International Journal of Engineering & Technology, 13 (1) (2024) 143-155

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Failure Prognosis using Dynamic Bayesian Networks and Decision Trees

Elie Djomo[1*], Wolfgang Nzié[1,2] and Aslain Brisco Ngnassi Djami[2]

[1]*Department of Mechanical Engineering, National School of Agro-Industrial Sciences, University of Ngaoundéré, Ngaoundéré, Cameroon*
[2]*Department of Fundamental Sciences and Techniques of Engineer, Chemical Engineering and Mineral Industries School, University of Ngaoundéré, Ngaoundéré, Cameroon*
[*]*Corresponding author E-mail: eliedjoel@gmail.com*

Abstract

Production managers commonly need to assess the reliability of production equipment throughout its life cycle. Since equipment is typically composed of multiple components, failures may stem from dependencies due to interactions. We have found that the literature does not adequately address these dependencies in the dynamic reliability models of the studied systems. The purpose of this study is to utilize decision trees (DT) and Dynamic Bayesian Networks (DBN) to approximate the failure probability of multi-component systems. We present a failure prediction method and utilize a parametric estimation approach grounded in a priori laws. The DBN learning process is assessed via a Monte-Carlo simulation method and a parametric compliance examination. Finally, this study illustrates the significant impact of using Dynamic Bayesian Networks (DBNs) in conjunction with Decision Trees (DTs) to evaluate the state of a water production system.

Keywords: *Prognosis, Probability of failure, Decision trees, Dynamic Bayesian networks, System.*

1. Introduction

The rising intricacy of production systems is becoming increasingly evident. Aside from the limitations tied to competitiveness, extant environmental protection laws are compelling manufacturers and the scientific community to devise answers that can enhance the efficiency, competitiveness, and safety of systems. One area of research happening in this regard is predictive maintenance, where research is being conducted on failure prognosis. Failure prognosis requires the assessment of a system's reliability over time, which has been undertaken by various authors using different prognostic methods such as physical models, experience-based methods, and data-based techniques[4]. For data-based methods, tools such as decision trees[9], expert systems, hybrid systems[6], neural networks[12], Decision Trees (DT)[11], and Bayesian Networks [13] are available. Bayesian networks can be categorized into three types, each with a specified use. These include static Bayesian networks, commonly used for diagnosis, naive Bayesian networks, used for classification, and dynamic Bayesian networks, utilized for prognosis and recognition purposes [2, 5]. While BNs are an effective tool for fault prognosis, they have limitations when it comes to determining the a priori laws of individual nodes. To address this challenge, we have integrated decision trees with RB due to their proficiency in swiftly deriving probabilities when data is accessible. Our purpose is to devise a method for anticipating the collapse of manufacturing systems by evaluating the interdependence among constituents and their conduct over time. This manuscript is structured as follows: Section 1 introduces Bayesian networks and Section 2 discusses decision trees. Then, in Section 3, we present a procedure for predicting failures. This procedure will be illustrated using an example of a water production system in Section 4. Finally, we will have a discussion in Section 5 and a conclusion.

2. Bayesian Network

2.1. Static Bayesian Network

A discrete Bayesian network, denoted by $B = (G, (p_n)_{1 \leq n \leq N})$, is mathematically defined as follows:

- Let G be a directed graph without circuits: G=(X,E) with X comprising n nodes, each linked to a random variable X_i in a finite and countable space Ω_{X_i}, and E representing the dependencies between said variables.
- $(p_n)_{1 \leq n \leq N}$ is a set of conditional probability distributions where p_n represents the CPD associated with the random variable X_n given its parents X_{pa_n}, where pa_n refers to the indices of the parent variables of X_n in G. $(p_n)_{1 \leq n \leq N}$ is a set of conditional probability distributions where p_n represents the CPD associated with the random variable X_n given its parents X_{pa_n}, where pa_n refers to the indices of the parent variables of X_n in G. The technical term abbreviations will be explained when first used.

2.2. Dynamics Bayesian Networks

When the variables in a static Bayesian network change over time, an extension called a dynamic Bayesian network (DBN) has been proposed [7]. The purpose of DBNs is to model the probability distribution of a sequence of variables $(X_t)_{1 \leq t \leq T} = (X_{1,t}, ..., X_{N,t})_{1 \leq t \leq T}$ for a duration of $T \in \mathbb{N}^*$. steps.
It consists of a pair of Bayesian networks $(B_1, B_{\longrightarrow})$:

- B_1 defines the a priori distribution, the initial model $P(X_{1,1}, ..., X_{N,1})$.
- $B_{\longrightarrow}$ defines the transition model, which describes the dependencies between the variables in the previous slices and the variables in slice t, i.e. the distribution of $X_t | X_1, ..., X_{t-1}$.

We define a dynamic Bayesian network of order n, denoted as n-DBN, to have n+1 time slices. The dependence of X_t on its preceding time slices is only conditional up to the $t-n$ slice. Given $X_{t-n}, ..., X_{t-1}$, X_t is independent of X_{t-n-1}. [15].
$B_{\longrightarrow}$ defines the transition model which describes the distribution $X_t | X_{t-1}, ..., X_{t-n}$ which verifies :

$$\begin{aligned} & P(X_t | X_{t-1}, ..., X_{t-n}) \\ & = P(X_{1,t}, ..., X_{N,t} | X_{1,t-1}, ..., X_{N,t-1}, ..., X_{1,t-n}, ..., X_{N,t-n}) \\ & = \textstyle\prod_{n=1}^{N} P(X_{n,t} | pa_{X_{n,t}}) \end{aligned} \tag{1}$$

where $pa_{X_{n,t}}$ are the indices of the parent variables of the variable $X_{n,t}$ in the graph of $B_{\longrightarrow}$.
For a dynamic Bayesian network with two time periods, the following applies:
B_1 describe the initial probability distribution of the variables by :

$$P(X_1) = \prod_{n=1}^{N} P(X_{n,1} | X_{pa_{n,1}}) \tag{2}$$

Where each node is indexed by a pair (n,t), where n is the proper index and t is the sequential index. The set of index pairs associated with the parents of the node $X_{n,1}$ in the graph of B_1 is denoted by $pa_{n,1}$.

For a dynamic Bayesian network with two time slices, we must write the initial probability distribution of the variables, since $B_{\longrightarrow}$ is a BN that describes the transition law of the process state at time t given its state at time $t-1$. In other words, $B_{\longrightarrow}$ describes the probability law of $X_t | X_{t-1}$, which is factored into $B_{\longrightarrow}$ by the conditional probability law.

$$P(X_t | X_{t-1}) = \prod_{n=1}^{N} P(X_{n,t} | X_{pa_{n,t}}) \tag{3}$$

Where $pa_{n,t}$ represents the indices of the parent nodes of $X_{n,t}$ in the graph of $B_{\longrightarrow}$.
The joint distribution of the sequence of n nodes $(X_t)_{1 \leq t \leq T}$ can be obtained by presenting the Bayesian Network 2D on a sequence of length T. Simply put, the Bayesian Network 2D can be used to calculate the distribution.

$$P((X_t)_{1 \leq t \leq T}) = P(X_1) \prod_{t=2}^{T} P(X_t | X_{t-1}) \tag{4}$$

By introducing the equation 2 and the equation 3 into the equation 4 we obtain equation 5.

$$P((X_t)_{1\leq t\leq T}) = \prod_{n=1}^{N} P(X_{n,1}|X_{pa_{n,1}}) \prod_{t=2}^{T}\prod_{n=1}^{N} P(X_{n,t}|X_{pa_{n,t}}) \tag{5}$$

A dynamic Bayesian network with two time slices is illustrated by the figure 1.

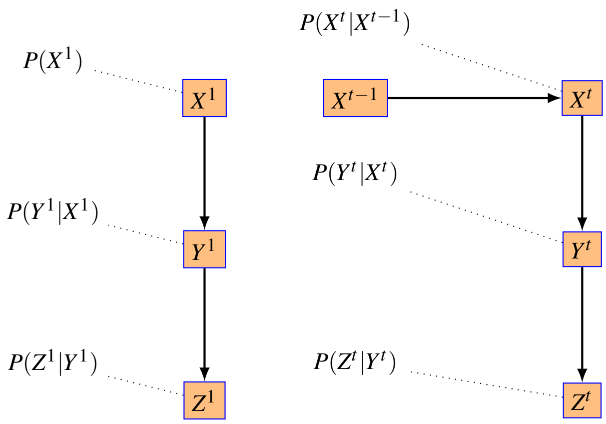

Figure 1: Dynamic Bayesian Network 2-DBN

3. Decision Tree (DT)

Decision Trees (DTs) are a non-parametric method commonly used for classification and prediction[14]. DTs are generally classified into two categories: classification trees and regression trees. Classification trees are used to predict discrete variables, while regression trees are employed to predict continuous variables. The primary advantage of decision trees (DTs) is that variable transformations are unnecessary. Variable transformations will neither change the tree structure nor the resulting outcomes, resulting in time savings. Furthermore, DTs can accurately model complex relationships between variables. Among the classification algorithms, decision trees are the most straightforward to use and interpret, yet they maintain high performance [16]. The modeling of decision trees typically comprises two main stages: stage 1 involves creating the trees, and stage 2 involves pruning them [3]. Pruning is essential for avoiding inappropriate nodes in the tree. A good algorithm produces a larger tree and prunes it based on detecting the ideal pruning threshold. In the case of pre-pruning, the tree grows until a specific criterion is met. In contrast, post-pruning involves building a complete tree, and then replacing the last sub-trees with leaves based on a comparison of the error before and after sub-tree substitution. In the next step, the gain ratio is calculated for all sub-nodes individually, and this process is repeated until all examples within a node belong to the same class. Let T represent the training dataset, consisting of subsets $T_i(i = 1,2,...,n)$. To determine the gain ratio for attribute X, use equation 9.

$$GainRatio = \frac{Gain(X,T)}{SplitInfo(X,T)} \tag{6}$$

where :

$$Gain(X,T) = Entropy(T) - \sum_{i=1}^{n} \frac{|T_i|}{|T|} Entropy \tag{7}$$

$$SplitInfo(X,T) = -\sum_{i=1}^{n} \frac{|T_i|}{|T|} \log_2 \frac{|T_i|}{|T|} \tag{8}$$

By introducing the equation 7 and the equation 8 into the equation 6 we obtain equation 9.

$$GainRatio = \frac{Entropy(T) - \sum_{i=1}^{n} \frac{|T_i|}{|T|} Entropy}{-\sum_{i=1}^{n} \frac{|T_i|}{|T|} \log_2 \frac{|T_i|}{|T|}} \tag{9}$$

In the example presented in figure 2, objective classification of each instance entails initial observation of the state of component X_5. Subsequently, the corresponding branch should be followed based on the status of the X_5 component - whether functioning correctly or otherwise. Therefore, for example, evaluation of component state X_8 requires following the branch X_5 = good, leading to node X_8 = good. A node is a group of instances that meet the prior criteria for separation. The highest node in the tree is X_3 and includes all instances that have a X_5 attribute condition of "good" and a X_8 attribute condition of "good". The highest node in the tree is X_3 and includes all instances that have a X_5 attribute condition of "good" and a X_8 attribute condition of "good". Abbreviations for technical terms will be fully explained upon their first usage. X_2 is considered good when X_1 and X_3 meet their respective conditions. The nodes within the tree provide further segmentation criteria, with the root acting as the entry point for all instances. The nodes within the tree provide further segmentation criteria, with the root acting as the entry point for all instances. Once the tree is constructed, we can extract the various configurations that result in the probabilities found at the root. The nodes within the tree provide further segmentation criteria, with the root acting as the entry point for all instances. This configuration enables the extraction of probabilities that can serve as prior probabilities in the Bayesian network. Examining the example in figure 2, we observe that component 5 failing is the only requirement for the system represented by the root node to fail. This can be observed in the condition $X_5 = failure$.

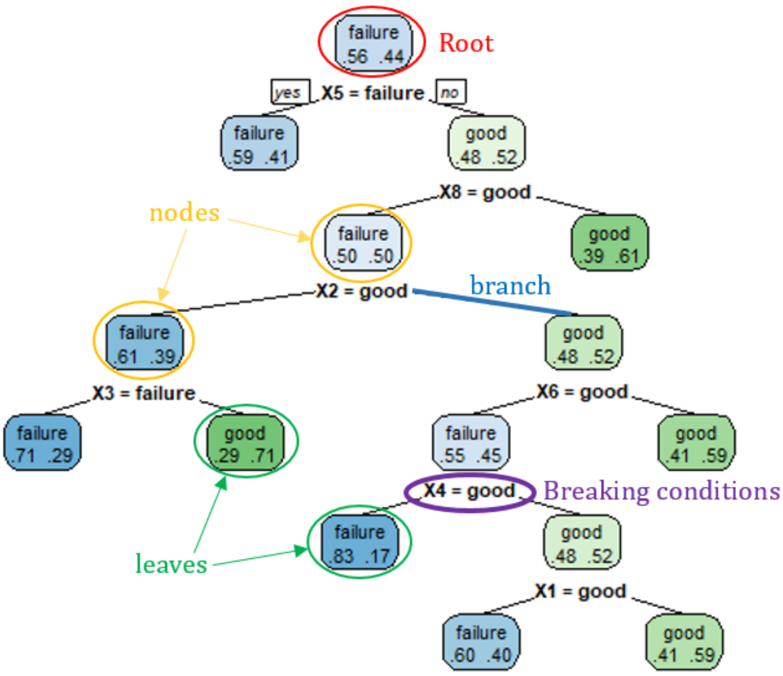

Figure 2: Example of a decision tree

We also note that the values included in each node and leaf in Figure 2 provide the probabilities of success and failure for the corresponding node or leaf.

4. Procedure for Predicting Failures

This section outlines a procedure for constructing a failure prediction model using the operational data of a dynamic system. The method involves utilizing two modeling tools: decision trees and dynamic Bayesian networks. The combination of these tools is justified by the need to establish causal relationships between the system's components when working with BN data. We also need to know the prior probability laws that define the behavior of the components. To facilitate failure prognosis, Dynamic Bayesian Networks (DBNs) are used for representation. The proposed prognosis procedure in this article involves four stages.

- Develop the graphical model using integral causality.
- Generate the probability laws for each component and the system.
- Create the Dynamic Bayesian Network (DBN).
- Conduct future behavior estimation of the system.

4.1. Development of the graphic model

This phase necessitates extensive knowledge of the system under investigation. The professional with a comprehensive comprehension of the system's operation and the causal connections governing component interdependence during operation is best suited to offer assistance at this stage. To account for the subjective nature of human response, it is advisable to conduct a survey with numerous system experts employing a response grid. This will provide a more realistic representation of the dependencies among the system components.

4.2. Generation of a priori probability distributions

For the determination of prior probability distributions, it is recommended by some literature to conduct an expert survey of the system. In order to reduce subjectivity in the prognostic process, we propose the use of a non-human subjective tool, specifically the decision tree. This tool can classify the probability of component failure by considering all other components, an important factor at this stage. This tool can classify the probability of component failure by considering all other components, an important factor at this stage. Technical terminology will be explained when used for the first time. This tool can classify the probability of component failure by considering all other components, an important factor at this stage. We propose the following approach: Utilize a database to display the behavior of the system and its components during a specific operational period. With the assistance of the graphical model's dependencies, categorize each component according to the other components to deduce the likelihood of each component being in a particular state based on the configurations. In our case study, these probabilities will be implemented as prior probabilities for dependent components. For components that operate independently, it is suggested to associate a behavior law with them. In this article, we suggest using the Weibull law because it encompasses a variety of laws and can accurately represent the behavior of degradable components throughout their lifespan. The probability density function for the three-parameter Weibull distribution W(λ,α,θ) is defined by the equation relation 10.

$$f(x,\lambda,\alpha,\theta) = \alpha\lambda(x-\theta)^{\alpha-1}e^{-\lambda(x-\theta)^{\alpha}}1_{[0,+\infty]}(x) \tag{10}$$

Its probability density is expressed by the relation 11.

$$F(x) = \int_t f(x,\lambda,\alpha,\theta) = -e^{-\lambda(x-\theta)^{\alpha}} \tag{11}$$

This density function serves to randomly generate a failure probability to be utilized in Bernoulli's Law for generating the states of non-dependent components. When the component is in optimal working order (denoted by "ok" or "0") at the previous time, the Weibull distribution generates the probability of failure for the current time. On the other hand, if the component failed in the previous period (ko or 1), it is immediately restored to a good condition in the next period. The random probabilities generated by the Weibull distribution are determined by equations 12 and 13.

$$P(X_t = ko|X_{t-1} = ok) = \frac{F(t) - F(t+dt)}{F(t)} \tag{12}$$

$$P(X_t = ko)|X_{t-1} = ok) = \frac{e^{-\lambda(t-\theta)^{\alpha}} - e^{-\lambda(t+dt-\theta)^{\alpha}}}{e^{-\lambda(t-\theta)^{\alpha}}} \tag{13}$$

With α, λ and θ which are the parameters of the weibull distribution.

4.3. Generation of the dynamic Bayesian network

In generating the dynamic Bayesian network, which is utilized to estimate the system's future state, we suggest that only the independent components are considered from a dynamic perspective. The state of these components at time (t) is contingent upon their state in time (t-1) (refer to equation 3). Conversely, the state of other components is dependent on that of their parents at each instant (refer to equation 2). The equation model in 4 provides the overall system model, as shown in figure 1.

4.4. Estimation

The Weibull law, an a priori law utilized in the simulation, extends the exponential law to capture lifetimes and is also applicable in extreme value scenarios. To estimate its parameters, a learning stage is necessary. For parameter learning, we consider the set of probabilities to be estimated as $(\theta_i, ..., \theta_D)$. The estimators, denoted as $(\hat{\theta}_i, ..., \hat{\theta}_D)$, can be established from the information contained in a database listing a specific number of realizations of the process studied. This database is commonly referred to as an example database or a feedback database in the industry. In 2006, Leray proposed various parameter learning methods, depending on whether the data is complete or incomplete. The Maximization Expectation and Maximum Likelihood algorithms can be employed for complete data. For incomplete data, the Maximum a Posteriori and Expectation a Priori algorithms can be applied. Two types of inference are known in the literature: exact inference and approximate inference. Exact inference is utilized for real-time estimation of the system's failure probability [8]. For comprehensive data analysis, we have access to the algorithms of Maximization Expectation, Maximum A Posteriori, Expectation A Priori, and Maximum Likelihood. The literature outlines two types of inference: exact inference and approximate inference. Exact inference is employed to estimate the probability of system failure in real-time. To facilitate more pertinent monitoring and active decision-making, it is essential to utilize a model that accounts for the temporal aspect.

5. Application to a water production system

The water production system is frequently utilized in Cameroonian homes, particularly in Ngaoundéré, to address the issue of limited access to drinking water [1]. To initiate the water production process, the ignition button on the start-up box must be activated, causing the pump to transfer water from the well to the storage reservoir, which can then be distributed for use. After water is pumped through the circuit, it is passed through a non-return valve to prevent backflow. Prior to reaching storage or usage points, the water is filtered to eliminate impurities. Other components in the system include a safety box that stops the pump when there is insufficient well water. This box is triggered by a sensor (probe) located in the well. The manometric kit is used to regulate pressure within the circuit (see figure 3).

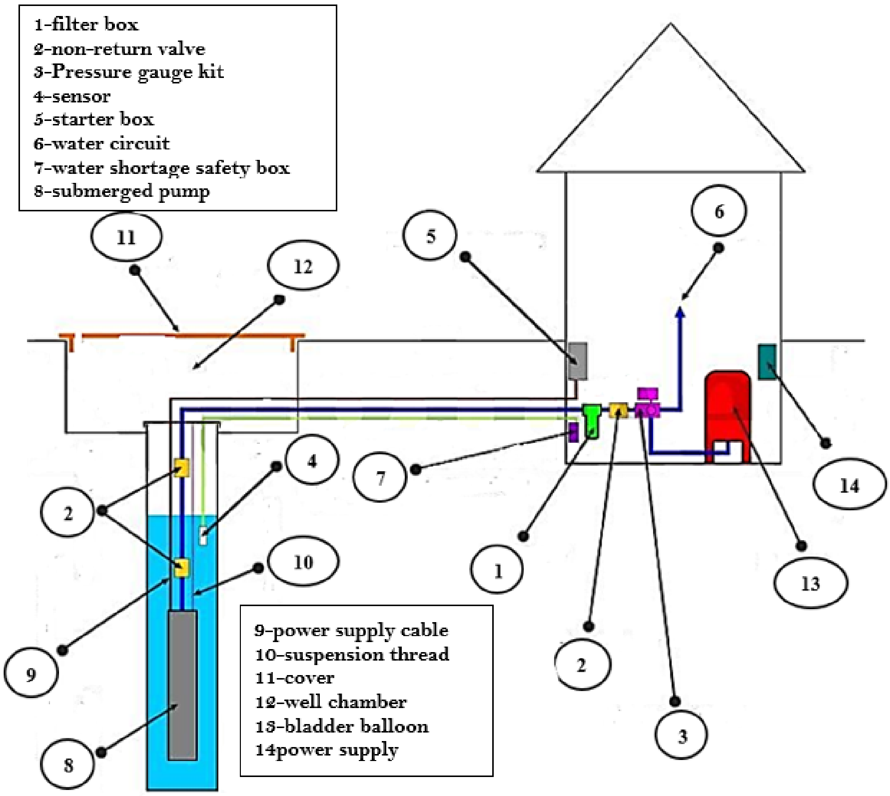

Figure 3: Water production system

This presentation illustrates the functioning of the system, inspired by the depiction of a production system by system experts. Technical abbreviations are explained when introduced. It helps identify the components of our application system, along with their operational dependencies. This is because the failure of the parent components impacts the state of the child components. This is because the failure of the parent components impacts the state of the child components. The components without parents are the ones whose state is not affected by any other component. This results in the graphical causality model presented in figure 4.
The table 1 helps us visualize the correspondences.

Table 1: Concordance table

Variables	Names	components
Compo 1	Filter box	component 1
Compo 2	Pressure gauge kit	component 2
Compo 3	non-return valve	component 3
Compo 4	Water shortage safety box	component 4
Compo 6	Water Circuit	component 6
Compo 7	starter box	component 7
Compo 5	Sensor	component 5
Compo 8	Submersible Pump	component 8

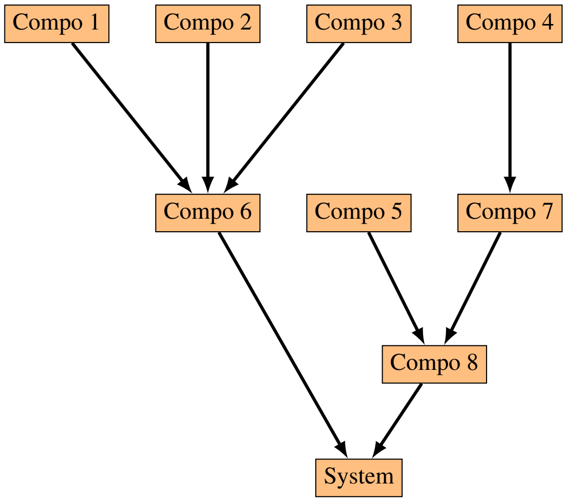

Figure 4: Bayesian network for the system

Once the causality model is constructed, we can proceed to determine the a priori probability distributions for each dependent component. To establish these probability laws, we consider the causal relationships between the components and use decision tree classification based on the data. The decision trees were constructed utilizing the Rpart library in the R software. With regards to the independent components, as previously stated, we assume their behavior during operation adheres to a Weibull distribution. We have assigned the following values to the Weibull parameters (α =0.95, λ=2, and θ= 0) to ensure the components behave similarly to that of the mature phase. The data utilized was obtained from the simulation of a water production system's operation spanning 260 weeks.
The probability distributions resulting from the classification of data using the decision tree tool are presented below.

- For component 6, which relies on components (1, 2, 3), we have derived the classification indicated in figure 5.

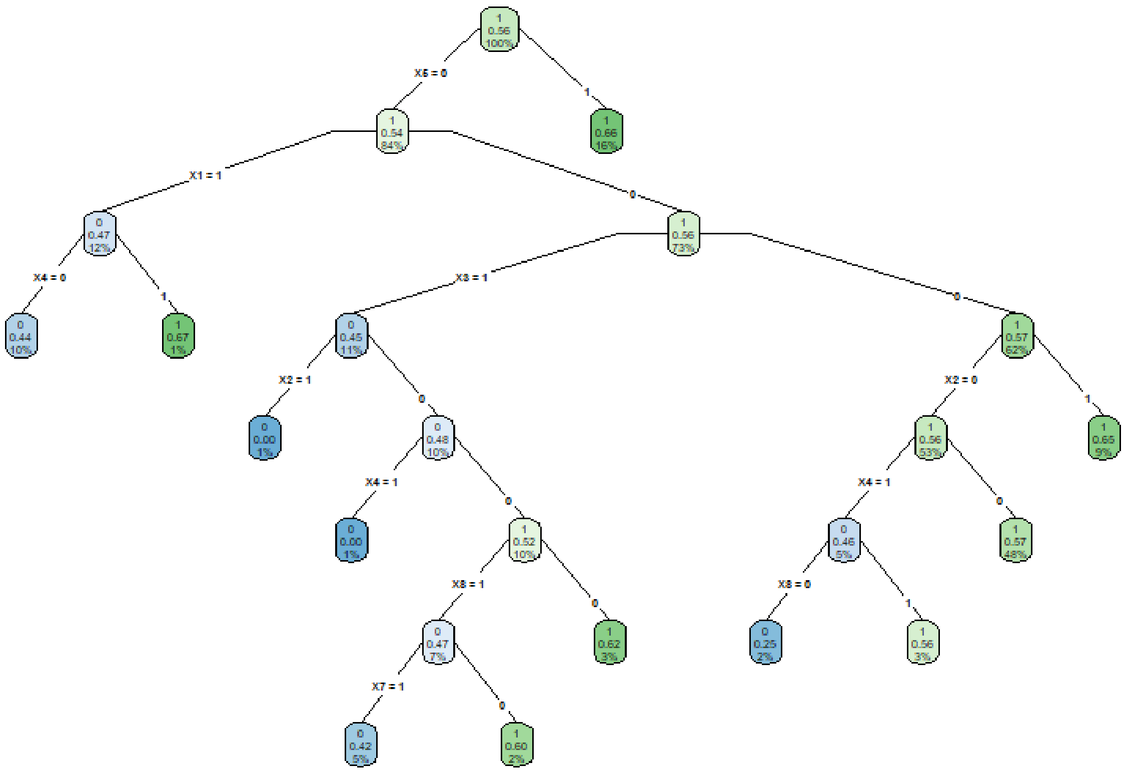

Figure 5: Decision tree for component 6

- For component 7, which is dependent on component 4, the classification displayed in figure 6 is obtained.

- For component 8, which relies on components (5, 7), the classifications shown in figure 7 are obtained.

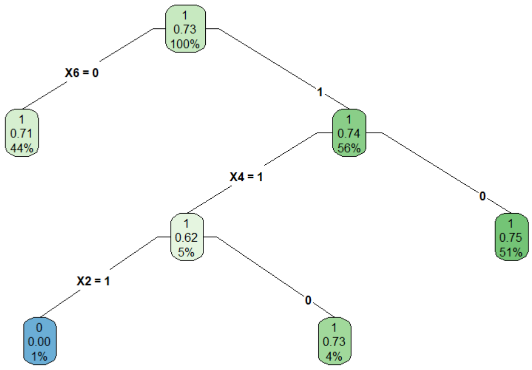

Figure 6: Decision tree for component 7

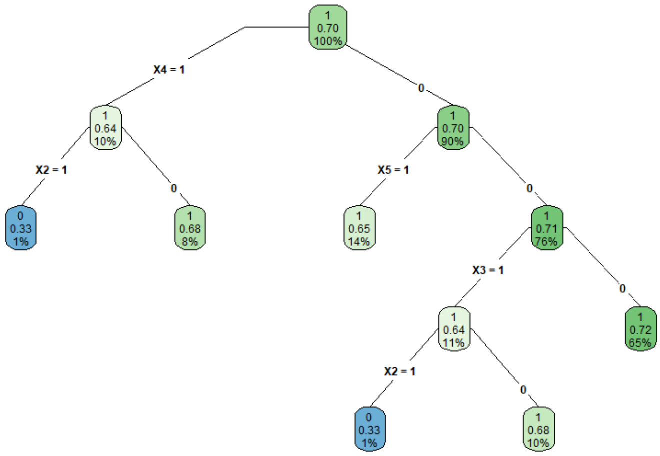

Figure 7: Decision tree for component 8

- For the system that depends on components (6,8), the figures depicted in figure 8 illustrate the classifications.

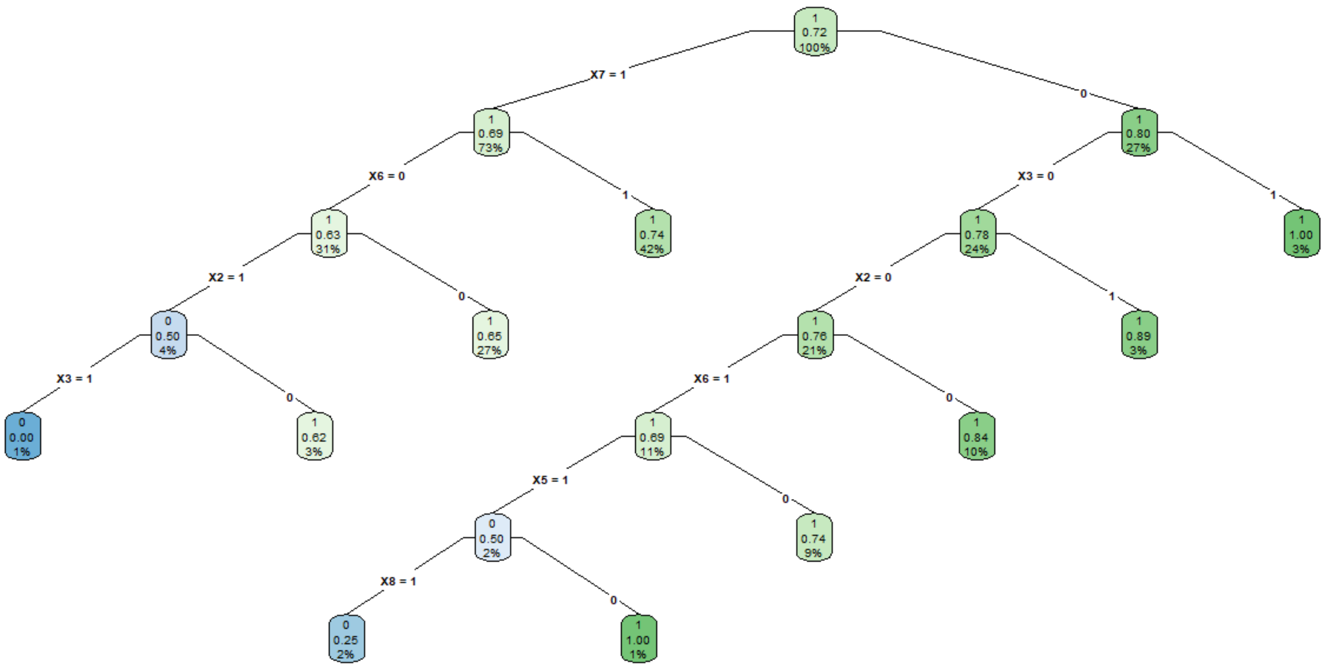

Figure 8: Decision tree for system

From the classification figures 5, 6, 7, 8, four tables 2, 3, 4, 5 emerge, each presenting the probability tables associated with respective nodes. The a priori probabilities are extracted by tracing the path that connects the different components, identified as parent nodes in the graphical model depicted in Figure RBSF. The value 0 denotes the operational status without any failure, while 1 signifies the operational status with a failure. To better understand the priori determination of probabilities, we can consider the example of component 6. The decision tree for component 6 is presented in figure 5. By observing figure 4, we can infer that the status of component 6 relies on that of components 1, 2, and 3. For this reason, to determine the a priori probability of component 6 being in a particular state, given that components 1,2,3 are in a certain state, we only need to follow the relevant paths on the decision tree that include components 1,2,3. This enables us to ascertain that: $P(X_6 = 0|X_1 = 0, X_2 = 1, X_3 = 1) = 0$. By adhering to the conditions $X_6 = 0, X_1 = 0, X_3 = 1, X_2 = 1$ we can reach the sheet which presents us with the desired outcome. To calculate the probabilities of paths that are not directly visible on the tree, the Bayes law is employed.

Table 2: Probability table for component 6

C1	C2	C3	0	1
0	0	0	0.44	0.56
0	0	1	0.48	0.52
0	1	0	0.35	0.65
0	1	1	0.00	1.00
1	0	0	0.47	0.53
1	0	1	0.21	0.79
1	1	0	0.16	0.84
1	1	1	0.00	1.00

Table 3: Probability table for component 7

C4	0	1
0	0.25	0.75
1	0.38	0.62

Table 4: Probability table for component 8

C6	C7	0	1
0	0	0.30	0.70
0	1	0.30	0.70
1	0	0.30	0.70
1	1	0.30	0.70

Table 5: Probability table for system

C6	C8	0	1
0	0	0.28	0.72
0	1	0.28	0.72
1	0	0.29	0.69
1	1	0.25	0.75

For the construction of the DBN model, temporal dependencies only exist among the incoming nodes of the static Bayesian network. This relationship is explained in sub-section 3.3, and is depicted in figure 9.

Knowing the graphical model, the next step is to estimate the probability of system failure at time (t+1), based on its state at time (t). To estimate the probability of failure of components that are part of the nodes without parents (network entry nodes) at time (t), we must find the optimal parameters $(\hat{\alpha}, \hat{\beta}, \hat{\lambda})$ of equation 13 using the Nelder-Mead method. This equation effectively presents the probability of a component failing at time (t), given it was in good condition at time (t-1). After determining optimal parameters using the available data at time (t), they are utilized in the same equation to evaluate the likelihood of component failures at time (t+1). As the system and its components are in binary operation, the binomial law is utilized to determine the state of each component and the system every time its probability of failure is estimated. It is important to note that technical term abbreviations will be explained when used for the first time. To determine the probability of failure of the children and their states, it becomes necessary to test the states of the components representing the nodes between using the associated probability tables for each node. By following this procedure, we can derive the probability of system failure at every point during the 50-year (260-week) functioning period, as illustrated in figure 10.

The figure 10 illustrates two curves, the black one representing the probability of system failure over time and the red curve displaying the estimated probability. It is noteworthy that there is a significant difference between the estimated probability and the actual probability in the first few weeks, but as time passes, the two curves converge. It can be inferred that as the data increases over time, the estimate becomes more reliable. The goal is to confirm this statement via a compliance examination. To verify our estimate, we will conduct a confidence interval test. This test will determine if the estimation error is genuinely zero or approaching zero. If so, we can confidently assert that the

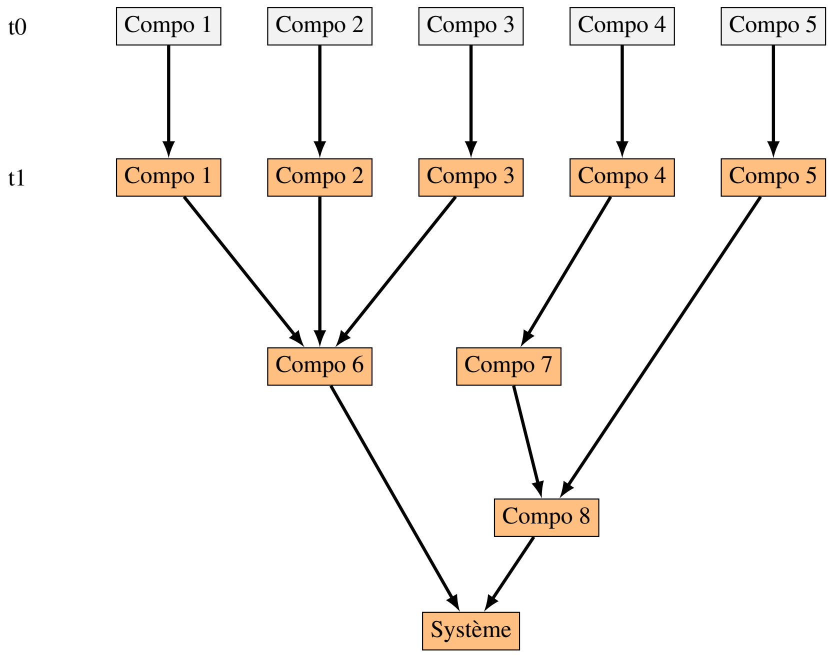

Figure 9: Dynamic Bayesian Network for the system

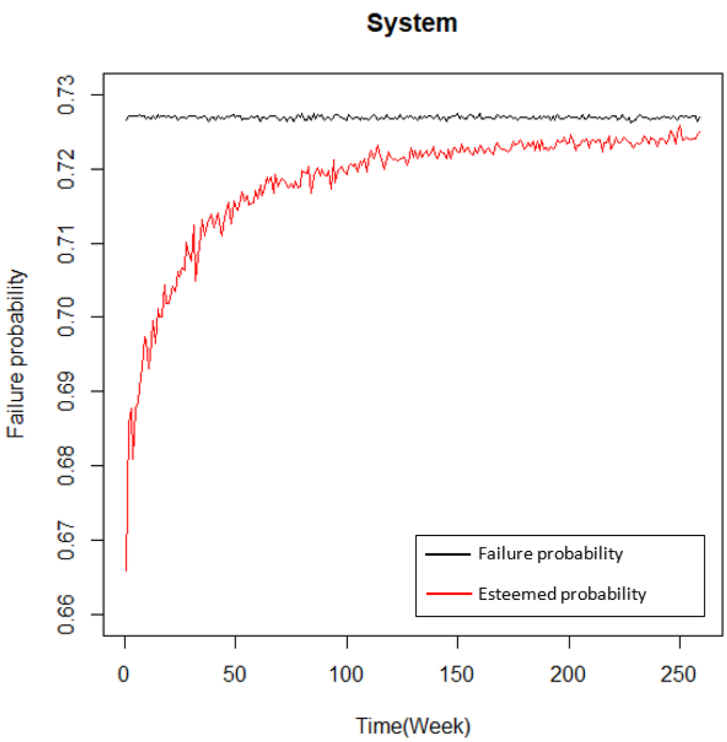

Figure 10: Probability of failure

probability of failure estimation is accurate with additional data. In the event that the estimation error is zero, the probability of failure estimation is identical to the true probability. To demonstrate that the estimation error is zero, it must be within the confidence interval of the estimation error, including the zero point.

The graph in figure 11 displays the time estimation error as the black curve, with an ideal average of estimation errors being equal to zero. The red and black curves show confidence intervals. This graph enables verification of any statistical difference between the system's probability and the estimated probability. In order to affirm equality to zero, it is imperative for the confidence interval of the estimation error to encompass zero. On the graph, the curve of zeros should be situated between the blue and red curves. Upon examination of the black curve, it is evident that almost none of the estimation errors match zero within the selected time interval. With time, we can observe a convergence towards the zero line. To conclusively determine if there are estimation errors that differ from zero, we constructed a two-sided 95 percent confidence interval for each estimation. After constructing the confidence interval, we conducted a two-sided test with the null

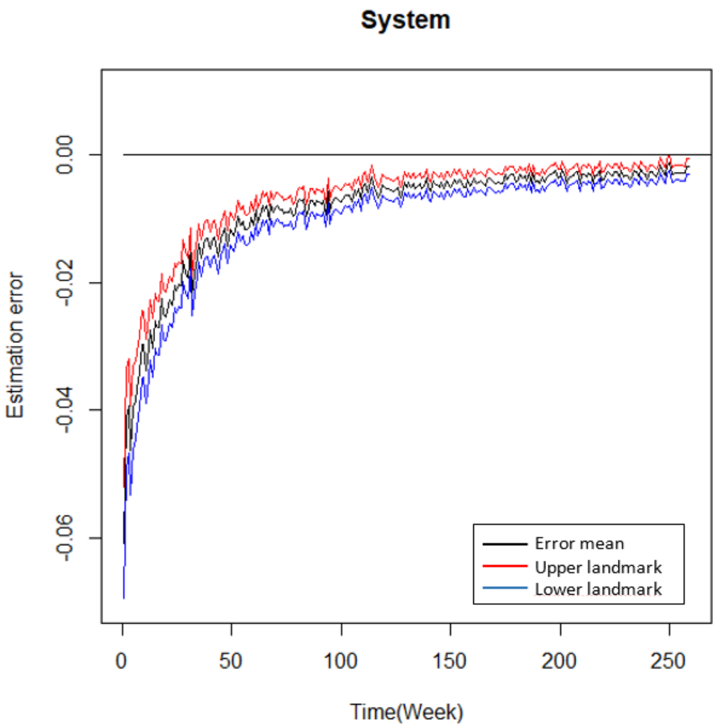

Figure 11: Error curve

hypothesis that the estimation error equals zero. As Figure 1 depicts, the null hypothesis is rejected for all lengths in a period of 5 years (260 weeks). Nevertheless, there is a swift approach towards the null value. In addition, we noticed that the initial weeks' estimation error is in the order of hundreds, which is acceptable.

6. Discussion

Dynamic Bayesian Networks (DBNs) and Decision Trees (DTs) are two popular machine learning techniques used in a variety of fields. DBNs are a type of probabilistic graphical model that captures causal relationships between variables in time-dependent systems. DTs, on the other hand, are a supervised learning method that classifies data based on a set of decision rules. While DBNs are highly effective for modeling complex systems with dynamic behavior, they require large amounts of data and can be resource-hungry. DTs, on the other hand, are computationally efficient and require relatively little data for learning, but they may not be as accurate as DBNs for complex problems. This study focuses on failure prediction for a bi-state system with application to a water production system. Pairing these two methods has made it possible to set up a prediction process that can be applied to complex systems with little data. The joint use of these two methods limits human subjectivity in the prediction process, in that the state of determination of the conditional probability tables is no longer the responsibility of the experts [10], but is determined using data and DTs. In the case of the water production system in our study, we realize that prediction becomes more accurate over time as the system evolves.

7. Conclusion

The aim of this article is to suggest a method for anticipating breakdowns of multi-component single-state production systems. We commence by introducing the tools used in our research. We then explain the prediction approach proposed to gauge the dependability of production systems and apply it to a water production system. Therefore, it appears that combining decision trees and dynamic Bayesian networks minimizes human subjectivity in estimating the reliability of production systems. The amount of data significantly affects failure prognosis as it works in opposition during the learning phase. This paper aims to gather more data to enhance machine learning for precise estimation and intends to conduct comparable work for multi-component, multi-state systems. As complete data is often difficult to obtain in reality, it would be valuable to consider the inclusion of missing data in failure prediction research.

Acknowledgement

The authors express appreciation to the HOLY SPIRIT for inspiring and supporting their work. They also thank the anonymous reviewers for their time and effort, which helped clarify the research contributions in the main document and improve its quality. No subjective evaluations

were included and a formal register was maintained throughout the text.

References

[1] N. N. Benjamim, L. Sara, and E. Emmanuel. problèmatique de l'acces à l'eau potable dans la ville de ngaoundéré (centre nord-cameroun). *Editorial Advisory Board e*, 18(2):223–230, 2005.

[2] T. Boukra. Diagnostic et pronostic des défauts des moteurs asynchrones.

[3] D. T. Bui, B. Pradhan, O. Lofman, I. Revhaug, and O. B. Dick. Landslide susceptibility assessment in the hoa binh province of vietnam: a comparison of the levenberg–marquardt and bayesian regularized neural networks. *Geomorphology*, 171:12–29, 2012.

[4] Y. Debbah. *Développement d'un outil de pronostic pour la maintenance des systèmes mécaniques.* PhD thesis, 2018.

[5] N. Ghanmi, M. A. Mahjoub, J. Khlifia, and N. E. B. Amara. Proposition d'un modèle de réseau bayésien dynamique appliqué à la reconnaissance de mots arabes manuscrits. 2012.

[6] R. Gouriveau, M. El Koujok, and N. Zerhouni. Spécification d'un système neuro-flou de prédiction de défaillances à moyen terme. 2007.

[7] M. Kevin, Patrick. Dynamic bayesian networks: representation, inference and learning. 2002.

[8] P. Leray. Réseaux bayésiens: apprentissage et modélisation de systèmes complexes. *habilitation à diriger les recherches, Université de Rouen*, 2006.

[9] J. Li, j.S. Rebecca, G. Wang, X. Liu, Z. Li, and M. Xu. *Hard drive failure prediction using Decision Trees", Reliability Engineering and System Safety,*. 2017.

[10] Z. Li, T. Xu, J. Gu, Q. Dong, and L. Fu. Reliability modelling and analysis of a multi-state element based on a dynamic bayesian network. *Royal Society open science*, 5(4):171438, 2018.

[11] V. L. J. K. Miroslav Kvassay, Elena Zaitseva. Binary decision diagrams in reliability analysis of standard system structures. 2016.

[12] M. Nahim, Hassan. *Contribution à la modélisation et à la prédiction de défaillances sur les moteurs Diesel marins*. PhD thesis, 2016.

[13] P. Naïm, P.-H. Wuillemin, P. Leray, O. Pourret, and A. Becker. *Réseaux bayésiens*. Editions Eyrolles, 2011.

[14] S. E. G. C. B. A. D. T. Nefeslioglu, H.A. Assessment of landslide susceptibility by decision trees in the metropolitan area of istanbul, turkey. In *Math. Probl. Eng.*, page 1–15. https://doi.org/10.1155/2010/901095., 2010.

[15] P. Weber and L. Jouffe. Reliability modelling with dynamic bayesian networks. 2003.

[16] I. Witten and E. Frank. Data mining practical machine learning tools and techniques. morgan kaufmann. 2005.