at increasingly abstract levels in the hierarchy of the system.

Finally a comparative study based upon several cases should be developed. A comparison with the proposed monitoring architecture and results with the works using a filtering mechanism of sensors signals integrating robustness values [7], should also be considered.

REFERENCES

1. NIEBUR, D., **Expert Systems for Power System Control in Western Europe**. IEEE Symp. on Intel. Control, Philadelphia, 1990.
2. CORDIER, M. O., C. DOUSSON, **Alarm Driven Monitoring Based on Chronicles**. 4th Symp. on Fault Detection, Supervision and Safety for Technical (Safeprocess), Budapest, 2000.
3. RIESE, M., **Diagnosis of Extended Finite Automata as a Dynamic Constraint Satisfaction Problem**. Intl. Workshop on Principles of Diagnosis (DX-93), Aberystwyth, 1993.
4. DOUSSON, C., P. GABORIT, M. GHALLAB, **Situation Recognition: Representation and Algorithms**, Intl. Joint Conf. on Artificial Intel. (IJCAI'93), Chambéry, 1993, pp. 166-172.
5. CORDIER, M. O., X. LE GUILLOU, S. ROBIN, L. ROZÉ, T. VIDAL, **Distributed Chronicles for On-line Diagnosis of Web Services**, 18th Intl. Workshop On Principles of Diagnosis (DX'07), Nashville, May 2007, pp. 37-44.
6. PENCOLÉ, Y., M. O. CORDIER, **A Formal Framework for the Decentralised Diagnosis of Large Scale**

Discrete Event Systems and Its Application to Telecommunication Networks. Artificial Intel. Journal, Vol. 164(1-2), 2005, pp. 121-170.

7. M'HALLA, A., N. JERBI, S. COLLART DUTILLEUL, E. CRAYE, M. BENREJEB, **Fuzzy Filtering of Sensors Signals in Manufacturing Systems with Time Constraints**. Intl. Journal of Computers, Comm. & Control (IJCCC). Vol. 5(3), 2010, pp. 362-374.
8. SILVEIRA, D., M. COMBACAU, A. BOUFAIED, **Prognosis and Recovery Evaluation in Flexible Manufacturing Supervision**. Journal of Decision Systems**,** Vol. 12, 2003, pp. 93-109.
9. GUILLOU, X., M. O. CORDIER, S. ROBIN, L. ROZE, **Chronicles for On-line Diagnosis of Distributed Systems**. 18th European Conf. on Artificial Intel. (ECAI'08), Patras, July 2008.
10. DOUSSON, C., P. LE MAIGAT, **Chronicle Recognition Improvement Using Temporal Focusing and Hierarchization**. Proc. of the 20th Intl. Joint Conf. on Artificial Intel. (IJCAI), Hyderabad, January 2007.
11. TAISNE, J., **Intelligent Alarm Process for DMS Based on Chronicle Concept**. 19th Intl. Conf. on Electricity Distribution (C I R E D), Vienna, May 2007.
12. BOUFAIED, A., A. SUBIAS, M. COMBACAU, **Distributed Time Constraints Verification Modelled with Time Petri Nets**. 17th IMACS World Cong. Scientific Computation, Applied Math. and Simulation, Paris, July 2005.
13. BOUFAIED, A., A. SUBIAS, M. COMBACAU, **Détection distribuée par reconnaissance floue de chroniques**. Journal Européen des Syst. Aut. (JESA), Vol. 40(2), 2006, pp. 233–259.
14. MHALLA, A., N. JERBI, S. COLLART DUTILLEUL, E. CRAYE, M. BENREJEB, **Distributed Monitoring Based on Chronicles Recognition for Milk Manufacturing Unit**. Journal. of Aut. & Syst. Eng. (JASE), Vol. 4(1), 2010.
15. ANDREWS, J. D., T. R. MOSS, **Reliability and Risk Assessment**, 2nd ed., Prof. Eng. Publishing Limited, Bury St Edmunds and London, 2002.
16. ENRICO, Z., F. D. FRANCESCO, **Processing Dynamic Scenarios from a Reliability Analysis of a Nuclear Power Plant Digital Instrumentation and Control System**. Journal of Annals of Nuclear Energy, Vol. 36(9), 2009, pp. 1386-1399.
17. YUHUA, D., Y. DATAO, **Estimation of Failure Probability of Oil and Gas Transmission Pipelines by Fuzzy Fault Tree Analysis**. Journal of Loss Prevention in the Process Industries, Vol. 18(2), 2005, pp. 83-88.
18. CHEN, J., J. HOWELL, **A Self-validating Control System Based Approach to Plant Fault Detection and Diagnosis**. Computers & Chemical Engineering, Vol. 25(2-3), 2001, pp. 337-358.
19. ERICSON, C. A., **Fault Tree Analysis — a History**. 17th Intl. System Safety Conf., Orlando, USA, 1999.
20. KAISER, B., P. LIGGESMEYER, O. MACKEL, **A New Component Concept for Fault Trees**. 8th Australian Workshop on Safety Critical Systems and Software, Canberra, 2003.
21. CORDIER, M. O, X. LE GUILLOU, S. ROBIN, L. ROZÉ, T. VIDAL, **Distributed Chronicles for On-line Diagnosis of Web Services**. 18th Intl. Workshop On Principles of Diagnosis (DX'07), Nashville, May 2007, pp. 37-44.
22. MORIN, B., D. HERVE, **Correlation of Intrusion Symptoms: An Application of Chronicles,** 6th Intl. Sym. on Recent Advances in Intrusion Detection, Pittsburgh, 2003.
23. SALLAK, M., C. SIMON, J. F. AUBRY, **A Fuzzy Probabilistic Approach for Determining Safety Integrity Level**. IEEE Trans. on Fuzzy Syst. 2007, pp. 239-248.
24. VOLKANOVSKI, A., M. CEPIN, B. MAVKO, **Application of the Fault Tree Analysis for Assessment of Power System Reliability**. Reliability Engineering & System Safety, Vol. 94, 2009, pp.1116-1127.