20

Improvement of Production Lines using a Stochastic Approach

Cecilia Zanni-Merk and Philippe Bouché
LGECO - INSA de Strasbourg
24 Bd de la Victoire, 67084 Strasbourg, France
{cecilia.zanni-merk, philippe.bouche}@insa-strasbourg.fr

1. Introduction

The search for productivity sources makes the improvement of the contemporary production systems necessary. The production actors are, in a systematic and permanent way, engaged in three stages: the audit, the diagnosis and the search for solutions to improve their production systems.
For the audit of production systems, different Internet and Intranet technologies allow measuring and storing the state of the different production resources in real time.
From these data and during the stage of analysis of production flows, the production personnel and the staff in charge must be able to find and formalize the problems inducing a faulty operation of the manufacturing system. Solutions must be imagined in order to increase the productivity at a given cost.
Nowadays, the stages of diagnosis and solution search are primarily instrumented by little formalized expert knowledge. This lack of formalism generates heavy development costs, does not guarantee reproducibility and does not support the necessary knowledge capitalization for the improvement of the production system within the same company. To solve these problems, a solution consists in formalizing the necessary knowledge to set and solve the problems related to that lack of productivity from the data collected during the audit stage. This formalization has to give birth to software tools for assisting the involved actors in a permanent and proactive way.
Several works have been carried out on the performance evaluation of unreliable production lines (Tempelbeier & Burger, 2001; Van Bracht, 1995; Xie, 1993). However, research on the simultaneous consideration of maintenance policies, production planning and quality improvement from an industrial point of view has still to be done.
Confronted with these industrial problems, there are two research lines. On the one hand, there is a great number of scientific works on the detailed modelling of production resources and activities. On the other hand, a much less developed research line is interested in the modelling of problem solving in production systems design. From these two categories, our research group is interested in the understanding and modelling of the field experts reasoning during the stages of production flow analysis and solution searching. We are also interested in automating this reasoning in order to bring proactive software assistance.

With this goal in mind, we will study the different behaviours of the production line that would lead to a lack of productivity, according to three axes:

- The production axis, by indicating the losses that are incurred by problems in the production planning (ergonomy of a workstation, lack of training of an operator, etc.),
- The quality axis, by indicating in which measure the quality problems affect the productivity,
- The maintenance axis, by taking into account the losses due to maintenance operations.

We will also be interested by the study of "cause-effect" relationships among these behaviours: is the lack of training of the operator causing quality problems encountered later?

This article presents, therefore, our approach for performance analysis and improvement of production flows in the three dimensional space we have just described. The approach is based on statistical and probabilistic methods and is a new case of application of the stochastic approach (Le Goc et al., 2006).

Section 2 presents the industrial context and describes the project. Section 3 presents the data graphical representation before setting the definition of phenomena in Section 4. Section 5 effectively presents the stochastic approach. Section 6 presents an application of the method on a real industrial case. Identification of breakdown models will be the base to propose action plans, presented in Section 7. Section 8 gives a quantification of the losses incurred by the occurrences of these anomalous events in the line. Section 8 states our conclusions and perspectives of future work.

2. The industrial context

As we have presented in previous communications (Zanni et al., 2007; Bouché & Zanni, 2008a; Zanni & Bouché, 2008b), our group is interested in the development of a software tool for allowing the decision makers in companies to have an analysis of their production line flows. This analysis will consist in a general and by-workstation productivity evaluation, the main objective being the maximization of this productivity in terms of the number of good produced parts in a given time window.

This diagnosis will be followed by an action plan for the improvement of the line, according to three criteria (quality, maintenance and yield) and a valorisation of the losses that could have been avoided if the action plan was executed. The general idea is to maximize the productivity by improving the production cycle time and by reducing the workstations breakdowns / outages and the number of rejected parts.

We are using a data acquisition system that, after having placed sensors in strategic places (Fig. 1) of the production line, allows the measuring of different indicators.

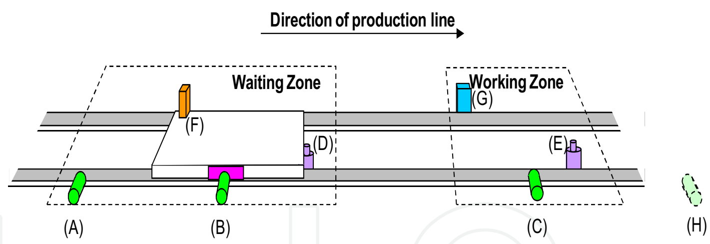


Fig. 1. Location of sensors in a workstation

During the production stage, we are able to detect if the part is good or bad (and, eventually, the associated fault code) and the times of (Fig. 2):

- The arrival of the part to the workstation (Sensor B),
- The beginning of processing of the part in the workstation (Sensor C),
- The end of processing of the part in the workstation (this fact is detected automatically for an automatic workstation or with an action on a sensor for a manual workstation),
- The exit of the part from the workstation (Sensor H).

They aim at defining a set of durations linked with the different stages of the work on the piece on the workstation (Fig. 2).

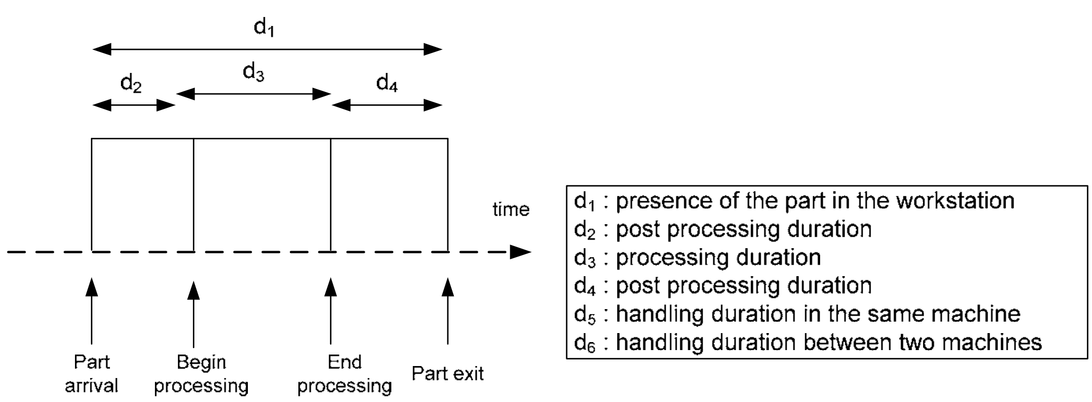


Fig. 2. Indicators to be measured during production

It is important to note that the only durations including effective working on a piece (d_1 and d_3) are durations with a value added.

The other durations are considered without a value added because they correspond to waiting, handling, or other *non productive* activities.

In the case of the failure of a workstation, the indicators we measure are (Fig. 3):

- The failure beginning time,
- The failure end time,
- The identification code of the failure.

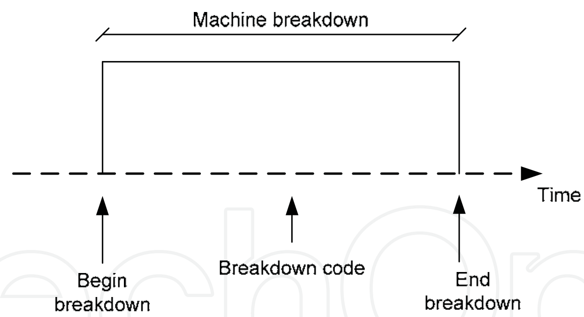


Fig. 3. Indicators to be measured during the breakdown station

The data acquisition system will also provide other necessary information, in particular, the control parameters of the workstations, i.e. some workstation characteristics that will be specific for each process plan. It will also provide maintenance data, information on production modifications, and other relevant information.
We study production lines such as the one described in Fig. 4:

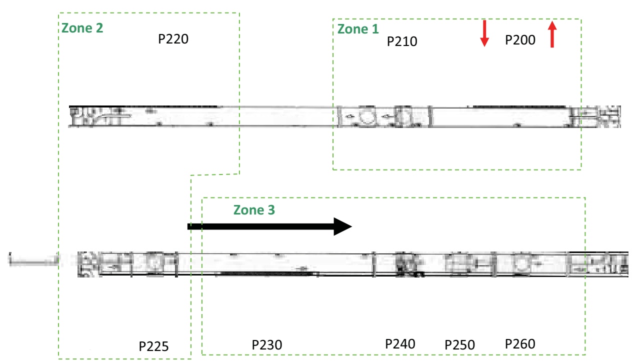


Fig. 4. Example of a production line

This is a closed loop, where there is a set of workstations, which can be automatic or manual. In Fig. 4, workstation P200, for example, is the point where pieces are injected in the line and where they go out. In addition, a finite set of pallets turns around the loop. This fact allows the transportation of the pieces from a workstation to the next one.
This organisation makes necessary to take into account a last set of parameters, which are:

- The instance of the production plan,
- The working team.

To take these parameters into account, data are separated by production type and/or by working team; the idea is to guarantee that time periods for analysis are uniform.

3. Data graphical representation

These data can be analyzed with frequencies and sequential methods.
In first place, we proceed to a Poisson analysis. A Poisson process is a process of enumeration, which describes the evolution of a quantity in time (Fig. 5). In our study, it will be a question of tracing the evolution in time of durations d_1, d_2… In the case of a perfect process, the Poisson curve is a line characterized by its slope λ (we will also speak about the speed of the Poisson process).

$$\lambda = \frac{number\ of\ parts}{time} \tag{1}$$

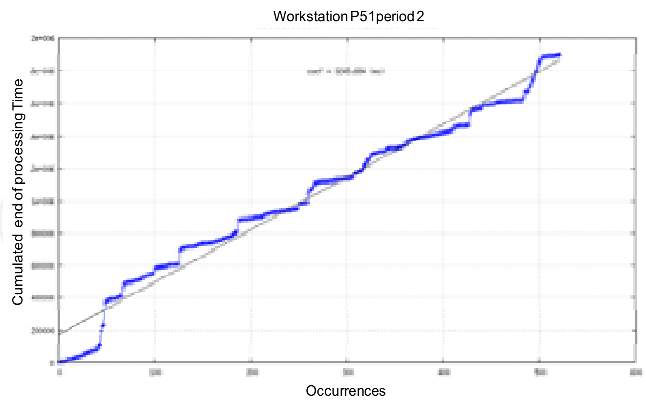


Fig. 5. Poisson process of evacuation times of a workstation

In real processes, we will observe various slopes, which will make possible, for example, to determine the moments when the production is faster, if there are intervals of drift in the workstation, and to define ranges where the behaviour of the station requires a more thorough analysis.
In second place, we can study the evolution of the working time with a model inspired in control charts, which are used in Statistical Process Control (Ishikawa, 1982). We trace the different durations of the tasks according to time (Fig. 6).

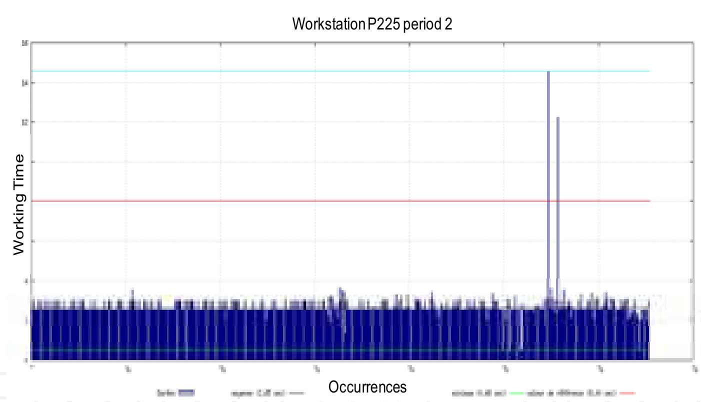


Fig. 6. "Control chart" of processing times of a workstation

That will make possible the study of possible drifts of the workstation to check if the process is under control; to identify the workstations where improvements could be made; or to identify changes of rate/rhythm or perturbations.
Finally, we can analyse properties of the distribution of durations (Fig. 7). We trace the frequency of the durations to study the setting under statistical laws of the station to consider.
From these curves, a certain number of analyses may be carried out, such as the analysis of dispersions, of aberrant values, or others.
Other graphical representation, such as synthesis representations, could be imagined. Nevertheless, the three types of presented representations are, for us, the base of the

analysis. This is a first method to have a better view of reality, and to identify specific zones of bad behaviour or specific phenomena. It corresponds to the more specific level of abstraction. To make better studies, we need to build meta-data, which will be associated to specific events or behaviours of the production. These behaviours cannot be directly deduced from data.
A set of transformations has to be applied to data to obtain the expression of certain behaviour under the form of a "phenomenon". The next section will define what we call a phenomenon before showing how we can compute phenomena from data.

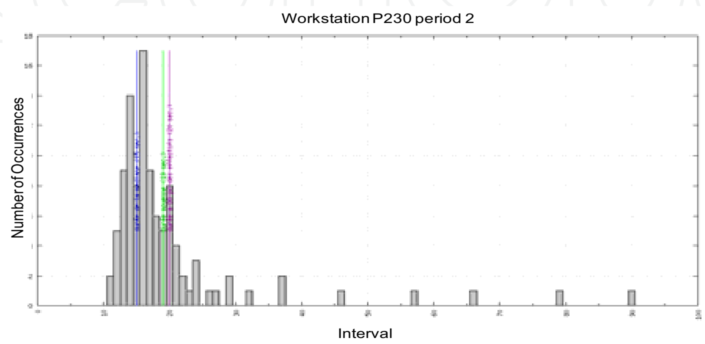


Fig. 7. Distribution of the processing times of a workstation

4. Phenomena

Phenomena are the expression of particular behaviours. They are described by a set of attributes, and at least (Le Goc, 2004b):

- A name,
- A characterization of the localisation in the production line,
- Two dates:
  - A begin date,
  - An end date.

4.1 Definition of Phenomena

While studying durations, we consider three parameters at the base of the Statistical Process Control principles (Ishikawa, 1982):

- The stability of the evolution of the duration to verify if the behaviour is stable or not (Fig. 8),

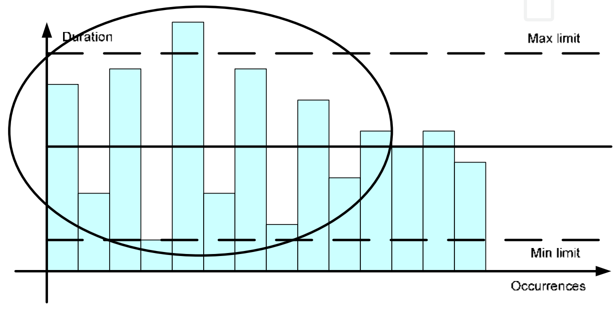


Fig. 8. Instability

- The drift of the evolution of the duration to check if the behaviour is constant or if there are positive or negative drifts (Fig. 9),

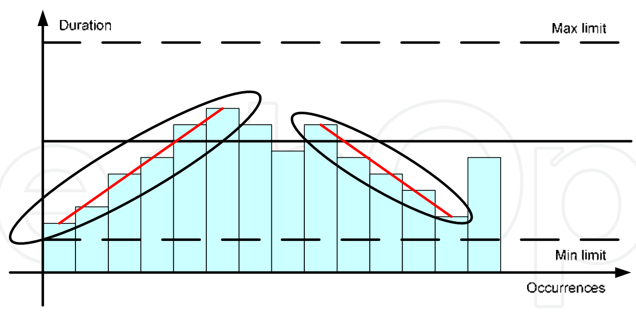


Fig. 9. Positive and Negative Drift

- The analysis of the values of the duration to verify if they are out of bounds (Fig. 10).

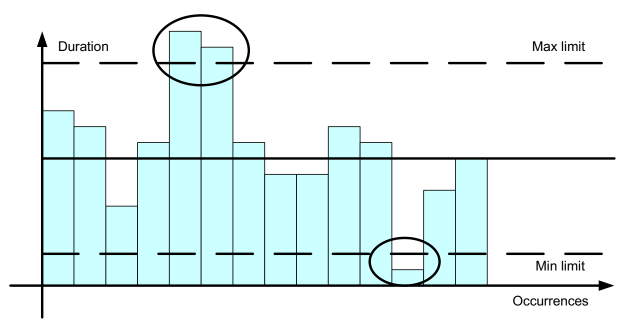


Fig. 10. Out of bounds values

These three characteristics are worth being studied on durations that are with a value added that is to say, if they are the result of an activity (human or made by a robot), such as d_3, the processing duration.
If the duration is with a value added, like in an automatic handling, the only point of interest will be to study the drift of the evolution of the duration.
Now we have six durations:

- d_1: the presence of the part in the machine,
- d_2: the pre-processing duration,
- d_3: the processing duration,
- d_4: the post-processing duration,
- d_5: the handling duration in the same workstation,
- d_6: the handling duration between two workstations.

As d_1 is the sum of d_2, d_3 and d_4, we will not make studies on this duration.
As d_2, d_4, d_5 and d_6 are durations without a value added, we will only study the drift on these durations.
In addition, to finish, we will carry out all the studies on d_3, the only duration with a value added.
Apart from these "duration linked" phenomena and that are related with the "production" axis of our space of study, we must also consider a set of phenomena related the "quality" and "maintenance" axes.

Regarding the "quality" axis, there are no studies on durations. There is only a discrete event characteristic of the behaviour that indicates that a piece is bad at a control workstation.

Other specific behaviours share the same characteristic as the last one, that is, the fact that there is only a discrete event of the behaviour: the start and the end of the production

Finally, we have phenomena related to the "maintenance" axis. We are able to detect the period of time when the workstation is stopped and the fault code that produced this stop. Studies on this duration (and its subcomponents, such as the time to wait for the maintenance team to arrive or the time interval between the arrival of the maintenance team and the effective restart of the workstation) might be carried out.

4.2 Phenomena related to the Production Axis

The following subsection gives a complete description of the phenomena we have retained with their characterization from an industrial point of view.

4.2.1 The Stock_Saturation phenomenon

It can be deduced from data of post-processing durations (d_4) and is characterized by an increase of the slope of those cumulated durations.

More precisely, on the Poisson curves, a *Stock_Saturation* phenomenon will be an increase of the speed of the process and thus a positive drift of the slope of the curve. On control charts, this phenomenon will be the translation of a quick increase in the execution times of a task in time (Fig. 11).

When there is a stock saturation, the workstation is not able to make the piece exit the working zone; this is why we detect this phenomenon by an increase of the speed of the post-processing duration.

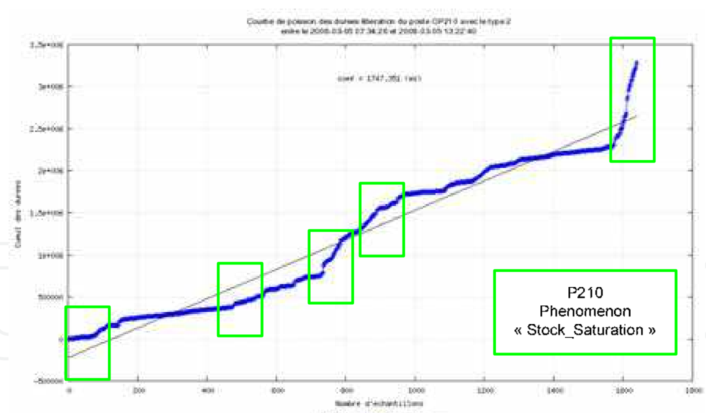


Fig. 11. Example of the *Stock_Saturation* phenomenon

4.2.2 The Workstation_Drift phenomenon

It is the expression of a drift of the production time on a workstation. We can observe it in the Poisson curves or the control charts with the study of the processing duration (d_3).

More precisely, on the Poisson curves, a *Workstation_Drift* phenomenon will be an increase or a decrease of the speed of the process and thus a positive or negative drift of the slope of

the curve. On control charts, this phenomenon will be the translation of a regular increase or decrease in the execution times of a task in time.
We will use two phenomena to differentiate if the drift is positive or negative (Fig. 12):

- *Positive_Worstation_Drift*
- *Negative_Workstation_Drift*

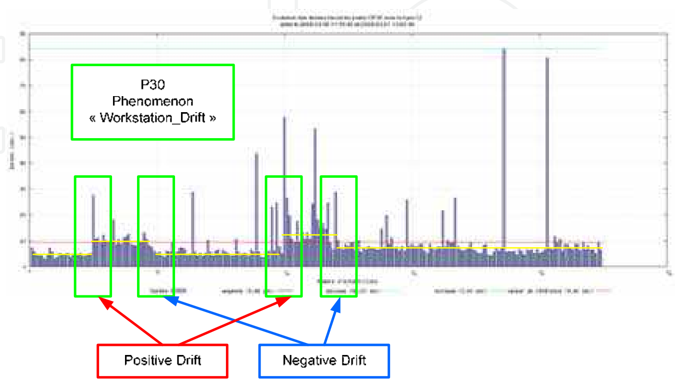


Fig. 12. Example of the *Drift* Phenomena

4.2.3 The Workstation_Instability phenomenon

It can be deduced from the working duration on a workstation (d_3).
The objective is to verify if the behaviour is stable or if there is any instability in the work.
It will be characterized by a great variability of durations on the control charts (Fig. 13).
In fact, SPC establishes that *two consecutive measurements that deviate from each other more than twice the value of the standard deviation indicate instability* (second rule of Shewart (Shewart, 1939)). In our case, this rule can be only applied to aberrant values.
Therefore, to detect this phenomenon, we identify, firstly, the couples of successive values in the current sample, which deviate from each other more than twice the standard deviation.
If this number of couples is, at least, 10 % of the size of the sample, we say we are in presence of a *Workstation_Instability* phenomenon

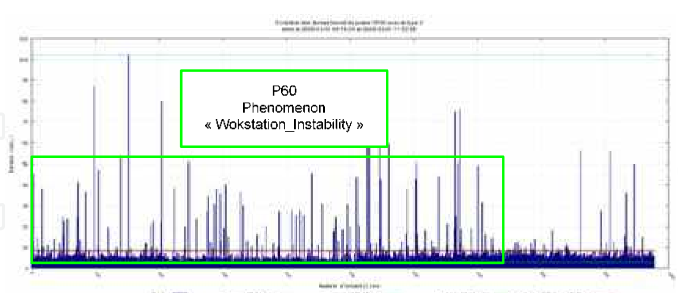


Fig. 13. Example of the *Workstation_Instability* phenomenon

4.2.4 The Out_Of_Bounds_Time phenomenon

It can be deduced from the working duration on a workstation (d_3).
The objective is to analyze this duration to identify if there are values out of limit. If values are bigger than the maximum limit, the speed of the workstation is too slow; we will have a *Slow_Working_Period* phenomenon. If values are under the minimum limit, the speed of the

workstation is too fast (which could be the cause of workstation failure or of the production of bad pieces), we will have a *Fast_Working_Period* phenomenon (Fig. 14).

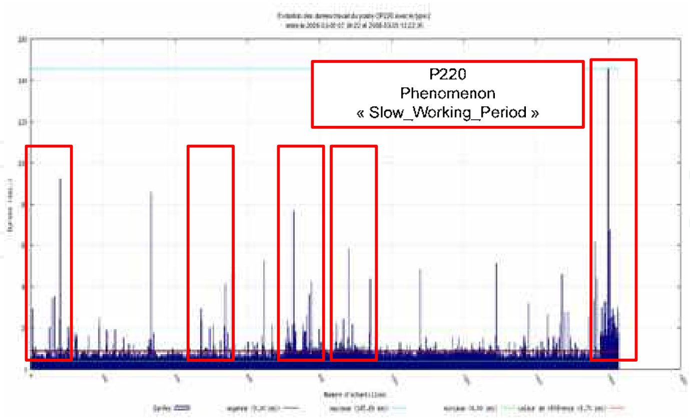


Fig. 14. Example of the *Slow_Working_Period* phenomenon

4.2.5 The Lack_Of_Components phenomenon

It can be deduced from data of pre-processing durations (d_2). It will be characterized by an increase of the slope of the cumulated pre-processing duration.

More precisely, on the Poisson curves, a *Lack_Of_Components* phenomenon will be an increase of the speed of the process and thus a positive drift of the slope of the curve. On control charts, this phenomenon will be the translation of a quick increase in the execution times of a certain task in time.

The lack of components makes impossible that the workstation begins its work; this is why we detect this phenomenon by an increase of speed of the pre-processing duration.

4.2.6 The Handling_Speed phenomenon

It can be deduced from data of handling durations in the same workstation (d_5). It will be characterized by an increase or a decrease of the slope of those cumulated durations.

This leads to the definition of two phenomena:

- *Slow_Handling_Speed*
- *Fast_Handling_Speed*

It is important to consider the case where the handling speed is too fast, because even if it is not a direct cause of productivity losses, it could cause saturation on workstations or other problems in the production line.

4.2.7 The Lack_Of_Stock phenomenon

It can be deduced from data of handling durations between two workstations (d_6). It will be characterized by an increase of the slope of those cumulated durations.

The lack of stock makes impossible that the workstation begins its work because there are no pieces in transit between two workstations; this is why we detect this phenomenon by an increase of speed of the handling duration between those two workstations.

4.2.8 Completeness of this set of phenomena

This set of phenomena has been defined in collaboration with experts of production, and in function of the goal of the analyses. The list we have retained is complete according to the explanations in section 4.1 (Fig. 15):

Studies on durations			
	Stability of the evolution of the duration	Drift of the evolution of the duration	Analysis of values of the duration
d_1			
d_2		Lack_Of_Components	
d_3	Workstation_Instability	Workstation_Positive_Drift	Slow_Working_Period
		Workstation_Negative_Drift	Fast_Working_Period
d_4		Stock_Saturation	
d_5		Slow_Handling_Speed	
		Fast_Handling_Speed	
d_6		Lack_Of_Stock	

Fig. 15. List of phenomena on durations

4.3 Phenomena related to the Quality Axis

The main element that we must consider for the quality axis is the production of bad pieces. In this case, we will not really have a phenomenon but just a discrete event (which is a kind of phenomenon where the end date is the same as the begin date).

Therefore, we will characterise the production of bad pieces by a *Bad_Piece* phenomenon with, at least, the following information:

- Date of detection,
- Fault code (if available),
- Localisation of the detection place.

4.4 Phenomena related to the Maintenance Axis

These phenomena are not deduced from durations, we obtain them directly from the data acquisition system.

We have a measure of the gravity of the failure on four levels:

- *Type 1 - Energy cut-off (air, electricity…) and immediate halt of the workstation cycles:* In general, this type of failure announces an emergency stop (light barrier crossed through, emergency stop button triggered…). Everything must be stopped in the position in which it is (for example, if a hydraulic or pneumatic actuator moves by its own weight without energy, there will be blockers to lock the actuator movements).

  To continue working with the workstation it is required:
  - To correct the problem (to give off the immaterial barriers, to rearm the emergency stop keys …) ,
  - To switch on energy on the workstation (electric, pneumatic…),

  - To acknowledge the failures appeared in the PLC (programmable logic controller),
  - To reset the workstation (with the initial settings of all the components),
  - To set on "start" the PLC cycle.
- *Type 2- Dead halt of the PLC cycles and immobilization of movements:* The cycles are stopped, but energies are not cut.
  In this scenario, movements are just blocked, but energy flows are not cut. In general, this type of failure is reported when a sensor is faulty. For example, the movement of an actuator arrives to its end without having triggered the corresponding sensor (for example, the end of movement sensor is out of order). The PLC will indicate a type 2 failure indicating the defective component.
  At our industrial partner's, when a workstation has a type 2 failure, all the pieces in progress are declared as bad, and therefore, cannot be re-injected in the line for further processing.
  To continue working with the workstation it is required:
  - To correct the problem (often we will have to switch to manual mode to release actuators of the workstation),
  - To acknowledge the failures appeared in the PLC,
  - To reset the workstation (if necessary),
  - To set on "start" the PLC cycle.
- *Type 3 – Message*: This type of failure has just for informational purposes. The workstation cycles are not stopped and energy flows are not cut.
  For example, a grease barrel reaches its low-level limit; a type 3 message will indicate this fact. If it is not replaced and it is empty, a type 2 failure will be declared with the consequent workstation stop.
  In addition, this type of message is used to indicate to the operator what he has to do with the product in front of him. These kinds of failures are acknowledged automatically.
- *Type 4 - A particular definition of our industrial partner and only in a few of his production lines:* This failure is the same as a Type 2 one, but the product can be taken again for further work after resumption of the cycle.

Therefore, we have defined four phenomena for each workstation:

- *Failure_Level_1*
- *Failure_Level_2*
- *Failure_Level_3*
- *Failure_Level_4*

They will be characterized by the gravity and their start and end dates.

4.5 How to build phenomena from data

After having established all phenomena and their description, we will see two examples of the algorithms we use to build phenomena.
For example, if we consider the *Workstation_Drift* phenomenon, it is the expression of a drift of the production time on a workstation. We can observe it in the Poisson curves or the control charts with the study of the processing duration (d_3) (Fig. 16).

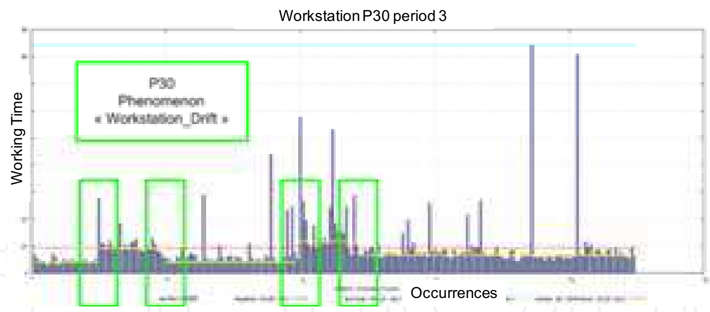


Fig. 16. Visual observation of the *Workstation_Drift* phenomenon

To compute occurrences of this phenomenon, we will calculate the slopes of the Poisson processes that are associated with this workstation during the considered time window. If we observe several slopes, named successively λ_1, λ_2… λ_n and that these slopes are decreasing, then we can diagnose a *Workstation_Drift* phenomenon.

$$\begin{array}{l}\forall \lambda_i, i \in [1,n], \text{IF } \lambda_i > \lambda_{i+1} \\ \quad \text{THEN } \textit{Workstation\_Drift}\end{array} \tag{2}$$

In the same way, the processing durations on a workstation can be directly used to diagnose a *Workstation_Drift* phenomenon. It will be characterized by an increase of the average of durations.

$$\begin{array}{l}\forall d_i, i \in [1,n], \text{IF Average}(d_1, ..., d_k) < \text{Average}(d_{k+1}, ..., d_n) \\ \quad \text{THEN } \textit{Worstation\_Drift}\end{array} \tag{3}$$

Let us consider the data in Fig. 17:

occurrence	date	added up durations	slope	lambdas
1084	19/09/2007 19:01	18759881		
			97771,96	0,00010233
1388	19/09/2007 20:00	21730557		
			12599,15	0,00007937
1611	19/09/2007 20:53	14540168		

Fig. 17. Data of workstation P200 of the production line

The slopes values on this manual workstation lead us to say that we are in presence of a *Workstation_Drift* phenomenon from 20h00 to 20h53.
The idea is to calculate the slopes on a temporal horizon that is coherent with the production line speed and the considered workstation and to compare the slopes regularly. The algorithm to build occurrences of this phenomenon is depicted in Fig. 18:

```
Constant : h time windows characterized by two dates d1 (begin) et d2 (end)
Constant : p tolerance
Variables : λ1 past slope, λ2 curent slope
Boolean : C Boolean uses to know if we are in Workstation drift period (1) or not (0)

Data :    NbOc (di) Number of pieces treated at time di
              DC (di) Cumulated period of work at time di

Temporal loop on h
          Compute λ2 = (NbOc(d2) – NbOc (d1))/(DC(d2)-DC(d1))
          Compare λ1 et λ2
          IF the difference is upper than p
                THEN IF  C = 0
                    THEN     Init occurrence of phenomenon Workstation_Drift
                             Begin date phenomenon = d1
                             C=1
          IF difference is lower than p
                    THEN IF C = 1
                             THEN     Stop occurrence of phenomenon Worstation_Drift
                                      End date phenomenon = d1
                                      C=0
          Make change the time, actualize d1 et d2
          Change value λ1 with value λ2
```

Fig. 18. Algorithm for identifying the *Workstation_Drift* phenomenon

Considering Fig. 16, the application of this algorithm will give us four occurrences of the *Workstation_Drift* phenomenon.

To have another example, let us consider the *Stock_Saturation* phenomenon. It can be deduced from data of post processing durations (d_4). It will be characterized by an increase of the slope of the cumulated post processing duration. Fig. 19 shows an example of a time window for workstation P210 where we can observe five occurrences of this phenomenon.

It is important to remark that now we have to consider five occurrences of the *Stock_Saturation* phenomenon rather than all the data on the same time window. We make, also, the assumption that, if no phenomenon is detected, the behaviour of the production line is correct.

Therefore, because phenomena are meta-knowledge, we are able to build a sequence of phenomena, which contains more information than the original data but "lighter" than the original set.

The construction of phenomena is, then, a kind of a discrete event abstraction (Le Goc, 2004a).

Post analysis may be performed on the phenomena sequence, by application of the stochastic approach to identify the correlations that can exist among phenomena. Next subsection will show the bases of the stochastic approach and the way we can use it to obtain fault models. These models will serve in the last step of our development to build action plans.

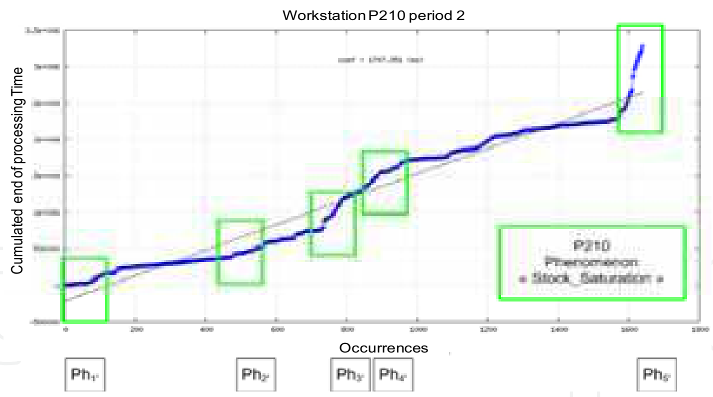


Fig. 19. Example of the *Stock_Saturation* phenomenon

5. The Stochastic Approach

A sequence $\omega = \{o_k\}_{k=0,\dots,m-1}$ is an ordered set of m occurrences $o_k \equiv (t_k, x, i)$ of discrete events $e_k \equiv (x, i)$, where:

- $x \in X$ is the name of a discrete variable,
- $i \in I_x \subseteq \aleph$ is a discrete value of x, and
- $t_k \in \Gamma = \{t_i\}$, $t_i \in \Re$ is the time of the assignation of the discrete value i to the variable x so that: $o_k \equiv (t_k, x, i) \Leftrightarrow x(t_k) = i$.

We have a continuous clock structure. Occurrences may happen at different times, but not necessary at regular intervals (that is to say, t_{k-2}-$t_{k-1} \neq t_{k-1}$-t_k):

$$\forall t_k \in \Re, \forall i \in \aleph, \exists t_{k-1} < t_k, \\ x(t_{k-1}) \neq i \wedge x(t_k) = i \Rightarrow o_k \equiv (t_k, x, i) \tag{4}$$

A couple (o_k, o_n) of two successive occurrences of discrete events related to a variable x describes the modification of the values of the variable x over the interval $[t_k, t_n[$:

$$\forall o_k \equiv (t_k, x, i), o_n \equiv (t_n, x, j), \\ (o_k, o_n) \Rightarrow \forall t \in [t_k, t_n[, x(t) = i \wedge x(t_n) = j \tag{5}$$

As a consequence, a sequence $\omega = \{o_k\}$ of discrete event occurrences $o_k \equiv (t_k, x, i)$ concerning variable x describes the temporal evolution of a discrete function $x(t)$ defined on $\aleph$.

$$R(C^i, C^o, [\tau^-, \tau^+]) \Leftrightarrow \exists o_n, o_k \in \omega, \\ (o_n :: C^o) \wedge (o_k :: C^i) \wedge (d(o_n) - d(o_k) \in [\tau^-, \tau^+]) \\ where\ \forall o_k \equiv (t_k, x, i) \in \omega, d(o_k) = t_k \tag{6}$$

A discrete event class is a set $C^i = \{e_i\}$ of discrete events $e_i \equiv (x, i)$. The notation "$e_i :: C^i$" (resp. "$o_k :: C^i$" or "C^i_k") denotes that the discrete event e_i (resp. the occurrence $o_k \equiv C^i_k$) belongs to the class C^i. A timed binary relation R $(C^i, C^o, [\tau^-, \tau^+])$ describes an oriented relation between two discrete event classes that is timed constrained. "$[\tau,\tau^+]$" is the time interval for observing an occurrence of the output class C^o after the occurrence of the input class C^i (equation 3).

5.1 Abstract Chronicle Model

In this context, an abstract chronicle model is a set of binary relations with timed constraints between classes of discrete events. Such a model is called an "ELP" model (ELP is the acronym of Event Language of Processing, (Le Goc et al., 2006)). For example, the ELP model M_{123}= {R_{12} (C^1, C^2, [τ_{12}^-, τ_{12}^+]), R_{23} (C^2, C^3, [τ_{23}^-, τ_{23}^+])} of Fig. 20 is made of two binary relations between three discrete event classes. A sequence ω satisfies the M_{123} ELP model when:

$$
\begin{aligned}
&\exists o_k, o_n, o_m \in \omega, (o_k :: C^1) \wedge (o_n :: C^2) \wedge (o_m :: C^3) \\
&\wedge (d(o_n) - d(o_k) \in [\tau_{12}^-, \tau_{12}^+]) \wedge (d(o_m) - d(o_n) \in [\tau_{23}^-, \tau_{23}^+])
\end{aligned}
\tag{7}
$$

ELP models can be used to predict the occurrences of discrete event classes (like C^3 in the ELP model M_{123}) in an unknown sequence ω'.

$$
C^1 \xrightarrow{[\tau_{12}^-,\ \tau_{12}^+]} C^2 \xrightarrow{[\tau_{23}^-,\ \tau_{23}^+]} C^3
$$

Fig. 20. ELP representation of the M_{123} model

To this aim, rules of the equation 5 form can be used in a diagnosis task. When such a rule predicts an occurrence of a discrete event class with a minimal confidence, the corresponding ELP model is called a "signature" (Le Goc et al., 2006).

$$
\begin{aligned}
&\forall \omega', \forall o_k, o_n \in \omega', \\
&(o_k :: C^1) \wedge (o_n :: C^2) \wedge (d(o_n) - d(o_k) \in [\tau_{12}^-, \tau_{12}^+]) \\
&\Rightarrow \exists o_m \in \omega', (o_m :: C^3) \wedge (d(o_m) - d(o_n) \in [\tau_{23}^-, \tau_{23}^+])
\end{aligned}
\tag{8}
$$

To measure the confidence of such rules, we define the anticipating ratio of an abstract chronicle model as the number of sub sequences of a sequence ω that matches the complete abstract chronicle model, divided by the number of the sub sequences that matches the abstract chronicle model but without the final binary relation (the class C^3 in Fig. 20). An abstract chronicle model is a signature when its anticipating ratio is equal to or greater than 50%.

5.2 The Stochastic Representation

When the discrete event classes are independent and the distribution of the inter-occurrence times of a discrete event class complies with a Poisson law of the form $f(t)=1-e^{-\lambda t}$[1], the couple

[1] λ is the average number of occurrences in a unit of time and is called the Poisson rate (Cassandras & Lafortune, 2001).

made by the process and its monitoring Knowledge Based System (KBS) can be considered as a stochastic discrete event generator (Le Goc et al., 2006).
Consequently, a sequence of discrete event classes provided by such a generator can be represented under the dual form of a homogeneous Markov chain and its associated superposition of Poisson processes. A chronicle model is then connected with a specific path in the state space of the Markov chain, and the timed relations will be provided by the corresponding superposition of Poisson processes.
To represent a sequence $\omega=(C^i_k)_{k\in K=\{0,\ldots,m\}}$ as a Markov chain $X=(X(t_k);\ k\in K)$, the set of discrete event classes $C^{\omega}=\{C^i\}_{i=0\ldots n-1}$ in ω is assimilated to with the state space $Q=\{i\}_{i=0\ldots n-1}$ of X. A binary sub sequence $\omega'=(C^i_{k-1}, C^j_k)$ of ω corresponds then to a state transition in X: $X(d(C^i_{k-1}))=i\rightarrow X(d(C^j_k))=j$, where d is the function providing the time of a class occurrence. A simple depth-first backward search algorithm (i.e. from an output class to the input classes) is used to generate the tree of the most probable paths that lead to an output class (Le Goc et al., 2006).
This tree and the matrix of transition probabilities are a first representation of the sequence of alarms. This result is interesting because, whatever the length of the sequence of alarms, it is entirely contained in a finite matrix. The tree of sequential relations can then be used to produce a functional model[2] of the couple (process, KBS) or to find signatures of the form of the equation 5.
To constitute a timed binary relation of the form $R(C^i, C^j, [\tau^-, \tau^+])$, the timed constraint $[\tau^-, \tau^+]$ is simply added to the sequential relation $R_s(C^i, C^o)$. Such a timed constraint is related with the average delay $D_{i\text{-}j}=E[d(C^j_k)\text{-}d(C^i_{k-1})]$ between two successive occurrences $o_{k-1}::C^i$ and $o_k::C^j$ in a specific $\omega_s^{i\text{-}j}$ sequence that contains only the occurrences of the two classes C^i and C^j of the sequence ω_s. The average delay D_{ij} between the occurrences of two classes C^i and C^j of ω is evaluated from two types of Poisson processes:

- A Poisson process $(N_{i\text{-}j}(t\text{-}t_{min});\ t\in T)$ that counts the number of sub sequences $\omega'=(C^i_{k-1}, C^j_k)$ in each $\omega_s^{i\text{-}j}$,
- A compound Poisson process $(N^D_{i\text{-}j}(t\text{-}t_{min});\ t\in T)$ associated to each Poisson process $(N_{i\text{-}j}(t\text{-}t_{min});\ t\in T)$.

The average delay D_{ij} is then given by (Le Goc et al., 2006):

$$D_{ij} = E\left[d\left(C_k^j\right) - d\left(C_{k-1}^i\right)\right] = \frac{1}{\lambda_{i-j}} = \frac{N_{i-j}^D\left(t_{\max} - t_{\min}\right)}{N_{i-j}\left(t_{\max} - t_{\min}\right)} \tag{9}$$

In our applications, the timed constraints are often intervals of the form $[0, 2/\lambda_{i\text{-}j}]$, which takes into account 60% of the occurrences[3].
These data structures and the associated algorithms have been implemented in a Java platform with a set of tools to help experts analysing sequences of phenomena. There are two algorithms linked with the stochastic approach: The BJT4T algorithm (Backward Jump with Timed constraints for Trees) and the BJT4S algorithm (Backward Jump with Timed constraints for Signatures) (Bouché et al., 2008b).

---

[2] A functional model is the description of all variables of the system and relations which can exist among these variables.
[3] See (Le Goc et al., 2006) to a more detailed explanation of this choice of timed constraints.

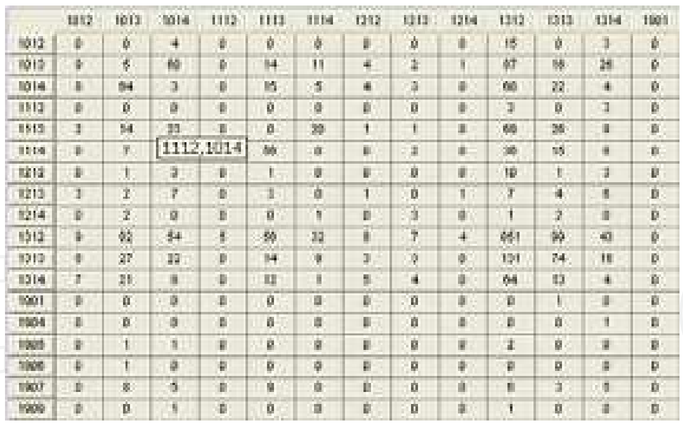
The role of the BJT4T algorithm is to compute the set of the most probable timed binary relations $R(C^i, C^j, [\tau^-, \tau^+])$ in a set Ω of sequences ω_i that leads to a specific discrete event class C^j. The BJT4S algorithm evaluates the anticipating ratio of each branch of the tree: the signatures are the branches of the tree having an anticipating ratio greater that an arbitrary threshold (we have made the choice to use 50%, see (Bouché et al., 2008b)).

6. Example: Identification of fault models on real data

In the following of this chapter, we will use data of a company that provides automotive parts (such as door locks). More precisely, we will show data from one production line of this company, even if we have all data on long periods, we will only use data on one week for this example.
In first place, we obtain the phenomena log (Fig. 21).

Fig. 21. Example of a log of a production line

Afterwards, we may carry out probabilistic studies, in order to identify if there exist correlations among phenomena and the temporal constraints on these correlations. The first step of the stochastic approach is, then, to build the transition matrix from the log of phenomena (Fig. 22).
The transition matrix is a counting matrix; we make the sum of transitions that we can observe from two phenomena in the log of phenomena. On the matrix of Fig. 22, the number 4 on the first row means that in the log of phenomena we have observed four transitions from a phenomenon 1012 to a phenomenon 1014.



	1012	1013	1014	1112	1113	1114	1212	1213	1214	1312	1313	1314	1901
1012	0	0	4	0	0	0	0	0	0	15	0	3	0
1013	0	6	60	0	14	11	4	2	1	97	16	25	0
1014	0	84	3	0	15	5	4	3	0	60	22	4	0
1112	0	0	0	0	0	0	0	0	0	3	0	3	0
1113	3	14	33	0	0	20	1	1	0	60	26	0	0
1114	0	7	1112,1014	[illegible]	66	0	0	2	0	30	15	0	0
1212	0	1	3	0	1	0	0	0	0	10	1	3	0
1213	3	2	7	0	3	0	1	0	1	7	4	6	0
1214	0	2	0	0	0	1	0	3	0	1	2	0	0
1312	0	92	54	5	50	32	0	7	4	651	90	40	0
1313	0	27	22	0	14	9	3	3	0	131	74	18	0
1314	7	21	8	0	13	1	5	4	0	64	13	4	0
1901	0	0	0	0	0	0	0	0	0	0	1	0	0
1904	0	0	0	0	0	0	0	0	0	0	0	1	0
1905	0	1	1	0	0	0	0	0	0	2	0	0	0
1906	0	1	0	0	0	0	0	0	0	0	0	0	0
1907	0	8	5	0	9	0	0	0	0	8	3	5	0
1909	0	0	1	0	0	0	0	0	0	1	0	0	0

Fig. 22. The transition matrix

As we have explained in section 0, the next step is the use of a representation under the dual form of a Markov chain model and a superposition of Poisson processes (Fig. 23). The transition matrix is used to compute the Markov matrix where we will have the probabilities of transition among phenomena. The transition matrix will also be used to determine, then, the time constraints among phenomena in the superposition of Poisson processes.

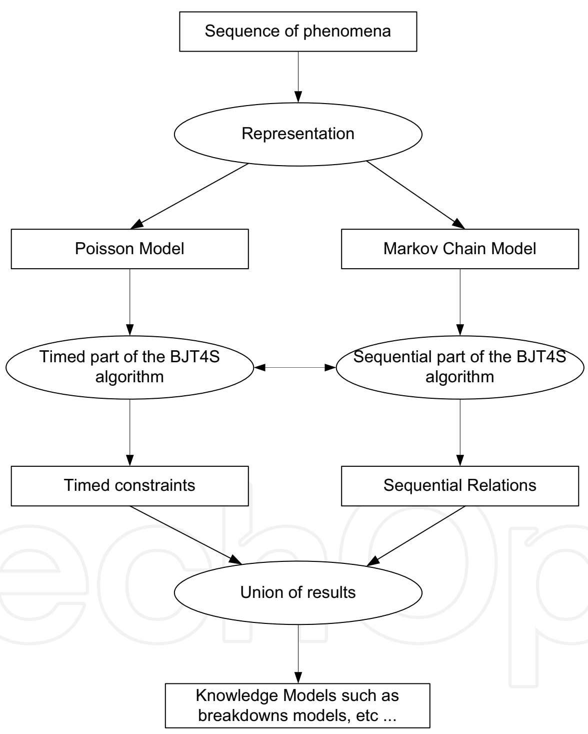


Fig. 23. The Stochastic Approach

The objective is to produce *behavioural models of breakdowns*. These models may be used to perform real time diagnosis, but also to define action plans and corrections on the production line. Fault models will probably reveal implicit links among the workstations of the considered line.

The application of the stochastic approach corresponds to the generic level of analysis of our project. The stochastic approach produces behavioural models that, according to our experience, are realistic indeed and can be used to make prediction or diagnosis.
Fig. 24 shows an example of correlation between two phenomena detected on workstations P210 and P220 (see the production line in Fig. 4).
We detect the *Slow_Working_Period* phenomenon on workstation P220 and the *Stock_Saturation* phenomenon on workstation P210. It is easy to see that, if there is an important working time on workstation P220, the line will be slowed down and a consequence is the saturation of stock that can be observed on workstation P210.
Therefore, to improve performance of the line in this context, the problem will not be to eliminate the *Stock_Saturation* phenomenon but to improve the working time on workstation P220. With a single action, we can act on two phenomena.

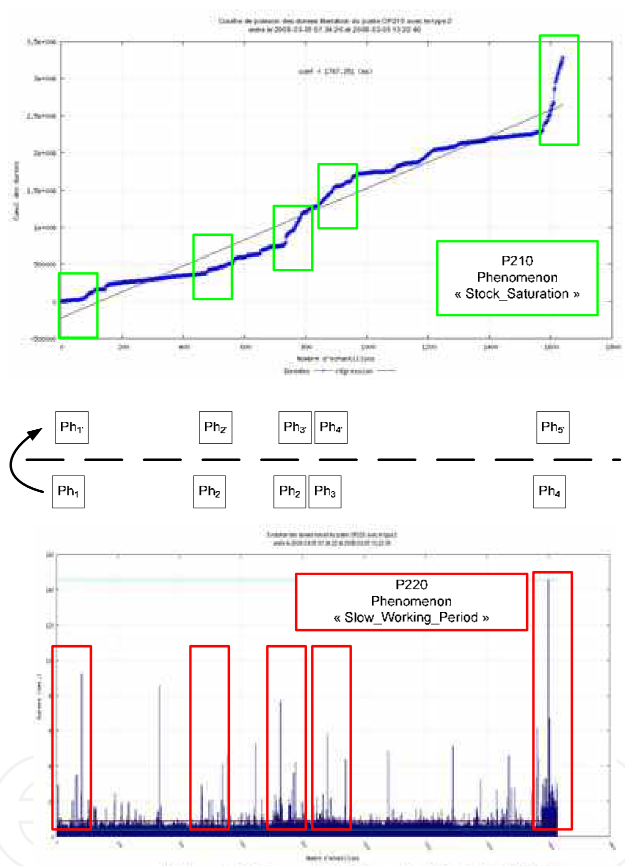


Fig. 24. Example of a correlation between two phenomena

While taking into account all the observations from the other workstations we realise that this binary relation can be generalized to all the production line to produce a global binary relation between two phenomena (Fig. 25):

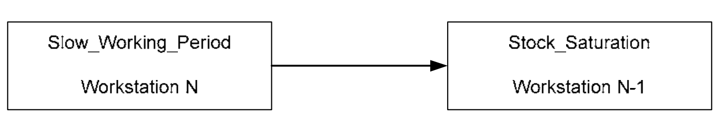


Fig. 25. Example of a binary relation between two phenomena

On our data, the stochastic approach permits to build the following tree of sequential relations (Fig. 26). The base of the tree (it is an arbitrary choice) is the phenomenon 1953, which corresponds to a *Failure_Level_2* phenomenon on workstation P140.

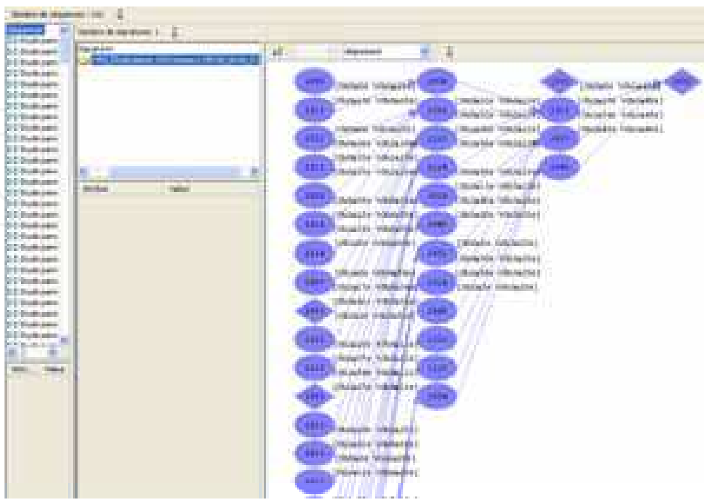


Fig. 26. Tree of sequential relations

From this tree, the BJT4S algorithm extracts the 1953 class signature in Fig. 27. A signature is the branch of the tree, which has an anticipating ratio greater than 50%.

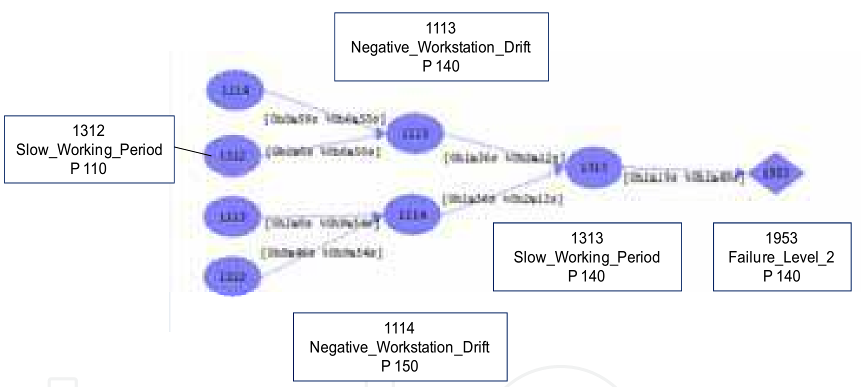


Fig. 27. Example of a signature or behavioural model

To explain how this model has to be read, let us consider the branch 1312 ⇒ 1113 ⇒ 1313 ⇒ 1953 in Fig. 27. This means that:

If we observe the occurrence of a *Slow_Working_Period* phenomenon on workstation P110

Followed by an occurrence of a *Negative_Workstation_Drift* phenomenon on Workstation P140 in the time interval [0s, 1m],

Followed by an occurrence of a *Slow_Working_Period* phenomenon on workstation P140 in the time interval [0s, 1m36s],

Then *there is a strong probability* to observe an occurrence of a *Failure_Level_2* phenomenon on workstation P140 in the time interval [0s, 1m19s].

The idea is, therefore, to use such a rule in a diagnosis task to make impossible the occurrence of such a succession of phenomena that leads to a severe failure. In this way, it is

possible to improve the performance of the production line, by preventing the occurrence of a serious phenomenon.
It is important to note that, by these means, we can identify relations among phenomena on different workstations. Therefore, we propose a global method of analysis that is not limited to the study of all workstations independently.
These models will be the base of the proposal of action plans to improve the performance of the production line in study. They can also be used on-line with a real time supervision system. In fact, the goal will be to observe phenomena on line, and to compare them with the knowledge base of fault models. If we detect the beginning of a model, we can generate an alarm to warn the operator on the risk of the occurrence of a total breakdown. In this way, corrective actions can be done before the occurrence of that eventual total breakdown.
Fault models can also be used to make new studies, by defining new phenomena with high abstraction levels.

7. Action plans

This project fits into the heuristic approach to knowledge-based diagnosis (Zanni et al., 2006). The basic assumption of this approach is that diagnosis is a heuristic process. It implies that experts rely on associational knowledge of the form *observations → faults*, that knowledge derives from experience with the device under consideration (a production system in our case) and that it can be elicited from domain experts.
The systems built under this approach can reach a high level of performance and may be very efficient in their reasoning.
However, what has to be done when we have identified a fault or an unsatisfactory state?
The goal of the supervisor of the production line is, precisely, to make the unsatisfactory state(s) disappear: an action is required when the state of the process is not satisfactory, and otherwise nothing has to be done.
Therefore, when required, the supervisor must decide and propose an action on the process.
An action is a modification of at least one of the input variables of the process. The causal relations between the variables will transform and propagate a modification of an input variable on the internal variables and ultimately on the output variables (Le Goc, 2004a).
To control the behaviour of a process, the relations linking causes to effects must be known.
The natural form of the expert knowledge is the "if-then" rule. Conceptually, the simpler form of such a relation is:

If the process is in the state X and
If the action U is executed (10)
Then the process produces the output Y

In this rule, the process output Y is the effect of applying a modification U. This relation depends on the process state X. The causal relation can be modelled, then, as a ternary predicate in first order logic of the form (Le Goc, 2004b) $C\ (Y, X, U)$.
In our case, the state X is the set of indicators measured by the data acquisition system and, particularly, the control parameters of the workstations.
Modifications of the set-up parameters of the workstations (whose set represents the action U) will produce changes in the state X that will be reflected as an output Y.

Our objective is to control the process behaviour.
To express this fact, it is necessary to reformulate the previous rules, by the introduction of a new term, the goal *G* of the supervisor (Le Goc, 2004a).

If the goal G is to obtain the output Y and
If the process is in the state X (11)
Then the action U must be carried out

Formally, the causal relation is now a 4-arity predicate *I(Y, U, X, G)* such that:

$$I(Y, U, X, G) \Longleftrightarrow C(Y, X, U) \wedge Equal(Y, G) \tag{12}$$

The pieces of knowledge *I (Y, U, X, G)* are the ones that we have elicited in the knowledge acquisition stage for the generic analysis. We have worked with experts on the development of an ontology of actions, with the associated phenomena. Fig. 28 shows a part of this ontology:

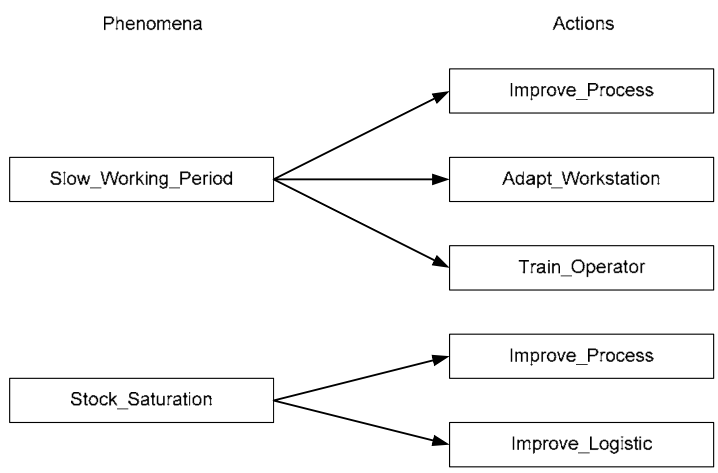


Fig. 28. A part of the ontology on relationships phenomena – actions

In this way, if we consider the model of a breakdown on Fig. 25, to improve the production line, we must act on the *Slow_Working_Period* phenomenon. Therefore, we will propose to improve the process, to adapt the workstation, or to train the operator.
The ontological study has been carried out by considering different sources. In particular, diverse ontologies on production systems or enterprise ontologies have been studied. Our main sources have been the MASON ontology (Lemaignan et al., 2006) and the one on Unified Assembly System Design developed by the University of Nottingham, the Royal Institute of Technology (Sweden) and the New University of Lisbon (Lohse et al., 2005)
Others ontologies we have considered are the TOVE (Fox, 1992; Fox & Grüninger, 1998) and the ENTREPRISE (Ushchold et al, 1998) ontologies.

8. Losses

To conclude, we remind that all this work has been done with the goal of improving the performance of the production lines. In fact, for our industrial partner *to improve performance of his production line* means *to produce more good pieces in a certain time window*.
Therefore, we can estimate the losses incurred (due to non-quality or to fault/breakdowns) in a given time period (Zanni & Bouché, 2008b).
Losses due to non-quality can be evaluated by the number of bad pieces produced per hour.

$$\begin{aligned} L_{quality} &= \textit{Total quantity of produced bad pieces / production time} \\ L_{quality} &= \textit{Number of bad pieces / hour} \end{aligned} \tag{13}$$

Losses due to production factors can be evaluated by the difference between the theoretical number of pieces that the line can produce (according to the specifications of the line) and the effective number of good pieces produced.

$$\begin{aligned} L_{production} &= \textit{(Theoretical number of pieces – Number of effectively} \\ &\qquad \textit{produced pieces) / production time} \\ L_{production} &= \textit{Number of non produced pieces / hour} \end{aligned} \tag{14}$$

We need to consider, also, the losses produced by maintenance problems. These losses may be evaluated as the total time of breakdown multiplied by the work pace of the line divided by the total production time. In this way, we have the number of non-produced pieces during breakdowns.

$$\begin{aligned} L_{maintenance} &= \textit{Breakdown periods * work pace of the production line /} \\ &\qquad \textit{production time} \\ L_{maintenance} &= \textit{Number of non produced pieces during break / hour} \end{aligned} \tag{15}$$

If we make the addition, we obtain a global estimation of losses:

$$\begin{aligned} \textit{Losses} &= L_{quality} + L_{production} + L_{maintenance} \\ \textit{Losses} &= \textit{Number of bad pieces / hour +} \\ &\qquad \textit{Number of non produced pieces / hour + Number of pieces non} \\ &\qquad \textit{produced during breakdowns / hour} \end{aligned} \tag{16}$$

The amount of these losses may be very important; on one of the lines we study, we have estimated losses in *185 pieces / hour*. This line should produce *450 pieces / hour*, that is, the losses represent more than 40%.
Estimation of losses will permit us to evaluate the pertinence of the proposed action plans. With this aim, we will implement a simulator of the production line, on which we can test them.

9. Conclusions

We have presented a global method based on a knowledge-based approach for the development of a software tool for modelling and analysis of production flows.

To the best of these authors understanding, the reasoning on the number of produced parts and the recommendations according to the three criteria, quality, maintenance and production, have not been fully addressed yet. In addition, the generic vs. specific analysis (global vs. by-workstation) approach will make the tool flexible and available for use by the production staff on site (not necessarily at ease with other possible performance indicators) and decision makers.

The method we propose is based on data processing and data mining techniques. Different kinds of techniques are used: graphic representation of the production, identification of specific behaviours to identify phenomena, and research of correlations among them on the production line. Most of these techniques are based on statistical and probabilistic analyses. To carry on high-level analyses, a stochastic approach is used to identify fault models.

Fault models can finally be used to propose action plans, which can be studied by simulation before implementation.

Therefore, the following steps of our project include the development of a simulator. We will use it to compute the effects of the action plan. The principle will be to build new sequences of data with the specifications of the action plan and to introduce them into real data to compute the effects. If a proposition of an action has no effect, it is not necessary to implement it. Furthermore, if the implementation of that action does not produce significant improvements (according to the decision maker) in the quantity of good pieces that were produced, a non-application of the action might be envisaged.

Our future works also include the possibility of exploring a new generation of expert system using multi-agents techniques for on-line analysis and diagnosis production chains.

The idea is to introduce, for each workstation in the line, an autonomous agent capable of monitoring its operation. To do this, it will generate the characteristic of the workstation behaviour, from statistics and probabilistic computations. Therefore, each workstation will be able to make its own diagnosis, based on its own behavioural models, but it will also be able to have a global view of the behaviour of the whole line through exchanges with the rest of the agents.

This communication among the agents will permit them to act together for the optimization of the operation of the line or to produce high-level alarms to prevent the occurrence of major failures.

10. References

Bouché, P. & C. Zanni. (2008a). A Stochastic Approach for Performance Analysis of Production Flows. *In ICEIS 2008, 10th International Conference on Enterprise Information Systems*, Barcelona, Spain.

Bouché, P.; Le Goc, M. & Coinu, J. (2008b). A global model of sequences of discrete event class occurrences. *In ICEIS 2008, 10th International Conference on Enterprise Information Systems*, Barcelona, Spain.

Cassandras, C.G. & Lafortune, S. (2001). *Introduction to dicrete event systems*. Kluwer Academic Publishers.

Fox, M.S. (1992). The TOVE Project: A Common-sense Model of the Enterprise, Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, Belli, F. and Radermacher, F.J. (Eds.), *Lecture Notes in Artificial Intelligence* # 604, Berlin: Springer-Verlag, pp. 25-34.

Fox, M.S. & Grüninger, M. (1998). Enterprise Modeling, *AI Magazine*, AAAI Press, Fall 1998, pp. 109-121.

Ishikawa, K. (1982). *Guide to Quality Control*. Unipub / Quality Resources.

Le Goc, M. (2004a). SACHEM, a Real Time Intelligent Diagnosis System based on the Discrete Event Paradigm. *Simulation*. The Society for Modeling and Simulation International Ed., vol. 80, n° 11, pp. 591-617.

Le Goc, M. (2004b). The discrete event concept as a paradigm for the perception based diagnosis of SACHEM. *Journal of Intelligent Systems* 8(3/4), 239–290.

Le Goc, M.; Bouché, P. & Giambiasi, N. (2006). Temporal Abstraction of Timed Alarm Sequences for Diagnosis. *In: the proceedings of COGIS'06, COGnitive systems with Interactive Sensors*, Paris, France.

Lemaignan, S.; Siadat, A.; Dantan, J.-Y. & Semenenko, A. (2006). MASON: A Proposal For An Ontology Of Manufacturing Domain. *Distributed Intelligent Systems: Collective Intelligence and Its Applications, 2006. DIS 2006*. IEEE Workshop on , vol., no., pp. 195-200, 15-16.

Lohse, N.; Valtchanov, G.; Ratchev, S.; Onori, M. & Barata, J.A. (2005). Towards a Unified Assembly System Design Ontology using Protégé. *In: 8th Intl. Protégé Conference*, Madrid, Spain.

Shewhart, W.A. (1939). *Statistical Method from the Viewpoint of Quality Control.* The Graduate School, U.S. Department of Agriculture, Washington, 1939.

Tempelbeier, H. & Burger M. (2001). Performance evaluation of unbalanced flow lines with general distributed processing times, failures and imperfect production. *IEE Transactions* 33(4), 419–446.

Uschold, M.; King, M.; Moralee, S. & Zorgios, Y. (1998). The Enterprise Ontology. *The Knowledge Engineering Review*, 13, pp 31-89

Van Bracht, E. (1995). Performance analysis of a serial production line with machine breakdowns. *In: IEEE symposium on emerging technologies and factories automation.* Paris, France.

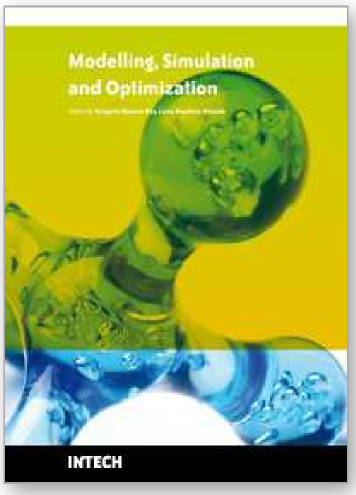
Xie, X. (1993). Performance analysis of a transfer line with unreliable machines and finite buffers. *IEE Transactions* 25(1), 99–108.

Zanni, C.; Le Goc, M. & Frydman, C. (2006). A conceptual framework for the analysis, classification and choice of knowledge-based diagnosis systems. *International Journal of Knowlegde-Based & Intelligent Engineering Systems* 10(2), 113–138.

Zanni, C.; Barth, M. & Drouard, L. (2007). A Knowledge-Based Tool for Performance Analysis of Production Flows. *IFAC MCPL 2007 – The 4th International Federation of Automatic Control Conference on Management and Control of Production and Logistics*, Sibiu, Rumania.

Zanni, C. & Bouché, P. (2008a). A Global Method for Modelling and Performance Analysis of Production Flows. *In EUROSIM/UKSIM 2008 10th International Conference on Computer Modelling and Simulation*, p740-745, Emmanuel College, Cambridge, England.

Zanni, C. & Bouché, P. (2008b). A Stochastic Approach to Improve Performance of Production Lines. *In EMSS 2008 20th European Modeling and Simulation Symposium (Simulation in Industry)*, Campora San Giovanni, Amantea (CS), Italy.



Modelling Simulation and Optimization

Edited by Gregorio Romero Rey and Luisa Martinez Muneta

ISBN 978-953-307-048-3
Hard cover, 708 pages
Publisher InTech
Published online 01, February, 2010
Published in print edition February, 2010

Computer-Aided Design and system analysis aim to find mathematical models that allow emulating the behaviour of components and facilities. The high competitiveness in industry, the little time available for product development and the high cost in terms of time and money of producing the initial prototypes means that the computer-aided design and analysis of products are taking on major importance. On the other hand, in most areas of engineering the components of a system are interconnected and belong to different domains of physics (mechanics, electrics, hydraulics, thermal...). When developing a complete multidisciplinary system, it needs to integrate a design procedure to ensure that it will be successfully achieved. Engineering systems require an analysis of their dynamic behaviour (evolution over time or path of their different variables). The purpose of modelling and simulating dynamic systems is to generate a set of algebraic and differential equations or a mathematical model. In order to perform rapid product optimisation iterations, the models must be formulated and evaluated in the most efficient way. Automated environments contribute to this. One of the pioneers of simulation technology in medicine defines simulation as a technique, not a technology, that replaces real experiences with guided experiences reproducing important aspects of the real world in a fully interactive fashion [iii]. In the following chapters the reader will be introduced to the world of simulation in topics of current interest such as medicine, military purposes and their use in industry for diverse applications that range from the use of networks to combining thermal, chemical or electrical aspects, among others. We hope that after reading the different sections of this book we will have succeeded in bringing across what the scientific community is doing in the field of simulation and that it will be to your interest and liking. Lastly, we would like to thank all the authors for their excellent contributions in the different areas of simulation.

51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821