



**HAL**  
open science

## A contribution to the evaluation of imprecise availability of complex systems using markov models

Joanna Akrouche, Mohamed Sallak, Eric Chatelet, Fahed Abdallah, Hiba Haj  
Chehade

### ► To cite this version:

Joanna Akrouche, Mohamed Sallak, Eric Chatelet, Fahed Abdallah, Hiba Haj Chehade. A contribution to the evaluation of imprecise availability of complex systems using markov models. 2nd International Conference on Uncertainty Quantification in Computational Sciences and Engineering (UNECOMP 2017), Jun 2017, Rhodes Island, Greece. pp.456-466, 10.7712/120217.5383.16866 . hal-02004699

**HAL Id: hal-02004699**

**<https://hal.science/hal-02004699v1>**

Submitted on 17 Feb 2020

**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.

## **A CONTRIBUTION TO THE EVALUATION OF IMPRECISE AVAILABILITY OF COMPLEX SYSTEMS USING MARKOV MODELS**

**Joanna M. Akrouche<sup>1</sup>, Mohamed Sallak<sup>1</sup>, Eric Châtelet<sup>2</sup>, Fahed A. Abdallah<sup>3</sup>, and Hiba  
Z. Haj Chhadé<sup>3</sup>**

<sup>1</sup> Sorbonne universités, Université de technologie de Compiègne, CNRS,

Heudiasyc UMR 7253, CS 60 319, 60 203 Compiègne cedex  
e-mail: {joanna.akrouche, mohamed.sallak}@hds.utc.fr

<sup>2</sup> Université de technologie de Troyes, CNRS, Institut Charles Delaunay/LM2S  
12 rue Marie Curie, CS 42060, 10004 Troyes cedex, France  
eric.chatelet@utt.fr

<sup>3</sup> Université Libanaise  
Beirut, Hadath  
{fahed.abdallah76,hiba.hajchhade}@gmail.com

**Keywords:** Dependability, imprecise availability, imprecise Markov chain, interval probabilities, contraction techniques.

**Abstract.** *When using classical methods for the availability assessment of a multi-state system, the precise values of state' probabilities are required. But, in many cases the available data does not describe the system's components, defining the system's state, precisely. To cope with this problem, the imprecision can be incorporated into the method in terms of imprecise rates [1] (failure and repair rates) by using imprecise probability theory [2]. Markov chain models are known for their simplicity and their great ability to model repairable systems, thus, they are well adapted for modeling stochastic failure and repair processes, where conditional probability distribution of future states depends only on the present state, and then computing the system's availability. To our best of knowledge, only a few works were developed in the context of imprecise continuous Markov chain [3]. The idea in this paper is to replace precise initial distributions and transition matrices by imprecise ones where imprecise rates are expressed in terms of intervals which are supposed to contain the true unknown initial probability and transition matrix. The contribution of this work is twofold: first, applying interval analysis techniques on existing algorithms for availability assessment of multi-state systems, and second, studying the stationarity, convergence and ergodicity properties related to the new proposed technique.*

## 1 INTRODUCTION

The exponential evolution of technology has increased the complexity of systems and has further reduced their design and manufacturing costs. Correspondingly, manufacturers rely on the criterion of quality to be distinguished in the market. To achieve this goal, they must master various tools that will enable them to keep a competitive position and take actions for improvement at all levels. All these reasons make operating reliability the undeniable means that must be mastered when designing any system. Dependability is defined as the property that enables system users to place a justified confidence in the service it delivers to them and it groups two key concepts: reliability and availability. The reliability  $R(t)$  is the ability of a system to remain constantly operational in a given duration, in other words, the probability that an item survives the time interval  $[0, t]$  and still functions at time  $t$ . The availability  $A(t)$  is the ability of a system to be operational at a specific moment [5].

The principle is to choose amongst several components with different performance, the best components of the system, and also find the optimal configuration, that is the connection between the different components entities that constitute the system. And this must be done while respecting that a certain level of availability is ensured by the complete system.

Dependability domain is extremely large and complex. In this paper we are particularly interested in the optimization of the availability of a product taking into account the multi-state case which is commonly encountered in practical life. We also consider that systems and their components can operate in different performance levels between working and failure states. Availability analysis helps to calculate the ability of a system to provide a required level of performance depending on the level of degradation.

Several methods are employed to calculate the availability, amongst them, there are: Universal Generating Function method (UGF) [6], Inclusion-Exclusion technique [7], and Markov Chain approaches [3], etc. There are many probabilistic techniques that can evaluate this criterion, but these techniques are only effective for very specific cases, for example the case of binary systems. The transition to multi-state [7,8] systems drastically restricts the application of most of these methods. Indeed, the evaluation of the reliability of a multi-state system [9] is more difficult than in the binary case because we have to take into account the different combinations of the component failure modes. In addition to the multi-state aspect, there is the structural aspect of the system which illustrates the type of connection connecting the components and of course the existence of uncertainties. In general, most methods can be applied efficiently only to systems with a simple structure: series, parallel, bridge or mixed. Nevertheless, there is virtually no practical and effective method that relates to systems with complex structures.

In this work, we demonstrate how Markov chains can be used to study the availability of a system by treating uncertainties like intervals and using interval analysis techniques.

The structure of the paper is as follows. Section 2 explains the Markovian approach with imprecise data for multi-state systems with several components. Section 3 introduces contractors and how we use them to obtain the availability. Section 4 presents some applications and the last section concludes the paper.

## 2 MARKOVIAN APPROACH

Traditional binary-state reliability models allow only two possible states for a system and its components: perfect functioning (up) and complete failure (down). However, a system can have a finite number of performance rates. Also, many real-world systems are composed of various components which in turn can have different performance levels and for which one cannot formulate an "all or nothing" type of failure criterion. Failures of some system elements lead, in these cases, only to performance degradation. Such systems are called

multi-state systems (MSS). Traditional reliability theory, which is based on a binary approach, has recently been extended by allowing components and systems to have an arbitrary finite number of states. Many methods have been proposed to study the availability of a MSS, in our approach we use the Markovian approach.

## 2.1 General description of the Markov Model

A stochastic process describes the evolution of a system over time using a set of probabilities of the system's states (or a subset of states) at time instants  $t \{X(t), t \geq 0\}$  [10]; where  $X(t)$  is a random variable that denotes the state of the process at time  $t$ . A Markov model is a class of stochastic processes where the future state depends only on the present state. When time is discrete, we have a *discrete-time Markov chain*; when time is continuous, we have a *continuous-time Markov chain* which is also called a Markov process. Formally, a discrete-time Markov chain is characterized by a (discrete) set of states  $S$  and the transition probabilities  $p_{ij}$ , where  $p_{ij}$  is the probability that the Markov chain moves at the next time instant to state  $j$ , given that it is at the present time point at state  $i$ . The matrix  $P$  grouping elements  $p_{ij}$  is called the transition probability matrix of the Markov chain. Note that the definition of the  $p_{ij}$  implies that the row sums of  $P$  are equal to 1. A continuous-time Markov process is also described by a discrete set of states  $S$ , in this case, however, transition probabilities  $p_{ij}$  are replaced by transition rates  $q_{ij}$  which will be grouped into a transition rate matrix denoted hereafter as  $Q$ . The matrix  $Q$  is an array of numbers describing the rate a continuous time Markov chain moves between states, it is expressed in terms of failure rates  $\lambda$  and repair rates  $\mu$ . Diagonal elements  $q_{ii}$  are defined such that:  $q_{ii} = -\sum_{i \neq j} q_{ij}$ , thus the row sums of  $Q$  are equal to 0. A Markov process is a stochastic process in which the future state does not depend on the past trajectory. Markov models are frequently used in RMS work where events, such as the failure or repair of a module, can occur at any point in time. The Markov model evaluates the probability of jumping from one known state into the next logical state (e.g., from "everything is working" state to "first item failure" state, then from "first item failure" state to "second item's failure", and so on,) until the system reaches the final or totally failed state (which depends upon the configuration of the system being considered). The basic assumption of a Markov process is that the behavior of a system in each state is memoryless. A memoryless system is characterized by the fact that the future state of the system depends only on its present state.

The state equations of a system  $S$  are defined in a discrete state space  $E$  by taking into account the transition rates  $q_{ij}(t)$  existing between consecutive system states. By considering the probabilities that the system stays in its current state or moves to any of the possible states in  $E$  for an elementary time interval  $[t, t + dt]$ , we obtain a system of differential equations, called equations of Chapman-Kolmogorov. Thus, for each state  $e_i$  [10]

$$P_i(t + dt) = P(S \text{ in state } e_i \text{ at } t \text{ and in } [t, t + dt]) + \sum_{e_j \in E - e_i} P(S \text{ in state } e_j \text{ at } t \text{ and in } e_i \text{ at } [t, t + dt]) \quad (1)$$

The behavior of a Markov chain is fully probabilistically described if the initial and transition probabilities are given, that is, if the following probabilities are known:

$$P(X_0 = x_i) = p_i \quad \text{and} \quad P(X_{n+1} = x_j | X_n = x_i) = q_{ij} \quad (2)$$

From (1) we obtain a system of Chapman-Kolmogorov equations:

$$\dot{P}(t) = P(t) \cdot Q \quad (3)$$

with  $P$  is a vector representing the probabilities of being in a state at a certain time  $t$  with

$$\sum_{j=1}^n P_j(t) = 1 \quad (4)$$

$P$  is the solution of the Chapman Kolmogorov and  $Q$  is the transition matrix between the states of the system. The solution of (2) is expressed in exponential form as:

$$P(t) = \exp(Qt) \cdot P(0) \quad (5)$$

Where  $\exp(Qt)$  is an  $n \times n$  matrix and  $P(0)$  is the initial probability vector describing the initial state of the system.

## 2.2 Stationarity

For each system of components, the availability of being in some state, either working or not, will change at the end [3]. So in our study, we focus on finding the availability after an infinite time which is equivalent to calculating the stationarity of the system. The Chapman-Kolmogorov equations at time  $t \rightarrow \infty$  lead to:

$$\Pi \cdot Q = 0 \quad (6)$$

where  $\Pi$  is the stationarity vector representing the probabilities that the system will be in a certain state:

$$\Pi = [ \pi_1 \ \pi_2 \ \dots \ \pi_i \ \dots \ \pi_n ] \quad (7)$$

An element  $\pi_i$  of  $\Pi$  is hence the probability that the system is at state  $e_i \in E$ , and  $\Pi$  verifies:

$$\sum_{i=1}^n \pi_i = 1 \quad (8)$$

## 2.3 Imprecise Markov Chain

The Markov assumption stating that  $X_{t+dt}$  is conditionally independent of  $X_s$ , for  $s < t$ , knowing  $X_t$  may not be realistic, especially for repair, also the transition rates may not be constant in time, but are usually affected by a variety of factors, and the estimation of the rates themselves may be difficult due to the lack of data. Particularly, under constant transition rates, repair times are exponentially distributed and are independent of the history of the system, but repairs will often follow a binomial distribution rather than an exponential distribution [10]; the same applies for failure rate. A full modeling of these details requires a lot of data and expert knowledge. Instead of ignoring this problem, a better way to cope with it is to incorporate the imprecision into the models. This becomes possible with the development of models of imprecise probabilities, such as the interval probability model, it seems therefore convenient to consider our transition rates as not being fixed, but instead being bounded by an interval.

Imprecision may exist on the initial probabilities or the transition matrix, and sometimes even on both. To model this imprecision, probabilities in (2) will be replaced by intervals. Thus we have the following imprecise Markov model [11]:

$$P(X_0 = x_i) \in [ \underline{p}_i, \overline{p}_i ] \text{ and } P(X_{n+1} = x_j | X_n = x_i) \in [ \underline{q}_{ij}, \overline{q}_{ij} ] \quad (9)$$

According to the above model, any probability vector that satisfies  $p_i \in [ \underline{p}_i, \overline{p}_i ]$  for each state with  $i = 1, \dots, n$ , can be considered as an initial distribution, and similarly any transition matrix  $Q \in [ \underline{Q}, \overline{Q} ]$  can be the transition matrix at time  $t$ .

As stated before, the purpose is to find the availability at  $t \rightarrow \infty$ . Under the assumptions in (9), the stationarity of the system is hence determined in form of a vector of intervals  $\Pi = \{ \pi_i \}$ ,  $i = 1, \dots, n$ , where  $\pi_i \in [ \underline{\pi}_i, \overline{\pi}_i ]$  is the probability of being in a state  $e_i$ . Finding the interval

of stationarity for each state is not as simple as it might seem. In the case of precise data, we have a constant transition matrix so we can find the answer by solving a system of equations, but in the case of imprecise data, the bounds of the intervals of stationarity cannot be obtained just by taking into account the two bounds of the transition matrix as some previous work suggests [3].

Several methods have been proposed to solve this problem by finding the solution of equation (6) in the presence of imprecise data. The exact method is a technique where we find all the possible transition matrices of a system and then for each matrix we solve (6) and we find a vector of stationarity, to form at the end a vector of intervals which contains all of the obtained vectors, this method gives an exact result but its complexity increases drastically with the system's size since it computes all the possible state values. BUGF (Believe Universal Generated Function) [12] and IUGF (Interval Universal Generated Function) [11] are two methods based on the UGF (Universal Generated Function) but they are applied on intervals and therefore are used in the case of interval-modeled imprecision, these two methods are efficient and give good results, the IUGF is also noted as more efficient than the BUGF in [12]. In our approach we propose to determine the availability of the system by using a new technique applied on intervals, that is the technique of contractors [4] which we introduce in section 3.

### 3 THE TECHNIQUE OF CONTRACTORS

#### 3.1 Definition of contractors

Consider  $n_x$  variables  $x_i \in \mathbb{R}$ ,  $i = 1, \dots, n_x$ , linked by  $n_f$  relations (or constraints) [4] of the form

$$(10) \quad \mathbf{f}_j(x_1, \dots, x_{n_x}) = 0 \quad , \quad j \in 1, \dots, n_f$$

Each variable  $x_i$  is known to belong to a domain  $X_i$ . For simplicity, these domains will be intervals, denoted by  $[x_i]$ . Define the vector  $\mathbf{x}$  as:

$$(11) \quad \mathbf{x} = (x_1, \dots, x_{n_x})^T$$

and the prior domain for  $\mathbf{x}$  is a box as:

$$(12) \quad [\mathbf{x}] = [x_1] \times \dots \times [x_{n_x}]$$

Let  $\mathbf{f}$  be the function whose coordinate functions are the  $\mathbf{f}_j$ 's. Equation (10) can be written in vector notation as:

$$(13) \quad \mathbf{f}(\mathbf{x}) = 0$$

This corresponds to a *constraint satisfaction problem* (CSP)  $H$ , which can be formulated as:

$$(14) \quad H: (\mathbf{f}(\mathbf{x}) = 0, \mathbf{x} \in [\mathbf{x}])$$

The solution set of  $H$  is defined as:

$$(15) \quad S = \{\mathbf{x} \in [\mathbf{x}] | \mathbf{f}(\mathbf{x}) = 0\}$$

Contracting  $H$  means replacing  $[\mathbf{x}]$  by a smaller domain  $[\mathbf{x}']$  such that the solution set remains unchanged, i.e.  $S \subset [\mathbf{x}'] \subset [\mathbf{x}]$ . There exists an optimal contraction of  $H$ , which corresponds to replacing  $[\mathbf{x}]$  by the smallest box that contains  $S$ . A contractor for  $H$  is any operator that can be used to contract it.

A contractor  $C$  is defined as an operator used to contract the initial domain of the CSP, and thus to provide a new box [11].

$$C([\mathbf{x}]) \subset [\mathbf{x}] \text{ (Contractance property)} \quad (16)$$

$$C([\mathbf{x}]) \cap X = [\mathbf{x}] \cap X \text{ (completeness property)} \quad (17)$$

These relations mean that contractions gives a sub-domain of the input domain  $[\mathbf{x}]$ , and the resulting subdomain  $C([\mathbf{x}])$  contains all the feasible points with respect to the constraints. No solution is “lost”.

### 3.2 Types of contractors

Several contractors exist, each works in a different manner and is efficient only for specific CSPs and for certain cases [13,14,15]. “Intervalization with Gauss elimination” [4] is an important class of CSPs for which intervalization of finite subsolvers can be employed only if the system is formed of linear interval equations and if all elements on the diagonal of the matrix are different than zero. “Gauss-Seidel” contractor is an efficient way to contract intervals of a linear system but the matrix in this case must be reversible [4]. “Krawczyk and Newton contractors” are two contractors which are not always applicable and are much complicated than the other contractors [4]. Finally, one popular contraction technique, which will be used in our approach, is the “Forward-backward propagation (FBP) contractor”. This technique is known for its simplicity and ease, it is also more general than the others since it works on all type of systems [4]. It also gives guaranteed results which means that during the contraction we always get an interval belonging to the initial interval. As compared to the “Gauss-Seidel” technique, the FBP offers comparable accuracy. The “Gauss-Seidel” approach is however less general and becomes time-consuming when the system's size increases. For all these reasons we chose to use the “Forward-backward” propagation technique to determine the stationarity as given in equation (6) for the case of imprecise data.

### 3.3 Forward-backward propagation technique

Forward-backward (FBP) contractor  $C_{\downarrow\uparrow}$  (also known as HC4Revise [16]) is a classical algorithm in constraint programming for contracting. This contractor makes it possible to contract the domains of the CSP  $H$  by taking into account each one of the  $n_f$  constraints apart. The algorithm works in two steps [4]. The forward step applies interval arithmetic to each operator of the function  $\mathbf{y}=\mathbf{f}(\mathbf{x})$ , from the variable's domain ( $[\mathbf{x}]$ ) up to the function's domain ( $[\mathbf{y}]$ ), this step considers the direct forms of the equations. The backward step sets the interval associated to the new function's domain  $[\mathbf{y}]$  to  $[0, 0]$  (imposes constraint satisfaction, since we are solving  $\mathbf{f}(\mathbf{x})=0$ ) and, then, applies backward arithmetic from the function's domain to the variable's domain, which means using the inverse of the functions that appear in the equations  $\mathbf{f}(\mathbf{x})$ . The following example explains the procedure of the FBP technique.

**Example:** Consider the constraint  $y = -5x_1 + 2x_2 = 0$  and the initial box-domain  $[\mathbf{x}] = [1, 4] \times [-3, 7]$ . This constraint can be decomposed as shown in (18) into three primitive constraints (i.e. constraints associated with a unique elementary function: multiplication or addition) by introducing two intermediate variables  $a_1$  and  $a_2$  defined as:  $a_1 = -5x_1$  and  $a_2 = 2x_2$ .

Initial domains for these variables are determined as follows:

$$\begin{aligned} a_1 &= -5x_1 = -5 \times [1,4] = [-20, -5] \\ a_2 &= 2x_2 = 2 \times [-3,7] = [-6,14] \\ y &= a_1 + a_2 = [-20,5] + [-6,14] = [-26,9] \end{aligned} \quad (18)$$

and this step is called the "forward propagation". A method for contracting  $H$  with respect to the constraint  $f(\mathbf{x}) = -5x_1 + 2x_2 = 0$  is to contract each of the primitive constraints in (18) until the contractors become inefficient. For this example:

Since  $f(\mathbf{x}) = 0$ , the domain for  $y$  should be taken equal to  $\{0\}$ , so we can add the step:

$$[y] := [y] \cap \{0\} \quad (19)$$

If  $[y]$  as computed in (18) turns out to be empty, then the CSP has no solution. Else,  $[y]$  is replaced by 0, which is the case in this example. After, a backward propagation is performed, updating the domains associated with all the variables to get:

$$\begin{aligned} [a_1] &:= ([y] - [a_2]) \cap [a_1] = [-14, -5] \\ [a_2] &:= ([y] - [a_1]) \cap [a_2] = [5, 14] \\ [x_1] &:= ([a_1]) / -5 \cap [x_1] = [1, 14/5] \\ [x_2] &:= ([a_2]) / 2 \cap [x_2] = [5/2, 7] \end{aligned} \quad (20)$$

Thus, we obtain the new box:

$$[\mathbf{x}](1) = [1, 14/5] \times [5/2, 7] \quad (21)$$

which is the result of the first FBP contraction. Iterating this procedure, the resulting sequence of boxes  $[\mathbf{x}](k)$  converges towards the smallest possible domain, after which the domains no longer change following another iteration of FBP.

#### 4 APPLICATION

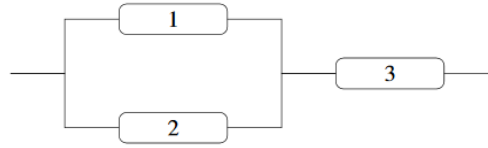


Figure 1: Flow transmission system

To illustrate the technique of contraction with the Forward-Backward propagation, we use the example presented in [11]. In this example, we evaluate the availability of a flow transmission system design presented in Fig 1 and made of three pipes. The flow is transmitted from left to right, and the performances of the pipe are measured by their transmission capacity (tons of per minute). It is supposed that components 1 and 2 have three states: a state of total failure corresponding to a capacity of 0, a state of full capacity, and a state of partial failure. The component 3 only has two states: a state of total failure, and a state of full capacity. All state performances of the components are precise. We want to calculate the availability of the system by using Markov chain and we will compare to the IUGF proposed in [11] and the BUGF proposed in [12]. For every component  $G_j$ :

$p_i^j$ : The probability of being at state  $g_i^j$

$\lambda_{i,k}^j$ : The transition or degrading rate from state  $g_i^j$  to  $g_{i-k}^j$

$\mu_{i,k}^j$ : The repair rate from state  $g_i^j$  to  $g_{i+k}^j$

In our framework, both transition and repair rates are given by intervals. Here are the transition and repair rates for each of the three components:

- For the first component  $G_1$ , there are three possible states:
  1. State 1  $g_3^1 = 1.5$  represents completely successful operation.
  2. State 2  $g_2^1 = 1$  represents degraded successful operation.
  3. State 3  $g_1^1 = 0$  represents total failure.



Let the possible transition rates (in terms of hours) be:

$$\lambda_{3,1}^1 = [10^{-5}; 3 \times 10^{-4}] h^{-1}$$

$$\lambda_{2,1}^1 = [4 \times 10^{-5}; 5 \times 10^{-4}] h^{-1}$$

$$\mu_{2,1}^1 = [2 \times 10^{-2}; 5 \times 10^{-2}] h^{-1}$$

$$\mu_{1,1}^1 = [4 \times 10^{-2}; 8 \times 10^{-2}] h^{-1}$$

• For the second component  $G_2$ , there are three possible states:

1. State 1  $g_2^2 = 2$  represents completely successful operation.
2. State 2  $g_2^2 = 1.5$  represents degraded successful operation.
3. State 3  $g_1^2 = 0$  represents total failure.

Let the possible transition rates be:

$$\lambda_{3,1}^2 = [2 \times 10^{-5}; 6 \times 10^{-4}] h^{-1}$$

$$\lambda_{2,1}^2 = [3 \times 10^{-5}; 4 \times 10^{-4}] h^{-1}$$

$$\mu_{2,1}^2 = [3 \times 10^{-2}; 6 \times 10^{-2}] h^{-1}$$

$$\mu_{1,1}^2 = [3 \times 10^{-2}; 7 \times 10^{-2}] h^{-1}$$

• For the third component  $G_3$ , there is two possible states:

1. State 1  $g_2^3 = 4$  represents completely successful operation
2. State 2  $g_1^3 = 0$  represents total failure

Let the possible transition rates be:

$$\lambda_{2,1}^3 = [10^{-5}; 4 \times 10^{-4}] h^{-1}$$

$$\mu_{1,1}^3 = [5 \times 10^{-2}; 9 \times 10^{-2}] h^{-1}$$

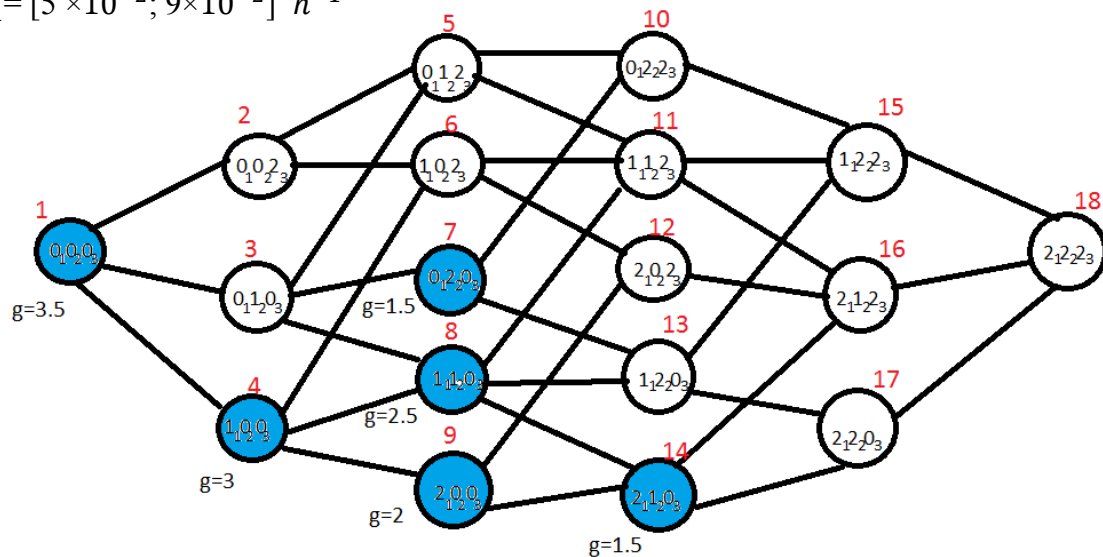


Figure 2: Markov chain of the system.

In this example, each of the components 1 and 2 have three possible states, while component 3 has only two states, hence the whole system has at most eighteen ( $3 \times 3 \times 2$ ) possible states degrading from the total functional state to the failure state as shown in Fig. 2. In this figure:  $0_j$  means that component  $j$  is in the completely working state,  $1_j$  means that component  $j$  is in the partial working state and  $2_j$  means that component  $j$  is in the completely failure state. In each state, the performance level  $g$  is calculated by taking into account the performance level of each component. For example, state 1 denoted by  $0_1 0_2 0_3$  refers to the state where all three components are completely working.

To obtain the performance level of the state : Components 1 and 2 are placed in parallel so the performance level resulting of them is the summation of the two performances which means  $2 + 1.5 = 3.5$ . Component 3 is placed in series, so the total resulting performance level  $g$  is the minimum, i.e.  $g = \min(3.5, 4) = 3.5$

In this example, we are studying the availability of the system for a demand level  $w = 1.5$ , we will consider the working states when the total performance level is greater than  $w$ , in Fig. 2 the colored states are the working states.

First, we calculate the corresponding availability of the system using IUGF and BUGF, results are grouped in table 1. After, we use the Forward-backward propagation contracting technique to solve the system of equations ensuring that the product of the stationarity vector times the transition matrix is equal to zero (18 equations) plus one last equation that is the summation of each stationarity is equal to one.

Our obtained results are shown in table 1. The result of the exact method is also given in this table.

Contraction technique	BUGF	IUGF	Exact method
[0.951; 0.998]	[0.95 ; 1.03]	[0.913 ; 1.09]	[0.952 ; 0.99]

Table 1: The availability at  $t \rightarrow \infty$  for each method.

Table 1 shows that the accuracy offered by the FBP contraction technique is close to that obtained when using the exact method; the interval availability of the contraction method is also more conservative than the other two methods (BUGF and IUGF). Therefore, the FBP contraction method turns out as an efficient technique for availability assessment since it offers accurate results and is more simple especially when handling complex systems.

## 5 CONCLUSION

We have applied the forward-backward contraction method on the equations obtained by the transition matrix of a Markov chain handling MSSs. Since multi-state systems are much more complicated than traditional two-state systems, an efficient method is desirable which can deal with system's size and complexity. Markov models for our best of knowledge are the most suitable methods, especially when the given data is imprecise. To be able to calculate the availability of a system by applying Markov model in the presence of imprecise data we need an accurate method for calculation. In our work, we proposed to use interval *Contraction method*. The main goal of the method is to reduce an initial big interval to its most possible minimum size. IUGF and BUGF are also good methods to calculate the imprecise availability of a MSS. These two methods are a pair of variants of the UGF methods, the first allows getting the availability by using imprecise probabilities, and the second calculates *bel* and *pl* by using mass functions. After testing these three approaches on different examples and comparing the results, we can clearly see that *the FBP Contraction method* is the most accurate amongst them, its results giving the smallest intervals containing those obtained by the exact method.

As future work, we can extend the proposed interval-based approach to more complicated scenarios and study the availability of a more complex system with multi-state components.

## ACKNOWLEDGEMENT

*This work was carried out in the framework of the regional project ODISYM.*

## REFERENCES

- [1] Marvin Rausand and Arnljot Hoyland (2004). System Reliability Theory Models, Statical Methods, and Applications. Wiley Series in Probability and Statics, 2<sup>nd</sup> edition .
- [2] P. Walley (1991), Statistical reasoning with imprecise probabilities, London: Chapman and Hall.
- [3] Matthias C.M. Troffaes, Jacob Gledhill, Damjan Skulj, Simon Blake (2015). Using Imprecise Continuous Time Markov Chains for Assessing the Reliability of Power Networks with Common Cause Failure and Non-Immediate Repair.9<sup>th</sup> International Symposium on Imprecise Probability: Theories and Application, Pescara, Italy.
- [4] Luc Jaulin, Michel Kieffer, Olivier Didrit, Eric Walter (2001). Applied Interval Analysis-Springer .
- [5] Amadou Gning and Ph Bonnifait (2006). Constraints propagation techniques on intervals for a guaranteed localization using redundant data. Automatica, 42(7): 1167–1175.
- [6] Gregory Levitin (2005). The universal generating function in reliability analysis and optimization. Springer.
- [7] Marija Mihova and Natasha Maksimova (2011). Estimation of minimal path vectors of multi state two terminal networks with cycles control. Mathematica Balkanica, 25(4): 437–447.
- [8] Marija Mihova and Nina Synagina (2008). An algorithm for calculating multi-state network reliability using minimal path vectors. In The 6th international conference for Informatics and Information Technology (CIIT 2008).
- [9] A. Lisnianski and G. Levitin (2003), Multi-State System Reliability: Assessment, Optimization and Applications. World Scientific Publishing413 Co Pte Ltd.
- [10] Mazen El Falou and Eric Châtelet (2011), Analytical Uncertainty Propagation for Availability Assessment of Stochastic Multi-state Systems, International Journal of Performability Engineering, 7(3): 251-261.
- [11] C.-Y. Li, X. Chen, X.-S. Yi, and J.-Y. Tao (2011), “Interval-valued reliability analysis of multi-state systems,” IEEE Transactions on465 Reliability, 60:. 323–330.
- [12] S. Destercke and M. Sallak (2013), An extension of Universal Generating Function in Multi-State Systems Considering Epistemic Uncertainties, IEEE Transactions on Reliability, 62(2), 504-514.
- [13] H. Haj Chhade, A. Gning, F. Abdallah, I. Mougharbel, and S. Julier (2014). Non parametric distributed inference in sensor networks using box particles messages. Mathematics in Computer Science, 8(3-4):455–478.
- [14] Fahed Abdallah, Amadou Gning, and Philippe Bonnifait (2008). Box particle filtering for nonlinear state estimation using interval analysis. Automatica, 44(3): 807-815.

- [15] Xiaopin Zhong, Mohamed Ichchou, and Alexandre Saidi (2010). Reliability assessment of complex mechatronic systems using a modified nonparametric belief propagation algorithm. *Reliability engineering & system safety*, 95(11): 1174–1185.
- [16] Araya, Ignacio, Bertrand Neveu, and Gilles Trombettoni (2009). "An interval constraint propagation algorithm exploiting monotonicity." *Int. Workshop IntCP* .