



HAL
open science

A rollingstock door system's dynamic maintenance strategies based on a sensitivity analysis through bayesian networks

Rony Rozas, Laurent Bouillaut, Patrice Aknin, Allou Same, Olivier Francois,
Guillaume Branger

► **To cite this version:**

Rony Rozas, Laurent Bouillaut, Patrice Aknin, Allou Same, Olivier Francois, et al.. A rollingstock door system's dynamic maintenance strategies based on a sensitivity analysis through bayesian networks. 10th World Congress on Railway Research (WCRR 2013), Nov 2013, Sydney, Australia. 8p. <hal-00863687>

HAL Id: hal-00863687

<https://hal.science/hal-00863687v1>

Submitted on 4 Oct 2023

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

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



HAL Authorization

A rollingstock door system's dynamic maintenance strategies based on a sensitivity analysis through bayesian networks

R. Rozas¹, L. Bouillaut¹, P. Aknin¹, A. Samé¹, O. François¹, G. Branger²

¹ Université Paris-Est, IFSTTAR, GRETTIA, 14-20 Boulevard Newton, Cite Descartes Champs sur Marne F-77447 Marne la Vallée Cedex 2

² Bombardier Transportation Place des Ateliers – BP 1 59154 Crespin

Abstract

In most industrial fields, and particularly in the railway industry, the optimization of maintenance policies has become a key issue. Dynamic Bayesian networks (DBN) have been proved as relevant to perform reliability analysis as they can easily represent complex systems behaviors. Based on this formalism, graphical duration models (GDM) were developed by (Donat, et al., 2009) to set all kind of sojourn time distributions for each state of the system. Unlike to some Markovian approaches that impose exponential behavior, this approach could better model the exact degradation dynamic of real industrial systems

But, what commonly happens when the degradation process suddenly changes? The operator has to face with an unexpected increasing number of severe defects (and then a strong drop of its availability). These changes are generally due to either new component, introduced in the system for obsolescence reasons, or to changing operating conditions.

The aim of the study introduced in this paper, focusing on Dynamic Maintenance Strategies, is to detect these drifts and to evaluate their impacts on the system's behavior.

1. Introduction

Reliability analysis is an integral part of system design and operating. Moreover it can be an input to optimize maintenance strategies. Recently, Dynamic Bayesian Networks (DBN) have been proved relevant to represent complex systems and perform reliability studies. The major drawback of this approach comes from the constraint on the sojourn times which are necessarily exponentially distributed, as in usual Markovian approaches. To avoid this constraint, a new DBN structure named Graphical Duration Models (GDM) was proposed (Donat, et al., 2009). Based on this formalism, a generic methodology was developed providing decision support tools for the evaluation, comparison or optimization of maintenance strategies. This approach named *VirMaLab* (virtual maintenance laboratory) allows considering systematic preventive maintenance actions as well as condition based maintenance actions (Bouillaut, et al., 2011) and was successfully applied to several railway applications.

In all these development, the aim was to obtain the optimal parameters satisfying a given set of constraints. But, what commonly happen when, for different reasons (such as obsolescence, operating changes, users' behavior changes...), the degradation process of the considered system changes? The optimal solution can change and operators generally become aware of this fact only by facing with an unusual number of failures (and all their consequences...).

The aim of the study introduced in this paper is to propose an extension of the generic *VirMaLab* approach that allows detecting system's behavior drifts through new REX data. Then, the proposed algorithm will have to evaluate if these drifts impact the optimality of used parameters and, if so, evaluate the new set of optimal parameters.

To address these aims, various problems had to be solved such has the drifts detection, the drifts impact evaluation, the computation time... The proposed solutions are briefly introduced in this paper and, finally, the *Dynamic maintenance strategy* algorithm is applied to the Bombardier Transport rollingstock doors system. This study takes part in the project *Surfer* (active railway monitoring), managed by the Bombardier Transport engineering department.

2. Framework

In this section, the *VirMaLab* approach is very briefly introduced. It is based on a modular modeling of complex systems, and provides decision support stochastic tools for evaluating comparing and optimizing various set of maintenance, diagnosis, or operating parameters...

The proposed approach for building such decision support tools is divided into three main steps:

- modeling the degradation process

- modeling the diagnosis and maintenance strategy
- optimizing maintenance parameters

Then, with such decision support tool, one can evaluate various maintenance strategies and determine, for a given cost functions, the best set of maintenance and diagnostic parameters. It can be applied to simple systems but also to multi-states and multi-components systems (with eventually interacting components). The learning of such modeling can be done with both expert advices and REX databases.

More details on *VirMaLab* can be found in (Bouillaut, et al., 2011) and some industrial applications in (Bouillaut, et al., 2011), (Bouillaut, et al., 2012).

Dynamic Bayesian networks (DBN) have been proved as relevant to perform reliability analysis as they can easily represent complex systems behaviors. This is the reason why this formalism was chosen to develop the *VirMaLab* approach. The following sections introduce a specific DBN proposed to model degradation processes for multi-component and multi-state system.

2.2. The considered formalism: Bayesian Networks

BN are mathematical tools relying on both the probability theory and the graph theory (Aven & Jensen, 1999). They allow to qualitatively and quantitatively representing uncertainty. Basically, BN are used to compactly describe the joint distribution of a collection of random variables $X = (X_1, \dots, X_n)$ taking their values in $\mathcal{X} = \{X_1, \dots, X_m\}$.

Formally, a BN denoted by \mathcal{M} is defined as a pair where $G = (\xi, \varepsilon)$ is a Directed Acyclic Graph (DAG) and $\{p_n\}_{1 \leq n \leq N}$ a set of Conditional Probability Distributions (CPD) associated with the random variables. These distributions aim to quantify the local stochastic behavior of each variable. The graph nodes and the associated random variables being both represented by $\xi = \{X_1, \dots, X_N\}$, ε is the set of edges encoding the conditional independence relationships among these variables. Finally, G is said to be the qualitative description of the BN.

Besides, both the qualitative (i.e. G) and quantitative (i.e. $\{p_n\}$) parts can be automatically learnt, if some complete or incomplete data or experts opinions are available (Jordan, 1999).

Using BN is also particularly interesting because of the easiness for knowledge propagation through the network. Indeed, various inference algorithms allow computing the marginal distribution of any sub-set of variables. The most classical one relies on the use of a junction tree (Cozman & Moral, 2000).

Finally, note that such modeling is able to represent dynamic systems (e.g. which contain variables with time dependent distributions) via the DBN solution (Murphy, 2002). A DBN is a convenient extension of the BN formalism to represent discrete sequential systems. Indeed, DBNs are dedicated to model data which are sequentially generated by some complex mechanisms (time-series data, bio-sequences, number of mechanical solicitations before failure...). It is therefore frequently used to model Markov chains.

Figure 1 illustrates this property, introducing a DBN modeling the Markov Chain of the sequence $X = (X_1, \dots, X_n)$ taking its values in the set \mathcal{X} . This DBN is described by the probabilities that quantify the transitions from one state of X to another.



Figure 1 : Dynamic Bayesian Network modeling a Markov chain

More precisely, a DBN defines the probability distribution of a collection of random variables $(X_t)_{t \geq 1} = (X_{1,t}, \dots, X_{n,t})_{t \geq 1}$ where t is the discrete time index.

Considering condition based maintenance approach, during a *VirMaLab* modeling a key point lead in the quality of the first step, i.e. the degradation process modeling. For this reason, a particular effort was made on this point to provide the modeling as close as possible from the exact behavior of the system. Many approaches are based on the Markovian approach that imposes exponential sojourns time distribution for each state of the system. This can be quite far from the reality. To overcome this drawback, a specific BN was proposed: the Graphical Duration Model (GDM) (Donat, et al., 2009).

2.3. Graphical Duration Models

The Graphical Duration Model is a specific DBN, using semi-Markov models. The main idea is the introduction of remaining time variables into the graph that allows to model multi-state systems featuring complex sojourn times. Figure 2 shows a GDM in its DBN form.

The red lines characterize dependencies between time slices, and the black lines represent dependencies of variable in a same slice. The model handles two kinds of variable:

- $(X_t)_{1 \leq t \leq T}$ represents the system state over a sequence of length T .
- $(X_t^D)_{1 \leq t \leq T}$ represents the remaining time before a system state modification (remaining sojourn time). These variables are called duration variables.

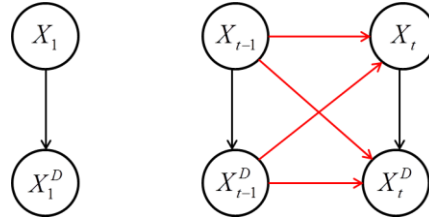


Figure 2 : Graphical Duration Model in the form of a Dynamic Bayesian Network

Besides, the DAG of a GDM shows that the current system state x_t depends on the previous system state x_{t-1} , the previous remaining duration x_{t-1}^D . On the other hand, the current duration variable x_t^D is dependent on the previous duration variable x_{t-1}^D , the current state x_t and, optionally, on the previous state x_{t-1} .

To complete this degradation model, for each node in the graph, a conditional probability distribution (CPD) must be defined. Considering the structure of the GDM (Figure 1), four CPD must be defined:

- The initial distribution of system's state $p(X_1)$
- The initial distribution of the remaining duration given a state $p(X_1^D | X_1)$
- The distribution of the remaining duration given a previous value of this variable and a previous state $p(X_t^D | X_{t-1}, X_{t-1}^D)$
- $p(X_t | X_{t-1}, X_{t-1}^D)$

On the practical point of view, this approach allows specifying arbitrary state sojourn time distributions by contrast with a classic Markovian framework in which all durations have to be exponentially distributed. This modeling is therefore particularly interesting as soon as the question is to capture the behavior of a given system subjected to a particular context and a complex degradation distribution. More details on this GDM (quantitative description, optional context description ...) can be found in (Donat, et al., 2009) (Bouillaut, et al., 2009).

3. Dynamic maintenance strategies

3.1. Change detection in the degradation process

In the framework of a GDM degradation process modeling, the estimation of the model parameters (i.e. the CPD for all variables) is made from natural transition of the considered system. A natural transition is obtained by letting the system grow old without maintenance until the failure. The result is a sequence of system states and sojourn time associated. From such data the method of maximum likelihood estimation can be applied.

Practically, r observations are initially known and used to learn the degradation model (characterized by a distribution p_0). Then, due to the system's monitoring devices, periodically, new REX data are available. Through these new data, the goal is to determine whether the degradation model must be updated or not. In other words, given a fixed length sequence containing the n observations X_1, \dots, X_n , we want to determine whether or not it contains a change point characterizing the initialization of a degradation dynamic drift. If no change point exists, the observations are independent and identically distributed (i.i.d.) according to the initial distribution p_0 . If a change point exists at time t_p , then the observations can be clustered: before t_p , REX data follow the distribution p_0 and after this point, they

follow a new distribution p_1 . The problem is equivalent to choosing between the following two hypotheses:

$$\begin{cases} (H_0) & X_i \sim p_0 \quad \forall i = 1, L, n \\ (H_1) & X_i \sim p_0 \quad \forall i = 1, L, t_p - 1 \\ & X_i \sim p_1 \quad \forall i = t_p, L, n \end{cases} \quad (1.1)$$

3.1.1 Algorithm of change detection

The proposed algorithm uses a sliding window of size N on the available data in such a way that, each time, the end of the window corresponds with the last obtained recording. Every time the window slides (i.e. new REX data are available), the change detection algorithm tries to split the window into two set of observations such as if the first set follows a distribution p_0 then the second set follows a distribution p_1 with $p_0 \neq p_1$. To do this, all potential change points are evaluated with a distance d between the two sets of both sides of this point. If the distance d is higher than a threshold c then the corresponding point of potential change is validated as a breaking point.

The metric distance is chosen according to the information available on the distribution p_0 and p_1 . For example, if the observations are assumed to be Gaussian, then it would be appropriated to use a two sample Student-t statistic to detect a mean shift, and Fisher statistic to detect a scale shift. To avoid making such distributional assumptions, a nonparametric test like Kolmogorov-Smirnov statistic (Frank & Massey, 1951) can be used. In most practical situations the parameters of these distributions will be unknown. In this paper, we are dealing with nonparametric change-point detection methods, without any priori information about data.

3.2 Evaluate change impact

When a change in the degradation process model is detected, it is necessary to know if this change impacts the “quality” of the maintenance strategy. In fact to evaluate the “quality” of maintenance policy some indicators are defined (system’s availability, quality of service, maintenance costs...), whom values depend on the system’s state (operation, faulty, failed...) and then on the degradation process model variables. In respect of some given constraints on these indicators, the maintenance strategy can be adjusted or not. Unfortunately the assessment of these indicators for a maintenance strategy is generally expensive in terms of computation time. In this section a new way to reduce the time inference in introduced.

3.2.1 Algorithm of breaking points impact evaluation

It is assumed that at each time t , system, diagnosis devices, and maintenance actions, respectively generate a utility U_t according to probabilities $p(X_t)$, $p(D_t)$ and $p(A_t)$. Given a temporal horizon T , the complete assessment of a scenario is calculated by aggregating all the utility U_t as follow:

$$U = \bigoplus_{1 \leq t \leq T} U_t$$

To assess a maintenance policy, it is therefore necessary to compute for all $t \in [1, T]$ the marginalized distribution $p(X_t)$, $p(D_t)$ and $p(A_t)$. The computation of these probabilities goes through some inference calculations using a specific algorithm developed for GDM (Donat, 2009). This algorithm uses the specific features of a GDM and allows reducing the computation time compared to the general methods of inference in DBN. However, the inference time is still too large to be effective in a dynamic algorithm.

A new way to do inference in GDM was therefore necessary. In this algorithm, a vector $\Theta_t = (\theta_1, \dots, \theta_n)$ containing all the parameters of CPD characterizing the GDM is defined. Using the symbolic inference (D’ambrosio, 1994), some analytic expressions giving $p(X_t)$, $p(D_t)$, $p(A_t)$ and Θ_{t+1} in respects of Θ_t are established. Once these expressions determined, for given GDM parameters, the assessment of a maintenance policy is carried out by simply replacing symbols $(\theta_1, \dots, \theta_n)$ with their respective values.

In practice, the analytical expressions used in this algorithm are exponential in size compared to the size of the vector Θ_t and their establishment is a bit more expensive in computation time than inference with only numeric values. However, once expressions need to be learned once and then, all evaluations of the impact of any degradation dynamic drift can be done with negligible computation time.

4. Case study: rollingstock doors maintenance

In a context of a constant passenger traffic increase, railway operators identified the doors system unavailability as a key point to optimize. Indeed, the least incident on rollingstock doors can have strong consequences on traffic steadiness, infrastructure availability, users' safety... In an effort to improve the availability of its rolling stock, the manufacturer Bombardier Transport® aims to improve its maintenance strategies (commonly based on a compromise between corrective maintenance and preventive systematic maintenance actions). The case study, introduced in this paper, takes place in the research project *SurFer* (active railway monitoring) aiming to propose new tools, algorithms... to optimize the availability of the main rollingstock functions (doors system introduced in figure 3, braking, air conditioning...). These developments are based on the use of a new dynamic monitoring system providing real time diagnosis data.



Figure 3: Doors system, aim of the introduced case study

In this context, to determine the optimal systematic maintenance period, a *VirMaLab* model was established, highlighting the four key elements to be followed in the doors system's degradation process.

The following picture introduces the proposed *VirMaLab*. It is divided into three main blocks.

- The first one is dedicated to modeling the various considered degradation processes. In this study, 4 degradations were taken into account: The "V" (the gap between doors used to be adjusted avoiding they form a "V" that might lead to an obstacle detection failure. It is described by the variable V), the screw home adjustment drift (variable H), the roller adjustment (variable R) and the doorstep height (variable D). Due to its degradation process higher complexity, the first one was modeled by a Graphical Duration Model. The other degradation processes are simpler and can therefore be modeled by single Markov chains (i.e. constant failure rates and exponentially distributed sojourn times).
- The second block is dedicated to systematic maintenance actions modeling with two variables: P encodes the maintenance period and M describes de maintenance action ("do nothing" if the various components are supposed correctly adjusted or if the systematic maintenance period is not reached or, on the other hand, "adjust").
- Finally, the last block provides some indicators useful for maintenance evaluation and optimization: ϕ evaluates the risk that the obstacle detection function fails while A estimates the availability of the door opening function.

The variable v describes the distance between the two upper doors. It takes its values among four states: "A" (<0, doors form a "A"), "N" (between 0 and 2mm, standard adjustment), "L" (between 2 and 5mm, light "V") and "V" (>5mm, doors form a "V").

H takes its value among three states : "Weak", "OK" and "Strong" while R is characterized by four states: "Weak", "Adjustment Required", "OK" and "Strong".

Finally, the doorstep height D can either in the state "OK" or "Damaged".

The indicators are defined according to Bombardier Transport experts' advices that provide a decision matrix quantifying when variable ϕ and A are in one or another of their states. On the whole, the obstacle detection failure (variable ϕ) happens when R = "Strong". It takes its values among "OK", "very weak", "weak" and "strong".

The availability of the door function fails when either R = "weak" or D = "Damaged". The variable A is characterized by both states "OK" and "failed".

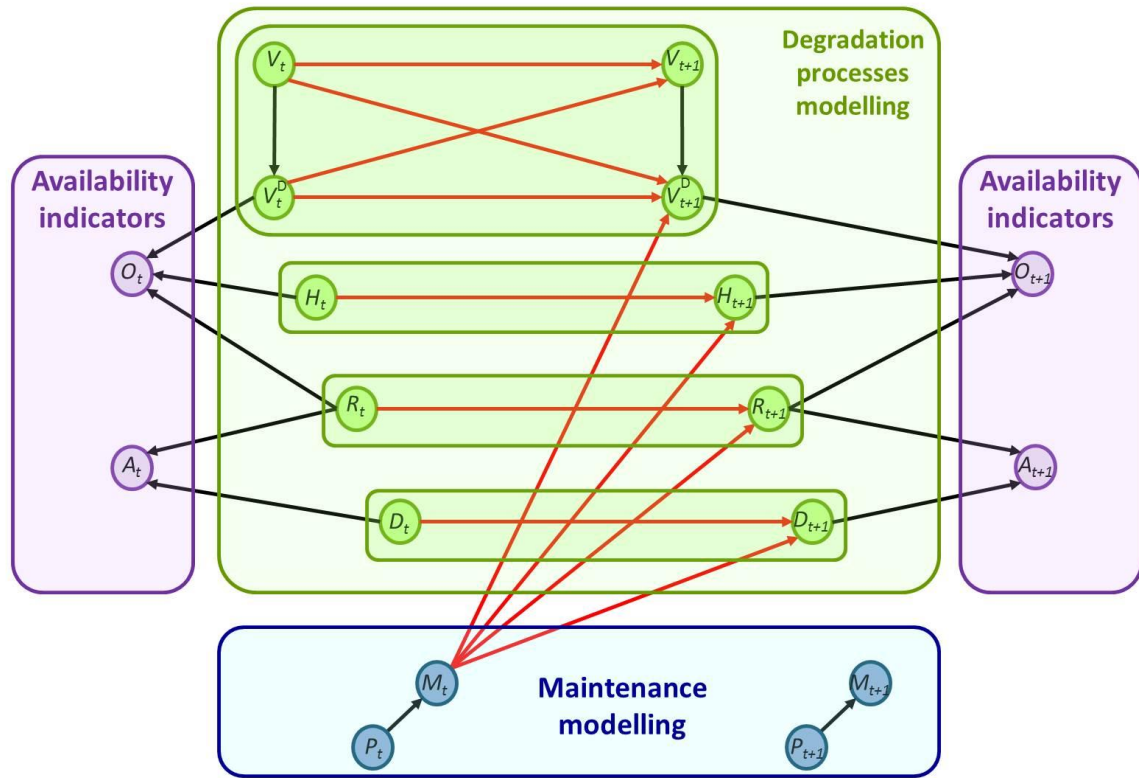


Figure 4: *VirMaLab* structure network for the rollingstock door system's maintenance

3.2.2 Simplified model

For some industrial reasons, the whole obtained results can't be published in this paper. Nevertheless, to illustrate the proposed method, results are given for the system limited to the variable "V" taking its value in {ok, damaged, failed}. Only the variables referring to the variable "V" are kept in this simplified model. Maintenance operations are done periodically every d months. The goal is to find the largest period d that would satisfy the given constraints on system availability. Figure shows the reduced model. The variables $(V_t)_{1 \leq t \leq T}$, $(V_t^D)_{1 \leq t \leq T}$, $(P_t)_{1 \leq t \leq T}$ and $(M_t)_{1 \leq t \leq T}$ represent respectively the system state, the remaining time before a system state change, the maintenance period, and the maintenance action executed over a time sequence of length T .

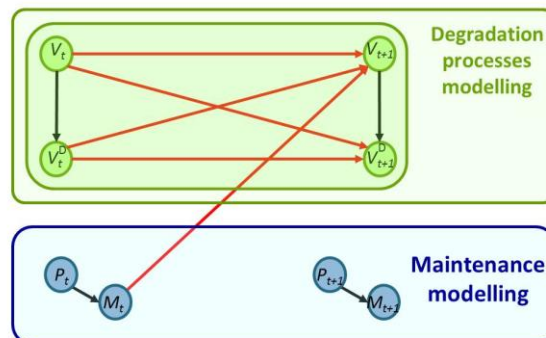


Figure 5: Simplified *VirMaLab* model limited to variable V

shows an example of data that can be used to learn the CPD of the variable V_t^D . It gives the time spent in the state "ok" for various observations. These data were generated randomly according to four different distributions to introduce a drift. In our case these data are used to learn the probability

distribution $p(V_t^D | V_t = ok)$. To simplify the example, the distribution $p(V_t^D | V_t = damaged)$ remains unchanged. Without loss of generality, the distribution $p(V_t^D | V_t = failed)$ is defined as follow:

$$p(V_t^D | V_t = failed) = \begin{cases} 1 & \text{if } t = 1 \\ 0 & \text{otherwise} \end{cases}$$

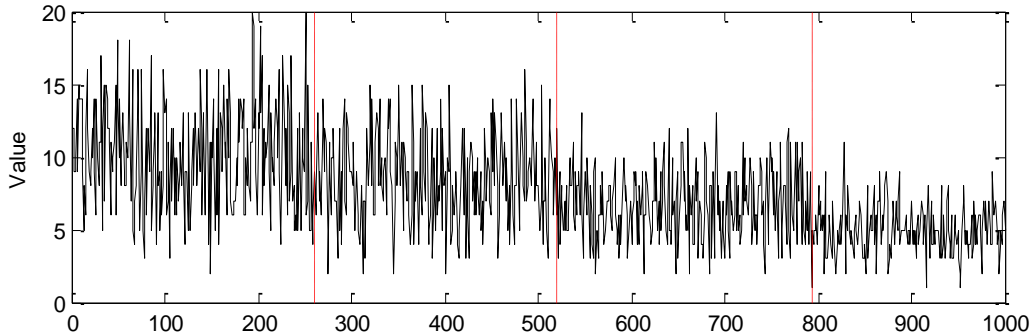


Figure 6: Time recoded before the first fault occurs

Using the previously introduced changes detection algorithm, three breaking points were detected at times $t_a = 245$, $t'_a = 500$ and $t''_a = 753$. When changes are detected, the parameters of the GDM are re-learned, and dynamic optimization method, detailed in section 3, is applied.

The utility function is defined as follows:

$$U = \min_{1 \leq t \leq T} p(X_t \neq \text{failed})$$

Under the condition $U \geq 0.95$, we proceed to an exhaustive search for the higher diagnosis period.

Table 1 presents utility U of the evaluated maintenance strategies (obtained for various values of d). Before any change appears in the data, the best period d satisfying $U > 0.95$ is $d=9$. When the first change is detected, the utility with the current optimal parameter doesn't satisfy anymore the constraint. The new optimal value becomes $d=8$. After the second breaking point t'_a , the maintenance period must be reduced 7. Finally, since the third breaking point does not have impact on the system's availability, the optimal maintenance parameter $d = 7$ is confirmed.

| Maintenance period | Utility before change | Utility after t_a | Utility after t'_a | Utility after t''_a |
|--------------------|-----------------------|---------------------|----------------------|-----------------------|
| 6 | - | - | - | - |
| 7 | - | - | 0,9566 | 0,9729 |
| 8 | - | 0,9538 | 0,8884 | 0,9238 |
| 9 | 0,9781 | 0,8993 | 0,7667 | 0,8259 |
| 10 | 0,9454 | 0,7999 | 0,6064 | 0,6923 |
| 11 | 0,9024 | 0,6728 | 0,4262 | 0,5346 |

Table 1: Obtained results

4. Conclusion

In this paper a *Dynamic Maintenance* algorithm is introduced. It aims to propose an extension of the generic *VirMaLab* approach that allows detecting system's behavior drifts through new REX data. These detections are made by Kolmogorov-Smirnov statistic tests on the duration spent in the OK state.

Then, the proposed algorithm evaluates if these drifts impact the optimality of maintenance parameters and, if so, evaluate the new set of optimal parameters. To overcome time computation drawbacks induced by standard Bayesian networks inference algorithms, a solution was proposed using the symbolic inference allowing, after determining the analytic expressions linking degradation parameters to maintenance indicators, to compute instantaneously the wished probabilities.

Finally, the *Dynamic maintenance strategy* algorithm is applied to the Bombardier Transport rollingstock doors system. This study takes part in the project *Surfer* (active railway monitoring), managed by the Bombardier Transport engineering department. Even if, for some industrial reasons,

the whole obtained results could not be introduced in this paper, a simplified system (limited to one of the 4 considered door's unavailability contributors) was detailed to focus on the ability of the proposed algorithm to adjust dynamically optimal maintenance parameters.

We are currently working on the deployment of a set of statistic tests (in place of the single Kolmogorov-Smirnov test) ensuring the detection of the degradation process drifts whatever dynamic these drifts follow.

Bibliographie

Aven, T. & Jensen, U., 1999. *Stochastic Models in Reliability*. s.l.:Springer.

Bouillaut, L., Aknin, P., Donat, R. & Bondeux, S., 2011. Virmlab:A generic approach for optimizing maintenance. *9th World Congress on Railway Research (WCRR)*.

Bouillaut, L., François, O. & Dubois, S., 2012. Optimal Metro-Rail Maintenance Strategy using Multi-Nets Modeling. *International Journal of Performability Engineering*, 8(1), pp. 77-90.

Bouillaut, L., Donat, R., Neji, A. & Aknin, P., 2009. Estimation of Multi-components System's reliability: Comparison of two Graphical Model Approaches. *13th IFAC Symposium on Information Control Problems in Manufacturing*.

Cozman, F. G. & Moral, S., 2000. Reasoning with imprecise probabilities. *Int. J. approx. Reasoning*, pp. 121-123.

D'ambrosio, B., 1994. Symbolic Probabilistic Inference in Large BN20 Networks. *UAI*, pp. 128-135.

Donat, R., 2009. *Phdthesis - Modélisation de la Fiabilité et de la Maintenance par Modèles Graphiques Probabilistes - Application à la Prévention des Ruptures de Rails*. INSA de Rouen: s.n.

Donat, R., Leray, P., Bouillaut, L. & Aknin, P., 2009. A Dynamic Bayesian Network to represent Discrete Duration Models. *Neurocomputing*, pp. 570-577.

Frank, J. & Massey, J., 1951. The Kolmogorov-Smirnov Test for Goodness of Fit. *Journal of the American Statistical Association*, 46(253).

Jordan, M., 1999. Learning in graphical models. *MIT Press*.

Murphy, 2002. *Phdthesis - Dynamic Bayesian Networks : Representation, Inference and Learning*. Berkeley: University of California.